الجمهورية الجزائرية الديمقراطية الشعبية

République algérienne démocratique et populaire.

Ministère de L'enseignement Supérieure de la recherche scientifique.

Université 8 Mai 45 –Guelma-

Faculté des Mathématiques, d'informatique et des Sciences de la Matière

Département d'Informatique

Mémoire de Fin d'études Master Filière : Informatique

Option : Système Informatique.

Thème :

Sélection et élimination des attributs redondants pour la classification des gros corpus textuels

Encadré par :

Dr. Farek Lazhar

Présenté par :

Khaled Khodja Anfel

Juin 2023

Remerciements

D’abord, je remercie le bon DIEU de m’avoir donné santé et courage pour réaliser ce travail. Je tiens `à exprimer ma profonde gratitude à mon encadreur Mr Farek Lazhar pour m’avoir encadré et guidé et surtout pour ses judicieux conseils qui ont contribué `à alimenter ma réflexion. Je remercie chaleureusement les membres de jury pour l’honneur qu’ils nous ont fait en acceptant de juger mon travail. Mes sincères sentiments vont `à mes parents qui ont sacrifié jusqu’aujourd’hui et leurs encouragements tout le long de mon parcours.

Dédicaces

Je dédie ce modeste travail À mes chers parents ma mère et mon père,

Ainsi qu'à mes sœurs « Rayen, Alaa » et mes frère « Daya, Fouad » pour leur patience, leur amour, leur soutien et leurs encouragements. A ma tante et sans oublier mon neveu « Yanis », et A tous mes amis et a tous mes camardes Sans oublier tous mes professeurs De l'enseignement supérieur et surtout à Mr le chef département « zineddine kouahla » et a tout ceux qui m'ont aidé dans l'élaboration De ce travail.

Merci …

Abstract

Feature selection is a crucial process in the pre-processing of data for machine learning. Its aim is to reduce the feature space, speed up the learning process and improve the performance of classification algorithms, while avoiding over-learning. Various statistical methods, such as Information Gain (IG), Chi-squared test (Ch2), Improved Gini Index (IGI), etc., have proved effective in finding the most representative attributes in text corpora, using a reduced execution time compared with methods based on information theory.

However, these methods can generate a large number of redundant attributes, which can adversely affect the performance of classification algorithms. In this work, we aim to eliminate this redundancy by measuring the correlation between attributes that have similar or close IG scores. Correlation can be assessed using the mutual information between attributes. Thus, attributes that are strongly related to the target variable (class) and weakly correlated with the other attributes are considered to be the most informative.

Keywords : selection, feature, mutual information, correlation, redundancy, classification, text.

Résumé

Le processus de sélection des attributs est une étape essentielle dans le prétraitement des données pour l'apprentissage automatique. Son objectif est de réduire l'espace des attributs, d'accélérer le processus d'apprentissage et d'améliorer les performances des algorithmes de classification, tout en évitant le sur-apprentissage. Plusieurs méthodes statistiques ont été développées pour identifier les attributs les plus pertinents dans les corpus textuels, en se basant sur des mesures telles qu'Information Gain (IG), le test du Chi carré (Ch2) ou l'indice amélioré de Gini (IGI).

L'avantage de ces méthodes statistiques est qu'elles permettent de trouver rapidement les attributs les plus représentatifs tout en réduisant le temps d'exécution par rapport aux approches basées sur la théorie de l'information. Cela rend la sélection des attributs plus efficace et pratique dans le contexte de l'analyse de grands corpus de textes. En outre, il est important de choisir la méthode appropriée en fonction des caractéristiques spécifiques du corpus de données et des objectifs de l'apprentissage automatique.

Cependant, ces méthodes peuvent générer un grand nombre d'attributs redondants, ce qui peut nuire aux performances des algorithmes de classification. Dans ce travail, notre objectif est d'éliminer cette redondance en mesurant la corrélation entre les attributs qui ont des scores IG similaires ou proches. La corrélation peut être évaluée en utilisant l'information mutuelle entre les attributs. Ainsi, les attributs qui sont fortement liés à la variable cible (classe) et faiblement corrélés avec les autres attributs sont considérés comme les plus informatifs.

Mots clés : sélection, attribut, information mutuelle, corrélation, redondance, classification, texte.

Table des matières

Liste des Figures

Liste des Tableaux

Introduction générale

1. Problématique

La sélection des caractéristiques, (en anglais Feature Selection - FS), contrairement à l'extraction des attributs nous permet de sélectionner des sous-ensembles à partir de l'ensemble d'origine. On trouve plusieurs applications de la sélection des caractéristiques dans différents domaines tels que la catégorisation de textes [56].

La sélection des caractéristiques est bénéfique pour réduire la dimensionnalité du problème, elle conduit à minimiser le temps de calcul et à améliorer les performances de la tâche de catégorisation.

Une bonne classification ne peut se faire sans avoir trouvé un meilleur ensemble de caractéristiques (features) discriminatoires servant pour représenter efficacement les documents. L'objectif principal de la classification est de déterminer automatiquement dans quelle catégorie classer le texte en fonction de son contenu.

La classification de texte est le processus d'attribution de nouveau texte libre à des catégories (classes) en fonction des informations qu'il contient. Dans ce regard, nous devons préparer un ensemble de textes pré-marqués, appelé ensemble d'apprentissage, à partir duquel le modèle prédictif le plus approprié est généré.

La classification automatique de texte résout plusieurs problèmes du monde réel et peut déterminer à quelle classe appartient un document donné.

Les méthodes de sélection de caractéristiques (FS) ont suscité un grand intérêt au sein de la communauté de la classification de textes en raison de leur capacité à améliorer l'efficacité du calcul. Parmi ces méthodes, on retrouve la fréquence des documents (DF), le gain d'information (IG), l'information mutuelle (MI), le Chi-square (Ch2), etc. Ces méthodes se sont avérées efficaces pour améliorer les performances des modèles tout en réduisant les coûts et le temps d'apprentissage.

Le problème auquel nous a été confrontés est de trouver le meilleur moyen de sélectionner les caractéristiques pertinentes. Pour cela, nous proposons une approche basée sur la combinaison entre les méthodes de filtrage (IG et MI) pour sélectionner les attributs informatifs et éliminer ceux qui sont non pertinents ou redondants et obtenir un sous-ensemble optimal en vue de réduire les coûts de calcul et améliorer la performance de classification globale, sachant que les

termes (features) ayant de scores IG similaires ou proches peuvent impliquer un nombre de important de features redondants et inutiles pour le processus de classification.

Dans nos expérimentations, nous comparons notre méthode de sélection des caractéristiques avec des méthodes classiques telles qu'IGI, CH2 et MI. En général, nous constatons que notre méthode est plus efficace que les méthodes communes existantes.

2. Organisation du Mémoire

Ce mémoire est organisé en quatre chapitres comme suit :

Chapitre 1. Classification automatique des textes : Dans ce chapitre, nous présentons d'abord la définition e le processus de la classification de textes. Ensuite, nous examinons en profondeur les algorithmes couramment utilisés pour cette tâche. Enfin, nous abordons en détail les différentes méthodes de pondération utilisées dans ce contexte.

Chapitre 2. Sélection des features pour la classification des textes (état de l'art) : Dans ce chapitre, nous consacrerons notre présentation à la sélection des features pour la classification des textes. Notre accent sera particulièrement mis sur les métriques de sélection les plus fréquemment utilisées, telles que l'Information Gain (IG), la Mutual Information (MI), le test du Chi-carré (Chi2), et bien d'autres. Nous examinerons en détail ces différentes métriques de sélection et discuterons leur pertinence dans le contexte de la classification des textes.

Chapitre 3. La méthode proposée : Dans ce chapitre, nous offrons une description détaillée de toutes les étapes de notre méthode proposée pour la sélection des features.

Chapitre 4. Implémentation : Nous présentons les outils et l'environnement de programmation utilisés, ainsi que les bibliothèques essentielles pour l'apprentissage automatique. Dans ce chapitre, nous allons mettre en clair les étapes de l'implémentation montrée dans l'architecture du modèle proposée.

Classification Automatique des Textes

1.1 Introduction

Le domaine de la classification supervisée des textes est un aspect fondamental de l'apprentissage automatique. La classification supervisée des textes est une technique très utile qui est appliquée dans divers domaines. Par exemple, dans le domaine de la classification des documents, cette technique est utilisée pour classer les textes en fonction de leur sujet, de leur genre ou de leur pertinence. Cela permet de trouver rapidement des informations pertinentes dans une grande quantité de données textuelles. En effet, la classification des textes permet de définir le contenu d'un document en seulement quelques mots. En outre, la classification supervisée des textes offre une approche puissante pour organiser, analyser et extraire des informations utiles à partir de grandes quantités de données textuelles. Elle est appliquée dans de nombreux domaines différents pour aider les utilisateurs à trouver rapidement des informations pertinentes à partir de documents textuels.

Ce chapitre explique ce qu'est la classification supervisée des textes, comment elle fonctionne, les différents types de classification existants, et les algorithmes couramment utilisés. Il aborde également les techniques d'évaluation de la classification pour mesurer l'efficacité des modèles. En somme, ce chapitre fournit une introduction complète et claire à la classification des textes.

1.2 Définition

La classification du texte est le domaine d'application d'algorithmes de classification à des documents texte. Cette tâche consiste à affecter des documents à une ou plusieurs classes en fonction de leur contenu. En règle générale, ces catégories sont triées sur le volet par les humains [1].

1.3 Processus de classification

Il est possible de construire un classificateur à partir d'un corpus de documents étiquetés, généralement étiquetés à la main. Ce corpus est ensuite divisé en deux ensembles distincts: l'ensemble d'apprentissage et l'ensemble de tests. Dans un premier temps, le classificateur est entraîné à l'aide de l'ensemble d'apprentissage, puis son efficacité est testée à l'aide de l'ensemble de test. Dans certains cas, le processus se termine par une étape de validation du

classificateur à l'aide d'un ensemble de nouveaux documents. La figure 1.1 illustre parfaitement ces différentes étapes. Cette méthode permet d'obtenir un classificateur précis et efficace pour la classification de documents. Elle est très utilisée dans les domaines de la recherche documentaire, de l'analyse de sentiments, de la détection de fraude, de la gestion de la relation client, de la surveillance des réseaux sociaux et bien d'autres encore. En somme, la création d'un classificateur à partir d'un corpus de documents étiquetés est une technique très utile pour organiser, analyser et extraire des informations utiles à partir de grandes quantités de données textuelles [2].

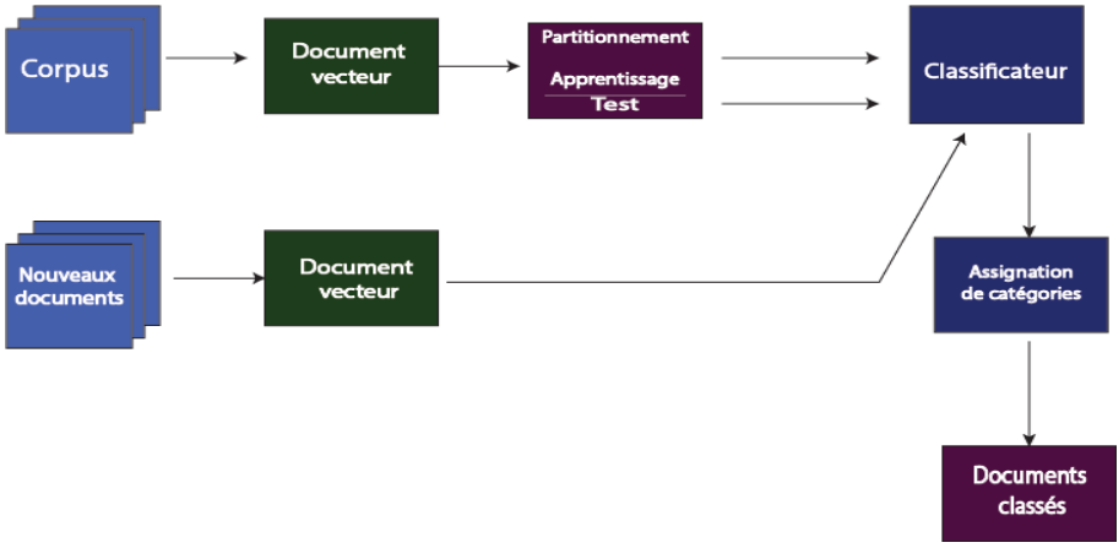


Figure 1. 1 : Processus de classification de documents [7].

1.4 Les Méthodes de classification automatique

Les méthodes de classification automatique font référence à un ensemble d'approches et d'algorithmes utilisés pour effectuer la classification de données de manière automatisée, sans intervention humaine directe.

1.4.1 Apprentissage non supervisé (Clustering)

L'apprentissage non supervisé consiste à apprendre à classer sans supervision. Au début du processus nous ne disposons ni de la définition des classes, ni de leurs nombres. C'est l'algorithme de classification qui va déterminer ces informations. Nous ne disposons pas non plus de données en entrée qui sont déjà classées, c'est aussi à l'algorithme de découvrir par lui-même la structure plus ou moins cachée des données et de former des groupes d'individus dont les caractéristiques sont communes [3].

La classification non supervisée consiste à trouver automatiquement une organisation cohérente pour un ensemble de documents homogènes afin d'établir des regroupements cohérents (classes ou clusters), ce qui correspond en statistique au clustering, terme utilisé en recherche d'information.

L'apprentissage non supervisé est utilisé dans plusieurs domaines tels que :

- Médecine : Découverte de classes de patients présentant des caractéristiques physiologiques communes.
- Le traitement de la parole : construction de système de reconnaissance de la voie humaine.
- Archéologie : regroupement des objets selon leurs époques.
- Traitement d'images, etc [4].

Il existe plusieurs types d'algorithmes d'apprentissage non supervisé tels que les algorithmes de partitionnement et les algorithmes de classification hiérarchique [5].

1.4.2 Apprentissage supervisé (Catégorisation)

La classification supervisée consiste à identifier la classe d'appartenance d'un objet à partir de certains traits descriptifs. Cette approche permet l'affectation automatique de documents dans des classes préexistantes [6].

L'objectif de la classification supervisée est essentiellement de déterminer des règles visant à classer des objets dans des classes à partir de variables qualitatives ou quantitatives qui caractérisent ces objets. Les méthodes sont fréquemment étendues aux variables quantitatives (régression). Au départ, on dispose d'un échantillon appelé échantillon d'apprentissage dont la classification est connue. Cet échantillon est utilisé pour effectuer l'apprentissage des règles de classification.

Il est important d'étudier la fiabilité de la classification et les comparés et appliqués dans le but d'évaluer le sous-ajustement ou le sur-ajustement (complexité du modèle). Souvent, un deuxième échantillon indépendant est utilisé, appelé échantillon de contrôle ou de validation [7].

Il existe de nombreuses méthodes de l'apprentissage automatique, et de nombreux algorithmes d'apprentissage supervisé. Parmi ces algorithmes, on peut citer notamment le Naïve Bayes, les

k-plus proches voisins, les arbres de décision, les machines à vecteurs de support, les réseaux de neurones, etc.

1.5 Quelques techniques de classification supervisée

1.5.1 Classification Bayésienne

Le classifieur naïf bayésien, développé par Lewis en 1998, est un algorithme faisant partie des méthodes de classification bayésienne probabiliste basées sur le théorème de Bayes [8]. Ce classifieur suppose que les caractéristiques descriptives sont indépendantes, bien que cette hypothèse soit généralement incorrecte.

La méthode de classification naïve bayésienne est un algorithme d'apprentissage supervisé (supervised machine learning) qui permet de classifier un ensemble d'observations selon des règles déterminées par l'algorithme lui-même [9].

C'est une méthode de classification statistique, ceci est principalement basé sur le théorème de Bayes. Elle est utilisée pour plusieurs applications.

Théorème de Bayes : Soient A, B et C trois événements. On a:

$$Pr\ [A/B,C]\ = \frac{Pr[B|A,C]Pr[A|C]}{Pr[B|C]} \tag{1.1}$$

Où :

- Pr [B|A, C] est la vraisemblance de l'événement B si A et C sont vérifiés.
- Pr [A|C] est la probabilité a priori de l'événement A sachant C.
- Pr [B|C] est la probabilité marginale de l'événement B sachant que C.
- Pr [A|B, C] est la probabilité a posteriori de A si B et C.

L'analyse discriminante est un cas particulier de l'approche bayésienne. Dans ce cas, les données d'apprentissage sont modélisées par des distributions gaussiennes. Sur la base de paramètres estimés, des fonctions discriminantes, sont fonctions, sont construites pour classer tout vecteur de caractéristiques [10].

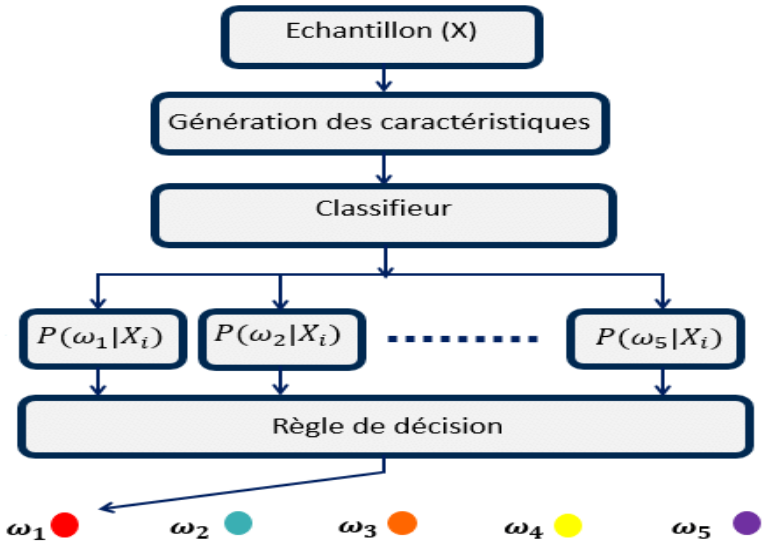


Figure 1. 2 : Schéma de classification Bayésienne [10].

Avantages

- La classification naïve bayésienne est très rapide pour la classification : en effet les calculs de probabilités ne sont pas très coûteux.
- La classification est possible même avec un petit jeu de données [11].

Inconvénients

- À la différence des courts documents, les longs documents présentent un défi majeur pour le classifieur naïf bayésien, car un vocabulaire étendu favorise les dépendances entre les descripteurs (termes). [11].

1.5.2 Machine à Vecteurs de Support (SVM)

SVM est un classificateur linéaire, ce qui signifie idéalement que les données doivent être linéairement séparables dans la classification textuelle. Il a initialement été développé comme un algorithme de classification binaire supervisée. Il s'avère particulièrement efficace de par le fait qu'il peut traiter des problèmes mettant en jeu de grands nombres de descripteurs, qu'il assure une solution unique (pas de problèmes de minimum local comme pour les réseaux de neurones) et il a fourni de bons résultats sur des problèmes réels [10] .

L'objectif de l'algorithme de la machine à vecteurs de support est de trouver un hyperplan dans un espace à N dimensions (N - le nombre de caractéristiques) qui classe distinctement les points de données [9].

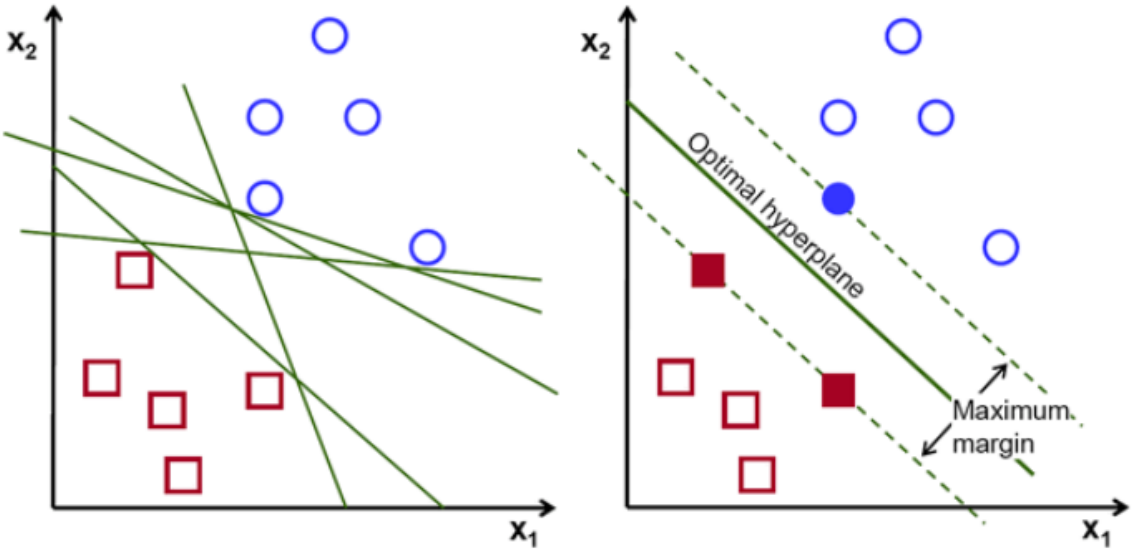


Figure 1. 3 : Schéma Hyperplans Possibles [9].

Pour séparer les deux classes de points de données, il existe de nombreux hyperplans possibles qui pourraient être choisis. Notre objectif est de trouver un plan qui a la marge maximale, c'est-à-dire la distance maximale entre les points de données des deux classes. La maximisation de la distance de marge fournit un certain renforcement, de sorte que les futurs points de données peuvent être classés avec plus de confiance [13].

Avantages

- Très efficace dans les hautes dimensions.
- Ils sont également efficaces dans le cas où la dimension de l'espace est plus grande que le nombre d'échantillons d'entraînement.

Inconvénients

- Lorsque le nombre d'attributs est beaucoup plus important que le nombre d'échantillons, les performances sont moins élevées.
- Étant donné l'existence de méthodes de discrimination de classe, elles ne fournissent pas d'estimations de probabilité.

1.5.3 Réseau Neuronaux

Le réseau de neurones artificiels, également appelé réseau de neurones, est l'un des algorithmes les plus couramment utilisés dans le domaine de l'apprentissage automatique. Cet algorithme peut être adapté à différents types d'apprentissage, qu'il s'agisse d'apprentissage non supervisé ou supervisé pour des tâches telles que la régression ou la classification. Bien qu'il soit

généralement non probabiliste, Specht a proposé une version probabiliste du réseau de neurones (Specht, 1990). Le réseau de neurones constitue aussi la base de l'apprentissage profond (deep learning).

Les réseaux de neurones ont été développés en tant que modèles mathématiques généraux qui imitent les neurones biologiques. Ils comprennent des éléments de traitement de l'information appelés neurones. Chaque neurone a son propre état interne interprété par une fonction d'activation. Il envoie ses activations sous forme de signaux à d'autres neurones. Les connexions entre les neurones sont réalisées par des liens dirigés et pondérés [15].

Un réseau de neurones artificiels (ANN) se compose de différentes couches de nœuds (ou neurones artificiels), dont une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. Chaque nœud ou neurone artificiel est connecté à un autre nœud et à des poids et des seuils associés. Si la sortie d'un nœud est supérieure au seuil spécifié, le nœud est activé et envoie les données à la couche réseau suivante. Sinon, aucune donnée n'est transmise à la couche réseau suivante.

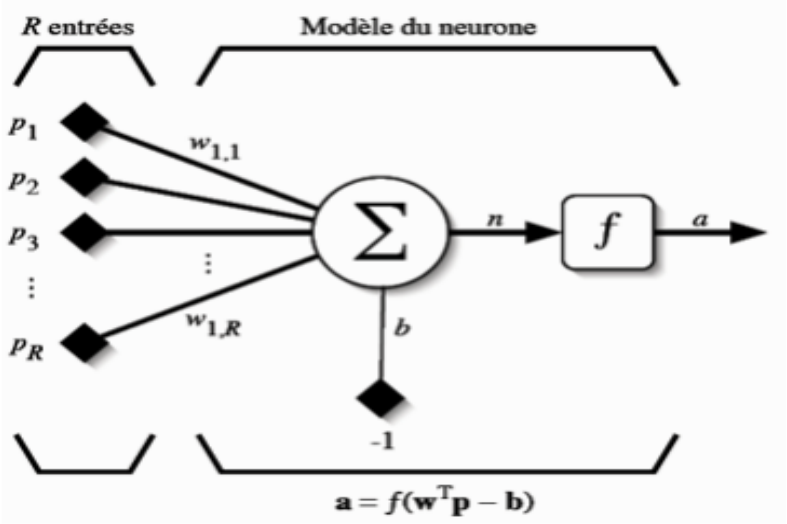


Figure 1. 4 : Schéma Réseau Neuronaux [14].

Avantages

- Capacité à représenter toute fonction linéaire ou non, simple ou complexe.
- Faculté d'apprentissage à partir d'exemples représentatifs, par « rétro-propagation des erreurs ». L'apprentissage (ou construction du modèle) est automatique.

- Facile à utiliser, beaucoup moins de travail personnel à fournir que dans l'analyse statistique classique. Aucune connaissance en mathématiques ou en informatique statistique n'est requise [17].

Inconvénients

- L'absence de méthode systématique permettant de définir la meilleure topologie du réseau et le nombre de neurones à placer dans la (ou les) couche(s) cachée(s).
- Le problème du sur-apprentissage (apprentissage au détriment de la généralisation) [14].

1.5.4 Forêts d'Arbres Décisionnels (Random Forest)

La forêt d'arbres décisionnels, communément appelée random forest, est un algorithme de classification et de régression développé par Leo Breiman dans son article de 2001. Cet algorithme s'inspire des travaux de Ho en 1995 et de Dietterich en 2000. L'objectif principal de la forêt d'arbres décisionnels est de générer une famille d'arbres de décision à partir de différents sous-ensembles de données et de variables. Pour ce faire, cet algorithme combine les principes du bagging (Breiman, 1996) et du bootstrap (Metropolis et Ulam, 1949) et les applique à la méthode des arbres de décision [18].

L'algorithme Random Forest est un algorithme de classification qui réduit la variance des prédictions d'un seul arbre décisionnel, améliorant ainsi ses performances. Pour ce faire, il combine plusieurs arbres de décision dans une approche de mise en sac [18].

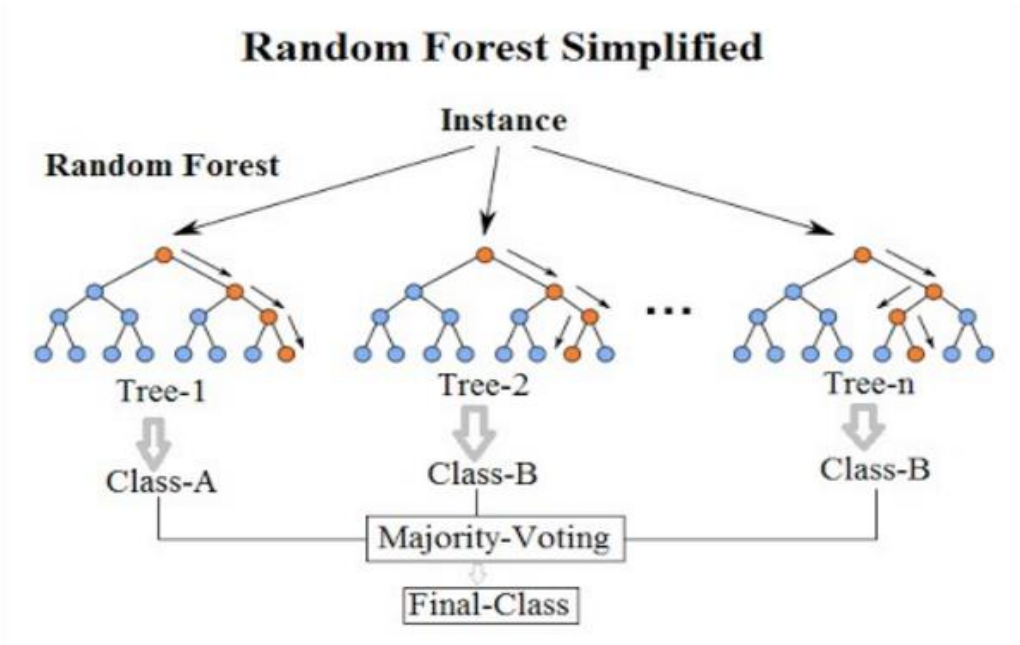


Figure 1. 5: Schéma Random Forest [18].

Un désavantage de la méthode de la forêt aléatoire par rapport à celle de l'arbre de décision réside dans la perte de lisibilité des modèles et donc dans la diminution de leur compréhensibilité.

1.5.5 Le Boosting

Il s'agit d'une méthode de classification qui émet des hypothèses moins importantes au départ. Plus d'hypothèses, plus il est vérifié, plus son indice de confiance augmente. Un algorithme élémentaire de boosting fonctionne de la manière suivante : Il commence par appliquer aux données d'apprentissage une certaine méthode (par exemple, un arbre de classification de type *C&RT* ou *CHAID*), dans laquelle chaque observation possède une pondération identique. Il calcule les classifications prévues, et applique des pondérations inversement proportionnelles à l'exactitude de la classification aux observations. En d'autres termes, il affecte une pondération plus forte aux observations difficiles à classer (qui présentent un taux de mauvaise classification élevé), et des pondérations plus faibles à celles qui sont faciles à classer (avec un faible taux d'erreur de classification). Dans le cadre de *C&RT* par exemple, différents coûts d'erreur de classification (pour les différentes classes) pourront s'appliquer, de façon inversement proportionnelle à l'exactitude de la prévision dans chaque classe. Il va ensuite appliquer à nouveau la classification aux données pondérées (ou avec d'autres coûts d'erreur de classification), et poursuivre avec l'itération suivante (application de la méthode analytique pour la classification des données).

L'algorithme de classification boosting a été conçu pour améliorer les performances de la classification en combinant plusieurs classificateurs faibles. Il est utilisé dans de nombreux domaines tels que la classification de textes ou le traitement du langage naturel. Le modèle le plus couramment utilisé pour le boosting s'appelle AdaBoost.

1.5.6 Arbre de Décision

Un arbre de décision est un algorithme d'apprentissage supervisé non paramétrique utilisé pour la classification et la régression. Il se compose d'une structure hiérarchique sous forme d'un arbre avec un nœud racine, des branches, des nœuds internes et des nœuds feuilles. Les Data Scientistes utilisent souvent l'arbre de décision ou Random Forest, qui est un modèle stable pour l'apprentissage supervisé, en termes de classification et de régression. Les arbres de décision sont couramment utilisés en apprentissage statistique, et ils suivent des principes de base dans leur construction.

Un arbre de décision commence habituellement par un nœud qui peut avoir plusieurs résultats possibles. Chacun de ces résultats peut ensuite conduire à d'autres nœuds, appelés nœuds enfants, qui offrent d'autres choix possibles.

Ces algorithmes d'apprentissage automatique ont pour objectif de structurer les données en une séquence de décision, afin de créer une représentation hiérarchique qui permet de distinguer les similitudes et les différences entre les attributs des exemples du jeu de données. Plus précisément, ces algorithmes cherchent à organiser les données en groupes homogènes en utilisant des critères de similitude entre les observations, tels que la distance euclidienne, la corrélation, etc. Ils utilisent ensuite cette structure pour classifier de nouvelles données ou pour extraire des informations utiles à partir des données existantes. Cette approche permet une compréhension plus approfondie de la structure des données et peut aider à identifier des tendances et des relations cachées qui ne seraient pas détectables avec des méthodes plus simples [21].

L'utilisation des arbres de décision dans les problèmes de classification se fait en deux étapes principales :

- La construction d'un arbre de décision à partir d'une base d'apprentissage.
- La classification ou l'inférence consistant à classer une nouvelle instance à partir de l'arbre de décision construit dans la première étape.

Avantages

- Le modèle de classification par arbre de décision est facilement compréhensible et interprétable. C'est un modèle de type boîte blanche : si l'on observe une situation particulière sur le modèle, il est facile de l'expliquer à l'aide de la logique booléenne. En revanche, les modèles de type boîte noire, tels que les réseaux de neurones, sont difficiles à comprendre et à expliquer.
- On peut évaluer la fiabilité d'un modèle en effectuant des tests statistiques.
- Cette méthode offre de bonnes performances sur les grands jeux de données : elle permet d'obtenir des résultats satisfaisants tout en utilisant peu de ressources de calcul.

Inconvenants

- La faible efficacité constitue le principal désavantage des arbres de décision pour moi, comparativement à d'autres algorithmes. Cependant, il peut arriver qu'il soit nécessaire de faire un compromis entre performance et interprétation.

- Un autre aspect à considérer est le risque de sur-apprentissage, qui représente un deuxième inconvénient majeur, voire une embûche à éviter. L'overfitting se produit lorsque l'algorithme s'entraîne avec une grande précision sur les données d'entraînement, mais ne parvient pas à produire des résultats satisfaisants sur de nouvelles données. Pour prévenir cette situation, il est primordial de procéder à un élagage méticuleux de l'arbre de décision.

1.5.7 K-Nearest Neighbor (K-NN)

L'algorithme des k plus proches voisins (KNN) est un algorithme d'apprentissage automatique supervisé simple et facile à mettre en œuvre qui peut être utilisé pour résoudre à la fois des problèmes de classification et de régression [23].

La méthode des k plus proches voisins (KNN K-Nearest Neighbors en anglais) repose sur une comparaison directe entre le vecteur de caractéristiques qui représente l'entité à classer et les vecteurs de caractéristiques qui représentent d'autres entités de référence. Cette comparaison implique un calcul de distance entre ces entités. Ensuite, l'entité à classer est affectée à la classe majoritaire parmi les k entités les plus proches, selon la distance utilisée [24].

Avantages

- La méthode des k plus proches voisins s'avère performante lorsque les données sont volumineuses et incomplètes [21].
- L'un des principaux avantages de l'algorithme des k plus proches voisins (KNN) est sa simplicité de mise en œuvre, car il ne nécessite aucune opération lourde. Il n'est pas nécessaire de construire un modèle complexe, de faire de nombreuses hypothèses ou d'ajuster plusieurs paramètres. De plus, KNN est un algorithme polyvalent qui peut être utilisé pour la classification, la régression ou simplement la recherche d'informations [25].

Inconvénients

- Cette méthode présente deux inconvénients majeurs, tout d'abord, le processus de classification d'une entité peut requérir un grand nombre d'opérations, surtout lorsque la base de référence est volumineuse. En outre, la méthode est sensible au bruit qui peut être présent dans les données d'apprentissage [24].
- Le temps de prédiction est très long, car il nécessite le calcul de la distance entre la nouvelle entité et tous les exemples de référence [21].

1.5.8 Régression Logistique (RL)

Régression logistique est un algorithme supervisé de classification**,** populaire en Machine Learning. Sont utilisés pour identifier la relation entre une variable dépendante continue et une ou plusieurs variables indépendantes. Lorsqu'il n'y a qu'une seule variable indépendante et une seule variable dépendante, mais elle est utilisée pour faire une prédiction sur une variable catégorielle par rapport à une variable continue. Une variable catégorielle peut être vraie ou fausse, oui ou non, 1 ou 0, etc.

La régression logistique peut être réalisée à l'aide de trois types de modèles :

A. Régression logistique binaire : C'est le type le plus courant de régression logistique, utilisé pour prédire une variable binaire (par exemple, vrai/faux, succès/échec).

B. Régression logistique multinomiale : Également appelée régression logistique multi classe, elle est utilisée lorsque la variable dépendante comporte plus de deux catégories exclusives.

C. Régression logistique ordinale : Ce type de régression logistique est utilisé lorsque la variable dépendante est ordonnée et présente une structure d'ordre [27].

1.6 Évaluation des modèles de classification

1.6.1 Matrice de Confusion

La matrice de confusion est un préalable nécessaire à la compréhension des performances des modèles de classification.

La technique de la matrice de confusion permet de mesurer les performances de la classification par apprentissage automatique. Avec ce type de modèle, vous pouvez utiliser des valeurs de vérité terrain connues sur un jeu de données de test pour différencier et classer les modèles. Le terme matrice de confusion est simple, mais déroutant [28].

		Classe prédites	
		Classe 1 = Positive	Classe 2 = Négative
Classes réelles	Classe 1 = Positive	VP	FN
	Classe 2 = Négative	FP	VN

Tableau 1. 1 : Matrice de Confusion.

— VP : vrai positif (true positif) : les cas où les prédictions sont positives, et la valeur réelle est effectivement positive.

— VN : vrai négatif (true négatif) : les cas où les prédictions sont négatives, et la valeur réelle est effectivement négative.

— FN : faux négatif (false négatif) : les cas où les prédictions sont négatives, et la valeur réelle est effectivement positive.

— FP : faux positif (false positif) : les cas où les prédictions sont positives, et ou la valeur réelle est effectivement négative.

1.6.2 Accuracy

C'est le pourcentage de prédictions correctes et est une mesure permettant d'évaluer les modèles de classification basée sur la matrice de confusion et qui mesure le taux de prédictions correctes sur l'ensemble des individus.

$$Accuracy = \frac{vrai\ positif + vrai\ négatif}{total} \quad (1.2)$$

1.6.3 Précision

La précision est le nombre d'observations correctement prédites par rapport à nombre total d'observations. Il mesure la capacité du modèle à ne pas faire d'erreur lors de prédiction positives.

$$Précision = \frac{vrai\ positif}{vrai\ positif + faux\ positif} \quad (1.3)$$

1.6.4 Rappel

Le rappel (sensibilité) est le taux de vrais positifs est défini comme le rapport entre le nombre de vrais positifs et le nombre total de positifs réels.

$$Rappel = \frac{vrai\ positif}{vrai\ positif + faux\ négatif} \quad (1.4)$$

1.6.5 Taux d'Erreur et de Succès

Indique une valeur relative au taux d'erreur, mesuré lors de la réception d'une transmission numérique, en fonction du niveau d'atténuation et/ou d'interférence du signal transmis.

Le taux d'erreur est calculé en divisant le nombre d'observations mal classées par le nombre total d'observations [29].

$$Taux\ d'erreur = \frac{vrai\ positif + faux\ négatif}{vrai\ positif + faux\ positif + vrai\ négatif + faux\ négatif} \quad (1.5)$$

Le taux de succès est le rapport entre les observations bien classées sur le nombre total des observations.

$$Taux\ de\ succès = \frac{vrai\ positif + vrai\ négatif}{vrai\ positif + faux\ positif + vrai\ négatif + faux\ négatif} \quad (1.6)$$

1.6.6 F1-Mesure

La mesure F1 est un indicateur qui met en relation le rappel et la précision, calculé par la formule suivante :

$$F1\text{-}Mesure = 2 \times \frac{précision \times rappel}{précision + rappel} \quad (1.7)$$

1 .7 Types de classification supervisée

1.7.1 Classification Binaire

La classification binaire (ou classification binomiale) est une transformation de données qui vise à répartir les membres d'un ensemble dans deux groupes disjoints selon que l'élément possède ou non une propriété / fonctionnalité donnée [2].

1.7.2 Classification Multi-Classes

La classification multi-classes est un problème de classification avec plus de deux classes. La classification multi classes suppose que chaque catégorie est affectée à une classe. Il existe de nombreuses façons de résoudre ce problème. Nous pouvons utiliser un classificateur binaire pour résoudre un problème de classification multiple. Les métriques standard utilisées dans le modèle multi-classes sont les mêmes que celles utilisées dans le cas de la classification binaire.

1.7.3 Classification Multi-Labels

La classification multi-labels n'est utilisée que si chaque document peut être regroupé simultanément à deux ou plusieurs classes (ou étiquettes).

1.8 Méthodes de pondération

Les méthodes de pondération sont des techniques qui attribuent différents poids à certaines données ou observations pour améliorer la précision de l'analyse. Elles sont largement utilisées dans divers domaines pour obtenir des conclusions plus précises et fiables.

1.8.1 BOW (Bag of Words)

Bag of Words (BoW) est une stratégie de traitement du langage naturel pour convertir un document texte en nombres pouvant être utilisés par un programme informatique. BoW est souvent implémenté comme un dictionnaire Python. Chaque clé du dictionnaire est définie sur un mot et chaque valeur est définie sur le nombre de fois où le mot apparaît [31].

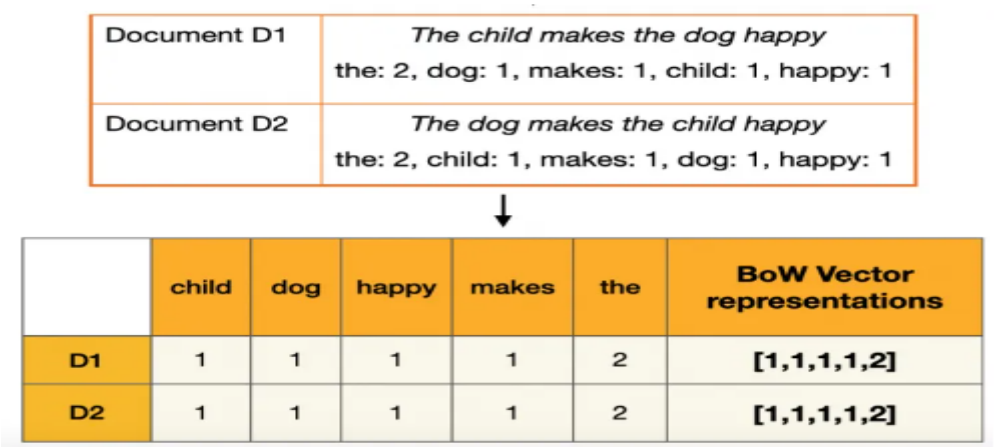
BoW est utilisé pour extraire des ensembles des features du texte pendant la phase de prétraitement des données. La stratégie consiste à décomposer un document en une liste de mots disparates et à noter combien de fois chaque mot est utilisé dans le document [31].

Avantages :

- Simplicité et applicabilité : le modèle du sac de mots est une représentation simple des données textuelles, facile à comprendre et à mettre en œuvre.
- Le modèle du sac de mots est clairsemé, ce qui signifie que la plupart des entrées du vecteur de caractéristiques sont nulles. Cela rend efficace le stockage et le traitement de grandes quantités de données textuelles [32].

Inconvénients :

- Le modèle de sac de mots considère toutes les occurrences d'un mot comme étant équivalentes, indépendamment de leur ordre d'apparition dans une phrase, ce qui implique qu'il est insensible à l'ordre des mots. Par conséquent, il ne peut pas saisir les relations entre les mots dans une phrase et leur signification [32].



Document D1	*The child makes the dog happy* the: 2, dog: 1, makes: 1, child: 1, happy: 1
Document D2	*The dog makes the child happy* the: 2, child: 1, makes: 1, dog: 1, happy: 1

	child	dog	happy	makes	the	BoW Vector representations
D1	1	1	1	1	2	[1,1,1,1,2]
D2	1	1	1	1	2	[1,1,1,1,2]

Figure 1. 6: Un Exemple sur BOW (Bag of Words)

1.8.2 TF-IDF (Term Frequency-Inverse Document Frequency)

La pondération TF-IDF est une méthode couramment utilisée dans la recherche d'informations et particulièrement dans l'analyse des textes. Cette technique statistique permet d'évaluer l'importance d'un terme présent dans un document, en tenant compte du corpus entier. Le poids du terme est proportionnel à son nombre d'occurrences dans le document et varie en fonction de sa fréquence dans l'ensemble du corpus. Cette méthode est souvent utilisée dans les moteurs de recherche pour évaluer la pertinence d'un document en fonction des critères de recherche de l'utilisateur, et il existe des variantes de la formule originale pour mieux s'adapter aux besoins spécifiques.

TF-IDF est une méthode alternative pour représenter un document en fonction des mots qu'il contient. Elle attribue un poids à chaque mot pour évaluer sa pertinence, plutôt que de simplement compter sa fréquence. En d'autres termes, elle remplace les nombres de mots par des scores [21].

Le score TF-IDF peut être calculé pour chaque mot, et ceux ayant un score plus élevé sont considérés comme plus importants, tandis que ceux ayant un score plus faible sont considérés comme moins importants.

$$TFIDF(t,d) = TF((t,d) \times IDF(t,d) = TF(t,d) \times log(N/DF(t) \qquad (1.8)$$

Où, TF(t,d) est le nombre d'occurrences du terme dans le document .

DF(t) est le nombre de documents qui contient le terme.

N : Le nombre total de documents.

1.9 Conclusion

Dans ce chapitre, nous avons présenté des détails sur la classification des textes, en conclusion, la classification automatique des textes est un domaine de recherche important dans le traitement automatique du langage naturel. Elle permet de catégoriser des documents en fonction de leur contenu, ce qui peut être utile dans de nombreux domaines tels que la recherche d'informations. Dans le chapitre suivant, nous aborderons de manière détaillée la sélection des caractéristiques (features).

Sélection des Features pour la Classification des Textes (Etat de l'art)

2.1 Introduction

Au cours des dernières années, la sélection d'attributs a suscité un grand intérêt dans des domaines tels que l'apprentissage automatique, l'exploration de données, le traitement d'images et l'analyse de données. Cette technique consiste à choisir un sous-ensemble d'attributs pertinents parmi un grand nombre d'attributs pour résoudre un problème spécifique. Bien qu'elle puisse être utilisée pour différentes tâches d'apprentissage ou d'exploration de données, nous concentrerons dans ce mémoire sur la sélection d'attributs pour la classification supervisée. L'objectif clé de cette approche est de trouver le meilleur sous-ensemble d'attributs possible qui doit présenter deux caractéristiques principales : d'une part, il doit être composé d'attributs pertinents, d'autre part, il doit éviter les attributs redondants. De plus, cet ensemble doit être conçu pour atteindre l'objectif visé, que ce soit la précision de l'apprentissage, la rapidité de l'apprentissage ou l'applicabilité du classifieur proposé.

Au cours de ce chapitre, nous aborderons de manière détaillée la procédure générale de sélection de features, ainsi que les méthodes et mesures de sélection les plus populaires dans le cadre de classification de textes.

2.2 Définition de la sélection des features

Sélection de features, une technique vise à choisir un petit sous-ensemble d'attributs pertinents parmi ceux d'origine en supprimant les d'attributs non pertinents, redondants ou bruyants.

La sélection des features conduit généralement à un meilleur apprentissage, c'est-à-dire une plus grande précision d'apprentissage, un coût de calcul inférieur et une meilleure interprétabilité du modèle [34].

La sélection des features est généralement définie comme un **processus** de recherche qui permet de trouver un sous-ensemble "pertinent" de features dans l'ensemble de départ. La notion de pertinence pour un sous-ensemble de features dépend toujours des objectifs et des critères du système [24].

C'est est une méthode, permettant de supprimer les attributs des données inutiles ou redondantes dans le contexte du problème à résoudre. Elle consiste à sélectionner automatiquement le sous-ensemble le plus utile pour la résolution du problème.

La définition proposée par dans (Pudil et Novovi, 1994) [36] est la suivante :

« *Étant donnée une fonction permettant de mesurer la qualité d'un sous-ensemble de caractéristiques (features), la sélection des caractéristiques est réduite au problème de recherche d'un sous-ensemble optimal par rapport à cette mesure* ».

Dans (Jain et al ., 2000) [36], la définition proposée est la suivante :

« Étant donné un ensemble de dimensions n, il faut sélectionner le sous-ensemble de dimension m tel que m<n, conduisant au taux d'erreur le plus faible ».

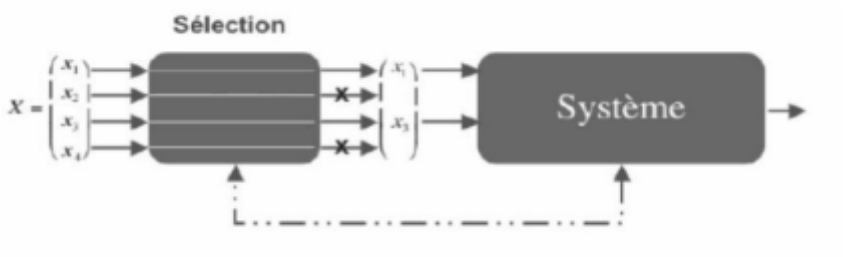


Figure 2. 1 : Principe de sélection des features.

2.3 Quelques avantages de la sélection des attributs

- Le processus de sélection des attributs vise à réduire leur nombre en éliminant les attributs non pertinents, redondants, inappropriés ou bruités. Il utilise des critères pour sélectionner les attributs qui sont en relation avec le problème traité et pour conserver efficacement les attributs importants. En d'autres termes, cette méthode permet d'identifier les caractéristiques significatives tout en éliminant les données inutiles ou nuisibles.
- La sélection des attributs peut améliorer la précision et les performances du classificateur en éliminant les données non pertinentes. Après cette étape, seuls les attributs les plus importants sont conservés, ce qui simplifie le modèle de classification et améliore sa capacité à résoudre le problème et à classer avec précision. En somme, la sélection des attributs permet d'optimiser le modèle en se concentrant sur les caractéristiques clés et en réduisant le bruit.

- Les attributs qui sont retenus sont ceux qui sont liés aux phénomènes d'intérêt, ce qui facilite leur interprétation. En effet, cette sélection permet de simplifier l'analyse en se concentrant sur les variables les plus significatives, ce qui facilite l'interprétation des résultats. En d'autres termes, la sélection des attributs permet d'obtenir une vue plus claire et plus simple des phénomènes étudiés.
- La sélection d'attributs permet de réduire le temps de calcul en simplifiant la complexité des calculs. Ainsi, le temps d'exécution de l'algorithme est amélioré, ce qui accélère la vitesse d'apprentissage et de traitement des données. En d'autres termes, en se concentrant sur les attributs les plus pertinents, la sélection réduit la complexité des calculs nécessaires pour traiter les données, ce qui améliore significativement la vitesse et l'efficacité de l'algorithme [38] .

2.4 Les objectifs de sélection des features

Les objectifs de la sélection des features sont les suivants :

- Créer des modèles plus simples et plus compréhensibles.
- Minimiser le taux d'erreur de classification [2].
- Diminuer le temps d'apprentissage, réduire les bases d'apprentissages de test [29].
- Améliorer la vitesse de la classification [2].
- Réduire les coûts de calcul et de stockage : Dans de nombreux problèmes d'apprentissage automatique, les données peuvent être volumineuses et comporter de nombreuses caractéristiques. Lorsque certaines caractéristiques sont redondantes ou peu utiles, les conserver peut entraîner des coûts élevés de calcul et de stockage. La sélection des features permet de réduire la dimensionnalité des données, en ne conservant que les caractéristiques les plus importantes, ce qui peut conduire à une réduction significative des coûts de traitement et de stockage [38].

2.5 Méthodes de sélection des features

Dans cette section, nous abordons les différentes méthodes de sélection de caractéristiques (features), qui peuvent être classées en deux grandes catégories : ***les méthodes supervisées*** et les méthodes ***non supervisées.***

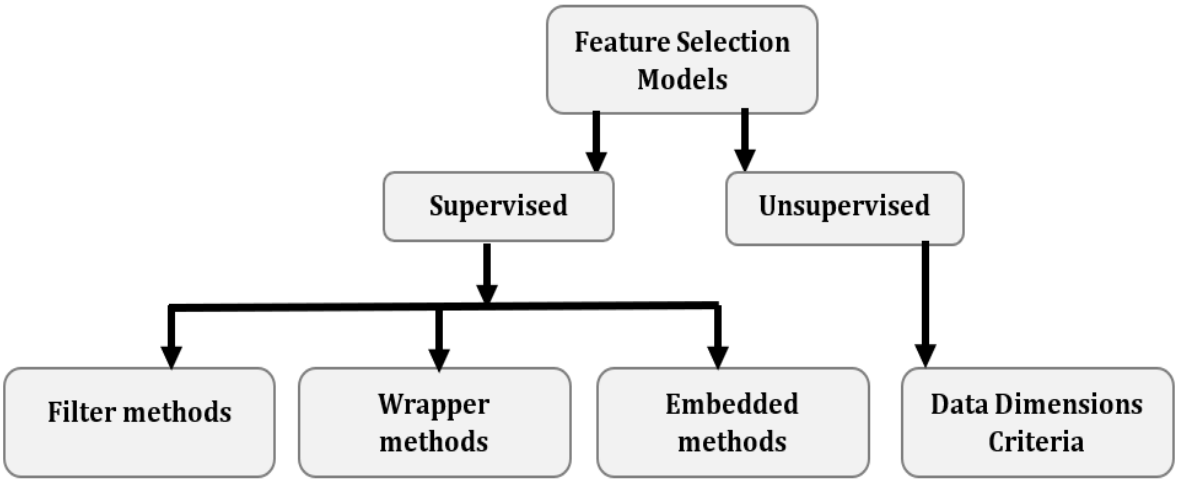


Figure 2. 2 : Méthodes de sélection des features.

2.5.1 Méthodes non Supervisées

Les méthodes de sélection de caractéristiques non supervisées sont très similaires aux méthodes supervisées, mais n'ont pas besoin d'informations d'étiquette pour la phase de sélection de caractéristiques et la phase d'apprentissage du modèle. Elles cherchent plutôt à identifier les caractéristiques qui préservent le mieux la structure multiple des données d'origine, sans utiliser d'étiquettes pour guider le processus.

Bien que la sélection de caractéristiques non supervisée puisse être très utile lorsque les données ne disposent pas d'étiquettes ou lorsque la qualité des étiquettes n'est pas suffisante, elle peut également être plus complexe et difficile à mettre en œuvre que la sélection de caractéristiques supervisée. Le choix de la méthode dépendra des spécificités de la tâche à accomplir et des caractéristiques des données disponibles.

En fin de compte, les méthodes de sélection de caractéristiques non supervisées sont des outils précieux pour le traitement des données, en particulier pour les tâches de regroupement. Toutefois, leur utilisation doit être considérée avec prudence et adaptée à chaque situation [39].

2.5.2 Méthodes Supervisées

Les approches utilisées pour évaluer un sous-ensemble de caractéristiques dans les algorithmes de sélection peuvent être classées en trois catégories principales : ”filter”, ”wrapper” et ”embedded".

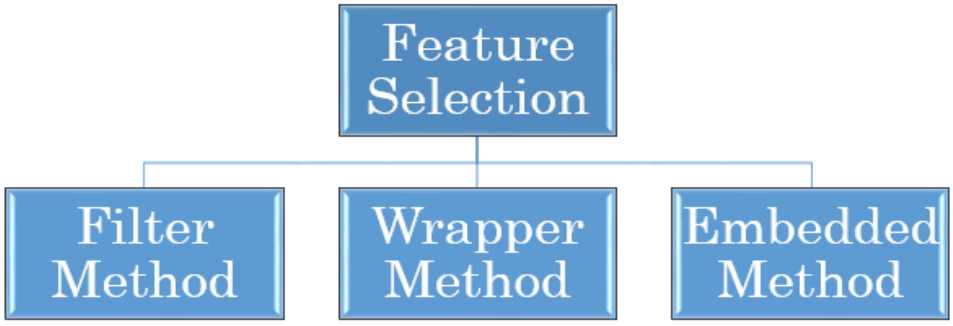


Figure 2. 3 : Approches supervisées de sélection des features.

2.5.2.1 Approche Filtre (Filtrage)

Cette méthode propose un sous-ensemble des attributs qui permettent d'expliquer la structure des données indépendamment de l'algorithme d'apprentissage choisi. Bien que les procédures de filtrage soient moins coûteuses en temps de calcul car elles évitent les exécutions répétitives des algorithmes d'apprentissage, leur inconvénient majeur est qu'elles ignorent l'impact des sous-ensembles choisis sur les performances de l'algorithme d'apprentissage [40].

L'approche par filtre intègre une mesure indépendante pour évaluer les sous-ensembles de caractéristiques sans impliquer un algorithme d'apprentissage. Cette approche est efficace et rapide à calculer (calcul efficace). Cependant, les méthodes de filtrage peuvent manquer des fonctionnalités qui ne sont pas utiles en elles-mêmes mais peuvent être très utiles lorsqu'elles sont combinées avec d'autres [41].

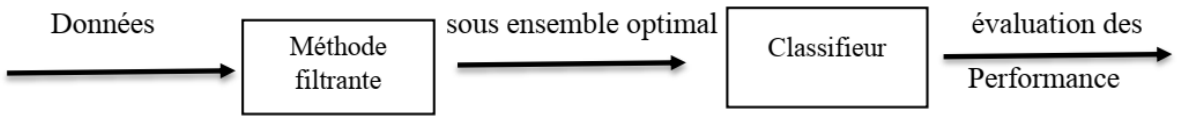


Figure 2. 4 : L'approche Filtre

2.5.2.2 Approche Embedded

Les méthodes Embedded sont des techniques d'apprentissage qui intègrent directement la sélection des attributs dans le processus. Les arbres de décision sont un exemple iconique, mais cette catégorie comprend toutes les techniques qui évaluent l'importance d'un attribut en cohérence avec le critère utilisé pour évaluer la pertinence globale du modèle. Autrement dit,

ces méthodes permettent de déterminer l'importance de chaque attribut dans le modèle en fonction de sa contribution à la performance globale de celui-ci [40].

Les techniques de sélection des attributs Embedded intègrent la sélection d'attributs directement dans le processus d'apprentissage, ce qui permet de déterminer simultanément le sous-ensemble optimal d'attributs tout en effectuant la classification. Contrairement aux techniques "Wrapper", les techniques Embedded sont spécifiques à un algorithme d'apprentissage donné, mais offrent l'avantage d'être plus rapides. Ces méthodes évaluent l'importance de chaque variable en cohérence avec la pertinence globale du modèle, tel que mesuré par l'algorithme d'apprentissage [41].

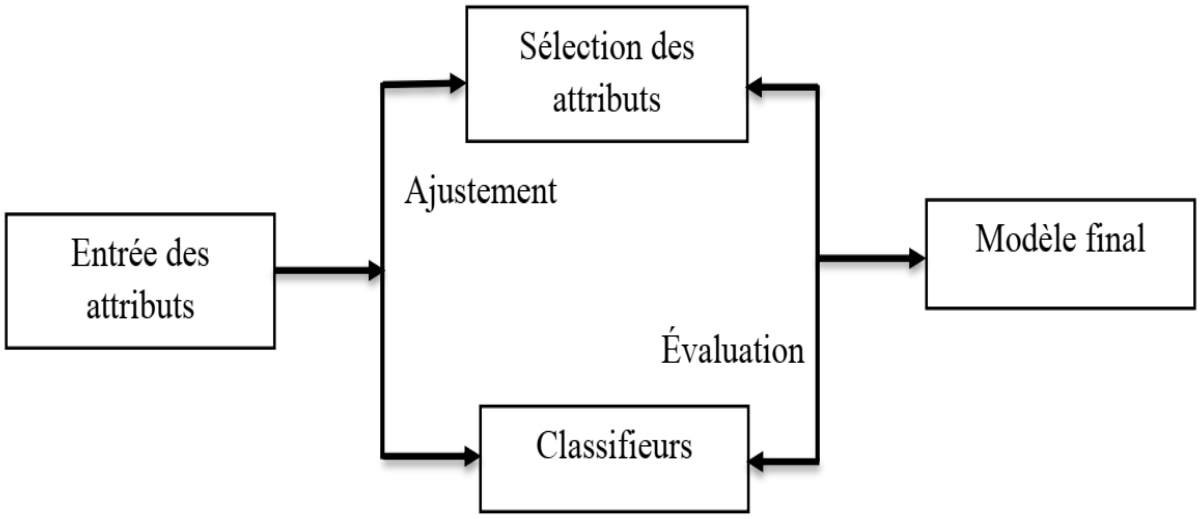


Figure 2. 5 : L'approche Embedded.

2.5.2.3 Approche Wrapper

La technique "filter" présente comme principal inconvénient de ne pas prendre en compte l'impact des attributs sélectionnés sur les performances du classificateur. Pour remédier à cette limitation, Kohavi et John ont proposé la technique "wrapper" dans leur travail [40]. Cette méthode évalue un sous-ensemble d'attributs en utilisant un algorithme de classification, produisant ainsi une précision plus élevée grâce à la sélection d'attributs correspondant mieux aux algorithmes d'apprentissage. Cependant, cette approche est plus coûteuse en termes de calcul que les méthodes "filter", car elle nécessite l'appel de l'algorithme de classification pour chaque sous-ensemble considéré. De plus, le sous-ensemble sélectionné dépend de l'algorithme de classification utilisé, ce qui implique de recommencer la sélection si l'algorithme est changé [39].

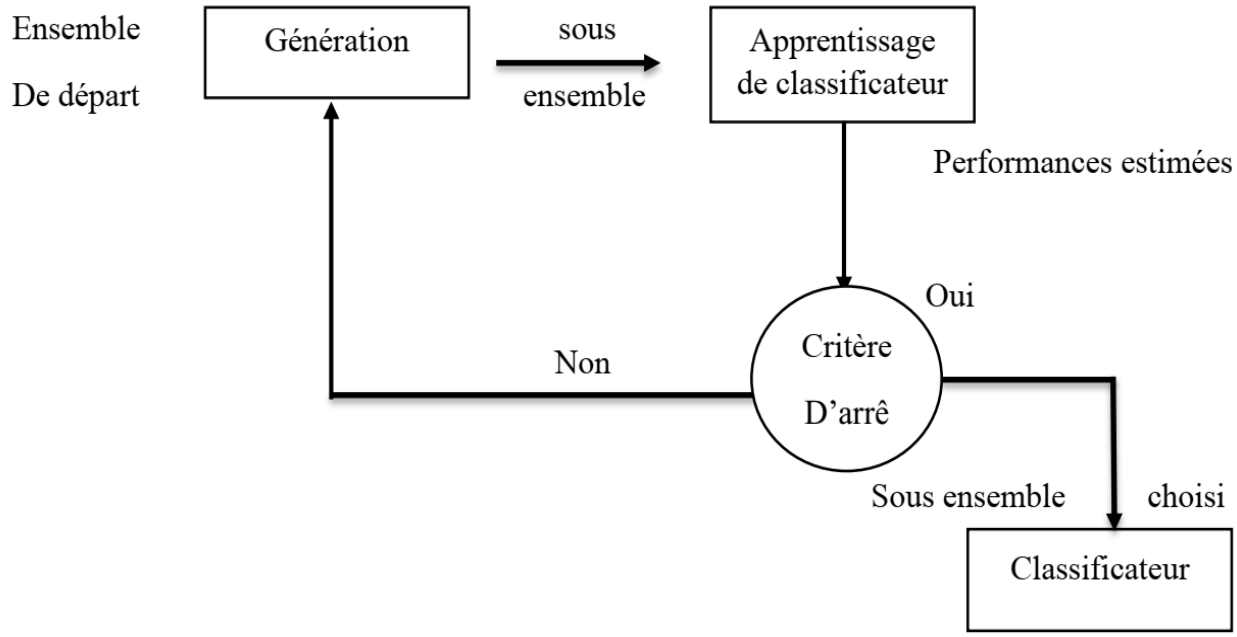


Figure 2. 6 : L'approche Wrapper.

2.6 Les avantages et les inconvénients des approches existantes

Le tableau 2.1 présente les avantages et les inconvénients des approches existantes [38] .

Méthode	**Avantages**	**Inconvénients**
Filter	- Exécution rapide - Coût de calcul faible	- Aucune interaction avec le classificateur
Wrapper	- Interaction avec le classificateur - Bonne performance de classification	- Coût de calcul élevé
Embedded	- Interaction avec le classificateur - Bonne performance de classification	- Coût de calcul élevé mais plus faible que Wrapper. - Exécution lente mais plus rapide que Wrapper. - Pas adapté à tous les types de classificateurs.

Tableau 2. 1 : Avantages et inconvénients des approches Filtres, Wrapper et Embedded.

2.7 Processus de sélection d'attributs

Une procédure générale pour développer un mode de sélection des attributs. Ce processus comporte quatre étapes distinctes, qui débutent avec un ensemble initial d'attributs. Les étapes sont les suivantes : la génération d'un sous-ensemble, l'évaluation du sous-ensemble, l'application des critères d'arrêt et la validation des résultats.

1. **La génération** d'un sous-ensemble est une méthode de recherche employée pour identifier des ensembles d'attributs candidats à évaluer [38].

2. Dans le processus de sélection des attributs, la performance de chaque sous-ensemble est mesurée par un **critère d'évaluation** spécifique. Ce critère est utilisé pour comparer la qualité de chaque sous-ensemble candidat avec celle du meilleur sous-ensemble précédent. Si le nouveau sous-ensemble s'avère être meilleur, il est sélectionné en remplacement du précédent. En répétant ce processus pour chaque sous-ensemble candidat, le sous-ensemble final sélectionné est celui qui présente la meilleure performance selon le critère d'évaluation [38 ,44].

3. Lors du processus de sélection d'attributs, il est important d'avoir **un critère d'arrêt** qui permette de déterminer si les attributs du sous-ensemble actuel ont atteint un niveau prédéfini. Pour ce faire, chaque sous-ensemble d'attributs doit être évalué et comparé au critère d'arrêt. Si le sous-ensemble actuel atteint les exigences prédéfinies, la sélection d'attributs s'arrête et le sous-ensemble courant est considéré comme le résultat final. Sinon, le processus de recherche continue jusqu'à ce que le critère d'arrêt soit satisfait [38,45].

4. La phase de **validation** implique généralement la vérification du sous-ensemble sélectionné à travers divers tests utilisant des données réelles ou simulées [43,46].

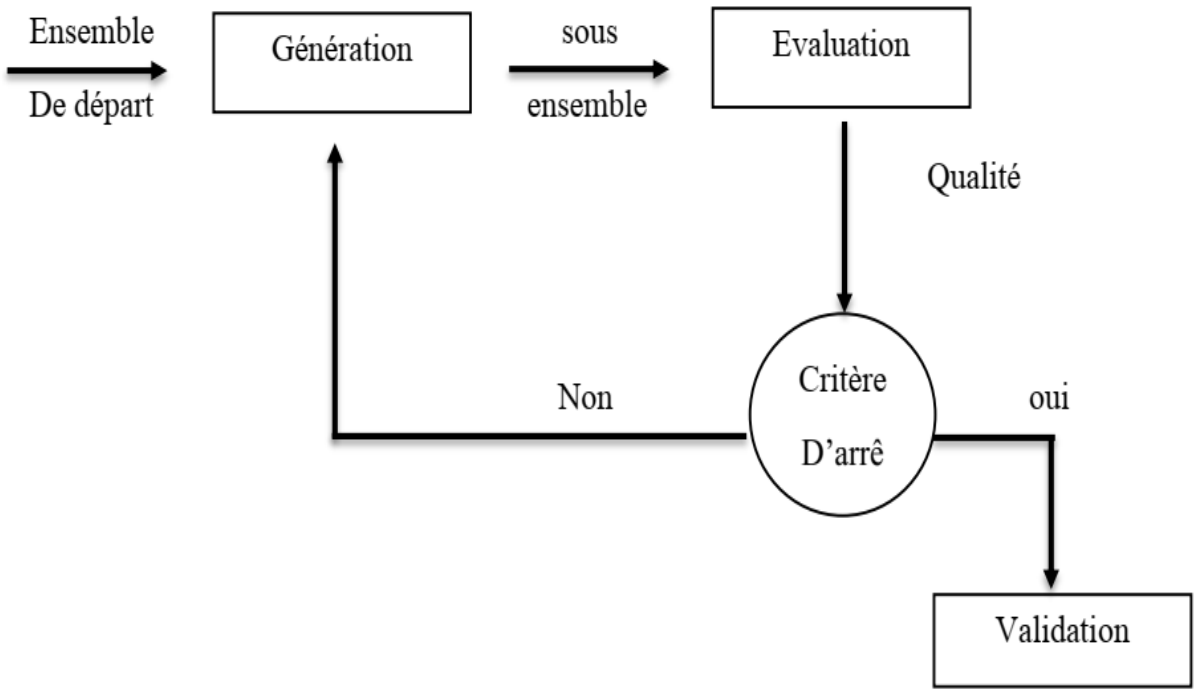


Figure 2. 7 : Processus de sélection de features.

2.8 Méthodes de sélection des features pour la classification des textes

Il y a des méthodes qui ont démontré leur efficacité dans la sélection des features pour la classification de textes, en obtenant de bons résultats exprimés en termes de la performance du classifieur:

2.8.1 Fréquence des Documents (en. Document Frequency - DF)

Le critère DF, abréviation de Document Frequency, représente le nombre de documents dans lesquels un terme apparaît dans un ensemble de données. Cette mesure est considérée comme le critère le plus simple pour la sélection des termes, car elle s'adapte facilement à un grand ensemble de données avec une complexité de calcul linéaire [47]. En d'autres termes, il s'agit d'une méthode efficace pour la catégorisation de texte qui consiste à repérer les termes les plus fréquemment utilisés dans un corpus donné. Cette méthode est simple mais efficace, car elle permet de cibler rapidement les termes les plus pertinents pour une recherche donnée. En effet, cette méthode permet de filtrer les termes qui ne sont pas significatifs pour l'analyse et de se concentrer sur les termes les plus représentatifs du contenu textuel.

2.8.2 Mutual Information (MI)

L'information mutuelle est un concept important dans la théorie de l'information de Shannon. C'est une méthode pour détecter le degré de corrélation entre les ensembles d'événements. L'analyse de corrélation ne peut détecter que le degré de corrélation linéaire entre les ensembles d'événements, tandis que les informations interactives contiennent des caractéristiques linéaires et non linéaires, qui sont un paramètre de corrélation relativement complet. Dans la théorie de l'information de Shannon, l'information est mesurée et étudiée mathématiquement. La quantité d'informations est mesurée par le degré d'incertitude de divers symboles dans la source qui suit l'information [49].

$$MI(x,y) = log\ \frac{P(x,y)}{P(x)P(y)} \tag{2.1}$$

Où P(x) et P(y) sont les probabilités marginales de x et y.

P(x, y) est la probabilité conjointe de x et y.

2.8.3 Information Gain (IG)

Gain d'information est une méthode supervisée utilisée comme critère de sélection dans le domaine de l'apprentissage automatique, et est utilisé pour déterminer les meilleures caractéristiques/attributs qui rendent le maximum d'informations sur une classe. Le gain d'information calcule la différence entre l'entropie avant et après la division et spécifie l'impureté dans les éléments de classe [46]. IG correspond à la réduction de l'incertitude dans l'identification des catégories sachant quand la valeur du feature a été observée. Pour un terme t et un ensemble des catégories C.

$$IG(C,t) = H(C) - H(C/t) \tag{2.2}$$

$$IG(C,t) = -\sum_{j=1}^{m} P(C_j)\, log\left(P(C_j)\right) + P(t)\sum_{j=1}^{m} P(C_j/t)\, log\left(P(C_j/t)\right) + P(\bar{t})\sum_{j=1}^{m} P(C_j/\bar{t})\, log\left(P(C_j/\bar{t})\right) \tag{2.3}$$

Où H(C) et H(C/t) sont les entropies avant et après la division de l'ensemble de données, m est le nombre de classe (C_j) est la probabilité d'un document appartenant à la classe (c_j).

P (C_j/t) et P (C_j/t) sont les probabilités d'une classe c inclut la présence et l'absence du terme t.

$P(C_j/\overline{t})$ *et* $P(C_j/\overline{t})$ sont les probabilités conditionnelles de la classe étant donné la présence ou l'absence du terme t.

2.8.4 Gini Index (IGI)

L'algorithme de sélection de caractéristiques basé sur l'indice de Gini, développé par Wen Qian Chang et ses collègues en 2007, utilise une approche innovante pour mesurer l'importance des attributs. Contrairement à l'approche originale de l'indice de Gini, qui se concentrait sur la mesure de l'impureté des attributs pour la classification, cet algorithme utilise une fonction de mesure de la pureté des attributs. En effet, de nombreuses études ont montré que plus la pureté de l'attribut est élevée, plus il est pertinent pour la tâche de sélection des caractéristiques. Ainsi, cette méthode permet de sélectionner les features les plus significatives pour une tâche donnée en utilisant l'indice de Gini [50].

$$Gini\ (t,c)\ =\ \sum_{i=1}^{m} P\ (t/c_i)^2 . P\ (c_i/t) \tag{2.4}$$

Sachant que P (t/c_i) et P (c_i/t)sont la probabilité de t sachant que la catégorie c_i est présente, et la probabilité de c_i sachant que le terme t est présent, respectivement.

2.8.5 Chi-Square

L'algorithme de sélection de caractéristiques supervisé Chi-square (Yang & Pedersen, 1997) permet de tester l'indépendance de deux variables statistiques. Ce test est effectué en calculant la corrélation du terme **t** avec la classe **C**, cela signifie que la caractéristique et la classe sont indépendantes, donc la caractéristique n'apporte aucune information importante sur la classe. Cette méthode est utilisée pour sélectionner les caractéristiques qui sont les plus pertinentes pour la classification. La formule de Chi-square est définie comme suit :

$$X^2 = \frac{N(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \tag{2.5}$$

Où : N est le nombre total des documents dans le corpus.

A est le nombre de documents dans la classe c_i contenant le terme t.

B est le nombre de documents contenant le terme t dans d'autres classes.

C est le nombre de documents de la classe C_i qui ne contiennent pas le terme t.

Et D est le nombre de documents qui ne contiennent pas le terme T dans d'autres classes [51].

2.9 Conclusion

Dans ce chapitre, nous avons offert une présentation structurée et complète de la sélection des features. Nous avons commencé par donner des définitions et des explications sur les objectifs de la sélection des features. Ensuite, nous avons examiné les méthodes de sélection supervisées et non supervisées, en détaillant les différentes approches de sélection supervisées, telles que Filter, Wrapper et Embedded, ainsi que leurs avantages et leurs inconvénients respectifs. Nous avons également présenté les principes de fonctionnement des métriques les plus couramment utilisées pour la sélection de données catégorielles, notamment CH2, IG, MI, GI et DF. Enfin, nous avons préparé le terrain pour le chapitre suivant pour présenter notre méthode de sélection des features qui se focalise base sur l'élimination des features redondants.

Chapitre 3 : La méthode proposée

3.1 Introduction

Les méthodes de sélection de caractéristiques pour la classification des textes peuvent être utilisées pour identifier et supprimer les attributs inutiles, non pertinents et redondants qui ne contribuent pas à la précision d'un modèle prédictif, ou qui peuvent en fait réduire la précision du modèle. Parmi les méthodes couramment utilisées, nous trouvons Chi-Square test (Chi2), Document-Frequency (DF), Mutual Information (MI), etc., qui sont des méthodes statistiques qui emploient seulement la fréquence des documents du terme pour calculer son score. Malheureusement, ces techniques traitent le mot comme une unité isolée sans prendre en considération son interaction avec les autres features (mots), ce qui peut conduire à sélectionner des features redondants et inutiles.

Dans ce chapitre, nous allons tout d'abord présenter l'inconvénient des métriques existantes de sélection des features en mettant l'accent sur le problème des features redondants de l'approche Filter. Puis, nous présentons notre méthode proposée qui vise à sélectionner les features les plus informatifs pour le processus de classification des textes et supprimer ceux qui sont redondants et inutiles.

3.2 Problème de features redondants

Les caractéristiques (features) redondantes sont celles qui sont corrélées à d'autres caractéristiques et qui ne sont pas pertinentes pour le processus de classification. Autrement dit, elles ne contribuent pas à améliorer la qualité de l'analyse des données ou la compréhension des phénomènes étudiés, et il est important de veiller à ce que les caractéristiques sélectionnées soient pertinentes et non redondantes [4]. En somme, il est essentiel de sélectionner soigneusement les caractéristiques pertinentes pour une analyse efficace des données et pour une meilleure compréhension des phénomènes étudiés.

Dans le domaine de l'analyse de données, la redondance en termes d'information mutuelle fait référence à la situation où certaines variables ou caractéristiques dans un ensemble de données fournissent des informations similaires ou redondantes [54]. Cette situation peut se produire lorsqu'il existe une forte corrélation entre deux variables et que l'une d'elles peut être prédite ou expliquée à partir de l'autre. La présence de ces informations redondantes peut entraîner une surcharge d'informations et une complexité inutile lors de l'analyse des données, ce qui peut rendre la compréhension des données plus difficile et conduire à des erreurs d'interprétation

[54]. Par conséquent, il est important d'identifier et d'éliminer les variables redondantes avant de procéder à l'analyse des données. Cela peut être accompli en utilisant des méthodes telles que l'analyse de corrélation et la réduction de dimensionnalité, ce qui permet d'identifier les variables qui sont fortement corrélées et de les supprimer de l'ensemble de données [54]. En plus, la réduction de la redondance des données permet d'obtenir une meilleure qualité d'analyse et de faciliter la compréhension des données [55].

3.3 Inconvénient majeur des approches et métriques existantes

L'un des principaux inconvénients des métriques de sélection des features, telles que Chi2, IG, MI, etc., est que chaque mot est traité indépendamment des autres mots, et la cooccurrence de certains mots ont un pouvoir discriminant plus élevé que les mots évalués individuellement. De plus, ces métriques ne sont pas fiables pour les mots à basse fréquence, qui sont filtrés en raison de leur poids et ne sont comptabilisés que lorsqu'ils apparaissent dans le document. Par conséquent, ils ignorent la fréquence d'occurrence des termes dans les documents. D'autre part, les termes à haute fréquence apparaissant dans quelques documents sont généralement considérés comme discriminants dans les corpus réels, à l'exception des mots vides. Par conséquent, la fréquence des termes et la corrélation sémantique entre eux doivent être prises en compte lors de la sélection des features pour une classification de texte précise et efficace.

De plus, la classification de texte est un processus complexe impliquant plusieurs étapes, y compris la représentation des données textuelles, la sélection des features et la classification elle-même. Par conséquent, il est important de choisir une méthode de sélection de features appropriée en fonction des caractéristiques de la tâche de classification et des données textuelles utilisées [55].

3.4 Méthode proposée

Le travail que nous proposons est basé sur la recherche des mots corrélés qui ont des scores proches pour éliminer les features (mots) redondants et non pertinents et garder ceux qui sont informatifs pour la variable classe, conduisant à déterminer un sous-ensemble optimal pour but d'améliorer les performances du modèle et minimiser le temps d’exécution.

La figure suivante présente un aperçu général de notre méthode :

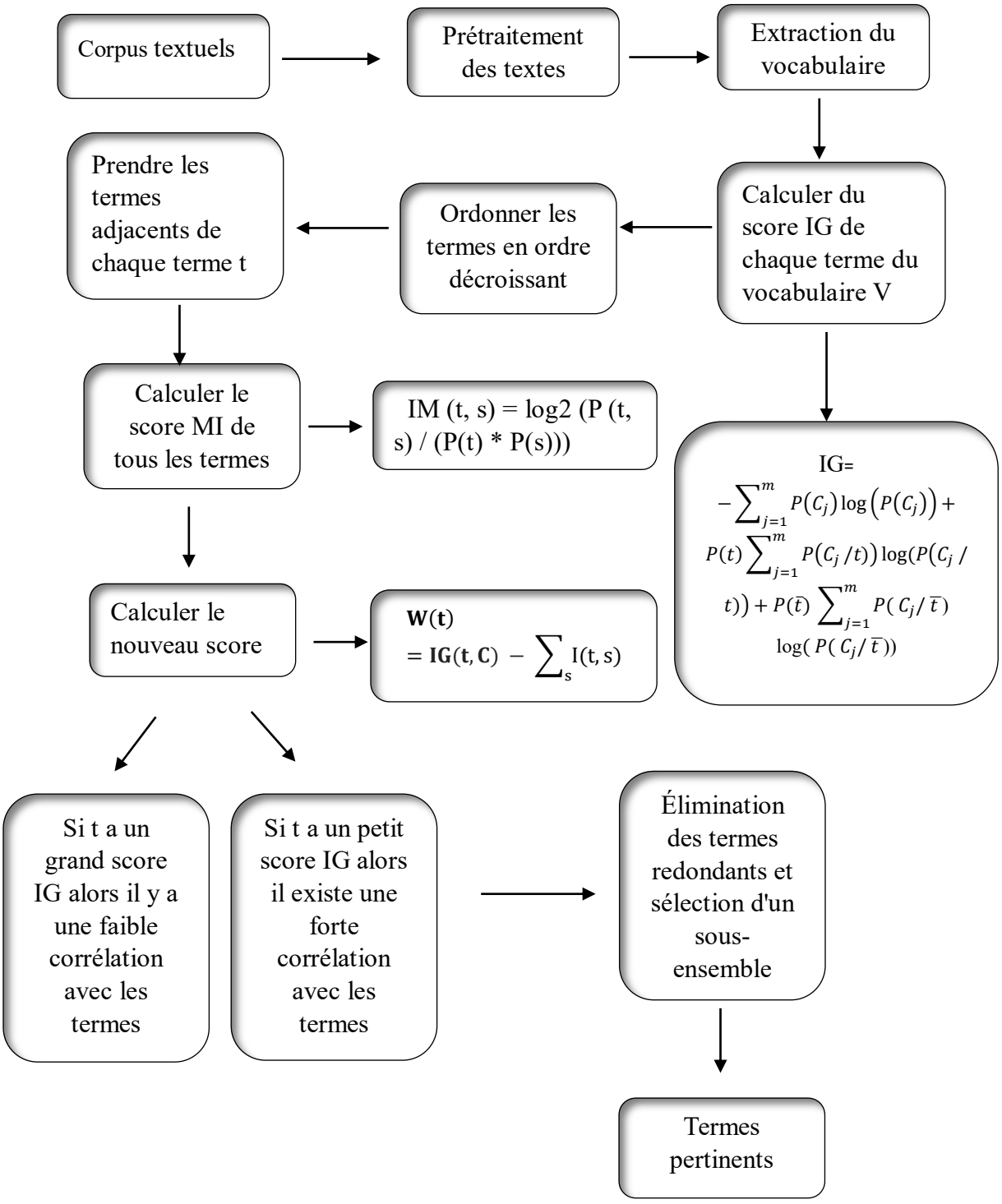


Figure 3. 1 : Architecture de notre méthode

La démarche de notre méthode est expliquée comme suit :

1) Nous choisissons un dataset textuel et le traitons (faire un prétraitement des textes et extraire le vocabulaire des mots uniques).

a) Calcul du score IG de chaque terme en suivant les étapes suivantes : Calcul de l'entropie de la variable classe C en utilisant la formule suivante :

$$H(C) = -\sum P(C_j) \log 2\, P(C_j)$$

b) Calcul de l'entropie de conditionnelle H $(C|\, t)$ en utilisant la formule suivante :

$$H(C|\, t) = -\sum P(t)\, H(C_j\, t) = -\sum p(t, C_j) \log (C_j / t)$$

c) Calcul de l'information gain pour chaque mot t en utilisant la formule suivante :

$$IG(t) = -\sum_{j=1}^{m} P(C_j) \log\left(P(C_j)\right) + P(t) \sum_{j=1}^{m} P(C_j/t)) \log(P(C_j/t)) +$$

$$P(\bar{t}) \sum_{j=1}^{m} P(C_j/\bar{t}) \log(P(C_j/\bar{t})).$$

L'objectif de cette étape est de mesurer l'importance de chaque terme dans un ensemble de données.

2) Ordonner les termes de plus haut au plus bas selon les scores IG. L'objectif de cette étape est de mettre en évidence les termes les plus informatifs et discriminants pour la tâche considérée.

3) Prendre les termes à gauche et à droite de chaque terme t. pour calculer la corrélation entre le terme t et leur termes adjacents.

4) Calculer l'information mutuelle (MI) entre chaque terme t et ses termes adjacents (dans la liste ordonnée) en utilisant la formule suivante :

$$IM(t, s) = \log 2\, (P(t, s) / (P(t) * P(s))).$$

5) Calculer le nouveau score pour permettre de mesurer l'importance du terme en utilisant la formule suivant :

$$W(t) = IG(t, C) - \sum_{s} I(t, s)$$

Le but de cette étape vérifie s'il existe une dépendance et une corrélation forte entre le terme t et les termes adjacents.

3.4.1 Prétraitement des textes

Dans tout processus de classification de texte, l'étape de prétraitement contribue efficacement sur la performance du classificateur. Le nettoyage peut supprimer le bruit généré par certains attributs et éliminer les attributs inutiles, les mots qui participent un rôle négatif dans le cadre de la classification et de sa contribution au texte, c'est l'étape la plus importante pour développer le modèle avec de bonne performance [56]. Dans cette étape, nous nous intéressons aux données textuelles, qui nécessitent le plus de prétraitement en raison de leur ambiguïté. Et pour cela nous allons passer par plusieurs étapes de suppression et de changement :

- ***La suppression des mots vides*** (ou stop-words) consiste à éliminer les mots qui ne portent pas ou peu d'information dans un texte, car ils sont considérés comme non significatifs. Les déterminants comme "le", "la" ou "du", ainsi que les conjonctions de coordination comme "donc" ou "car", sont des exemples de mots vides. Lorsque les textes sont représentés par des mots simples, ces mots vides sont généralement éliminés. Toutefois, lorsqu'il s'agit de la représentation de groupes de mots ou de concepts, ces mots vides peuvent être nécessaires pour identifier des groupes nominaux et sont donc conservés [57].

- ***La tokenisation*** consiste à découper un texte en unités atomiques appelées "tokens". L'objectif de cette étape est de segmenter le texte en mots ou en phrases. Par exemple, si l'on prend la phrase "Une guitare possède six cordes", la tokenisation en mots donnerait les tokens suivants : "Une", "guitare", "possède", "six" et "cordes" [58].

- ***La lemmatisation*** consiste à transformer les mots d'un texte en leur forme canonique ou fondamentale, c'est-à-dire leur forme de base. Par exemple, pour un verbe, cela peut être son infinitif, et pour un nom, son masculin singulier. L'objectif principal de la lemmatisation est de conserver le sens des mots utilisés dans le corpus, tout en réduisant les formes flexionnelles des mots à leur forme de base. De cette manière, les variantes d'un même mot peuvent être regroupées pour faciliter l'analyse et la modélisation des données textuelles [58].

- ***La racinisation*** (ou stemming en anglais) est une technique qui vise à réduire les mots d'un texte à leur racine commune en éliminant les suffixes et préfixes. L'objectif est de

ne conserver que l'origine des mots, en supprimant les variations flexionnelles. Contrairement à la lemmatisation qui utilise un dictionnaire pour identifier la forme canonique des mots, la racinisation est un processus plus simple et rapide qui consiste essentiellement à réduire les mots à leur racine commune. Cette technique est souvent utilisée en prétraitement de données textuelles pour faciliter l'analyse et la modélisation [58].

- ***La normalisation*** simplifie les différentes variantes d'un mot, supprime des mots fréquents, filtrage de mots clés [59].

Les méthodes de prétraitement de données ont pour objectif de transformer les données dans un format qui convient à l'extracteur de features, en éliminant les informations redondantes ou inutiles. En suivant ces étapes, vous pourrez obtenir une structure de données prête à être analysée, optimisant ainsi le processus d'analyse et de modélisation.

3.4.2 Extraction du vocabulaire

L'objectif principal de l'extraction du vocabulaire est utilisé pour convertir un texte de n'importe quelle configuration qu'il peut être facile de traiter par apprentissage supervisé. De plus, elle présente des connaissances concernant les textes comme la fréquence maximale des termes pour chaque texte. Le choix des mots-clés associés et l'identification de la méthode permettent de coder ces mots-clés en apprentissage automatique supervisé. Ces mots clés peuvent avoir un impact énorme sur la capacité des techniques de classification à extraire le meilleur modèle.

Le vocabulaire est créé en sélectionnant les mots uniques restants après les étapes précédentes de prétraitement des textes courants en supprimant les contenus indésirables tels que les balises HTML, les caractères spéciaux, les chiffres, les mots vides, la conversion des textes en minuscules et la lemmatisation. Supprimer les signes de ponctuation (point, virgule, point-virgule, etc.), caractères non ascii, et Stop-Words. Convertir les majuscules en minuscules. Élimination des mots fonctionnels : Ils pourraient être des articles (de, des, les, etc.), des pronoms (ses, moi, etc.) ou de certains verbes (sont, serons, etc.). Élimination des mots dont la taille est inférieure à un seuil donné. Ces mots uniques constituent le vocabulaire de base qui sera utilisé pour l'analyse des données textuelles ou pour l'apprentissage automatique.

Par exemple dans la phrase « « Je suis une étudiante de master informatique », le vocabulaire extrait de cette phrase est la liste des mots V= [je, suis, une, étudiante, de master, informatique],

mais après prétraitement la liste devient V= [étudiante, master, informatique] et cela après élimination des stop-words comme je, suis, une, de.

3.4.3 Schéma de pondération

1. **Calcul du score IG de chaque terme du vocabulaire V**

L'information gain est une méthode supervisée utilisée comme critère de sélection dans l'apprentissage automatique qui quantifie la qualité de la division de l'ensemble de données. [59].

Le score par information gain est une mesure couramment utilisée en apprentissage automatique pour la classification de texte. Il permet de déterminer l'importance de chaque mot ou caractéristique dans la classification des textes en différentes catégories.

Le score par information gain est calculé en utilisant l'entropie et la fréquence des occurrences de chaque mot. Voici les étapes détaillées pour calculer le score par information gain.

- **Calcul de l'entropie de la variable classe**

L'entropie de la catégorie globale représente la quantité d'incertitude dans les données avant d'effectuer la classification. Elle est calculée en utilisant la formule suivante :

$$\mathrm{H}\,(C) = -\sum \mathrm{P}\,(C_j)\,\log 2\,\mathrm{P}\,(C_j) \qquad (3.1)$$

- **Calcul de l'entropie conditionnelle**

Nous définissons également l'entropie conditionnelle d'une variable aléatoire donnée par une autre comme la valeur attendue des entropies des distributions conditionnelles, pondéré selon la variable aléatoire conditionnelle [53].

On peut définir l'entropie conditionnelle $\mathrm{H}(C_j \mid t)$ qui est l'entropie d'une variable aléatoire C conditionnelle à la connaissance d'une autre variable aléatoire t.

La réduction de l'incertitude due à une autre variable aléatoire s'appelle l'information mutuelle [29].

L'entropie de chaque mot représente la quantité d'incertitude dans la catégorie en fonction de la présence ou de l'absence de ce mot [29].

$$\mathrm{H}\,(\mathrm{C}\,/\mathrm{t}) = -\sum \mathrm{P}\,(\mathrm{t})\,\mathrm{H}\,(\mathrm{C_j}\,\mathrm{t}) = -\sum \mathrm{p}(\mathrm{t}, \mathrm{C_j})\,\log\,(\mathrm{C_j}\,/\,\mathrm{t}) \qquad (3.2)$$

- **Calcul de l'information gain pour chaque mot**

L'information gain représente l'importance de chaque mot dans la classification des textes en différentes catégories. On le calcule selon la formule suivante :

$$IG\,(C,t) = H(C) - H(C_j/t)$$

$$= -\sum_{j=1}^{m} P(C_j)\, log\left(P(C_j)\right) + P(t)\sum_{j=1}^{m} P(C_j/t))\, log(P(C_j/t)) +$$

$$P(\bar{t})\sum_{j=1}^{m} P(C_j/\bar{t})\, log(P(C_j/\bar{t})) \qquad (3.3)$$

Où

H(*C)* est l'entropie de la variable classe.

H(C|t) est l'entropie de conditionnelle.

En résumé, le score par information gain est une mesure importante pour la classification de texte car il permet de sélectionner les mots les plus importants pour la classification et d'améliorer les performances du modèle.

L'algorithme suivant montre comment calculer le score IG :

Algorithme 1 : Score IG

Entrées: V = {mot uniques}; C = {c_1, c_2, ...c_n};

Sorties: un dictionnaire contenant le score IG pour chaque terme ;

Pour $c_j \in$ c **Faire**

Pour $t \in$ V **Faire**

Calculer :

H (C) ← - ∑ P (C_j) log2 P (C_j) # l'entropie de la variable classe

Calculer:

H (C| t) ← -∑ P (t) H (C_j t)= - ∑ p(t, C_j) log (C_j / t) # l'entropie conditionnelle

Calculer :

$$H(C_j) - H(C_j/t) \leftarrow -\sum_{j=1}^{m} P(C_j)\, log\left(P(C_j)\right) + P(t)\sum_{j=1}^{m} P(C_j/t))\, log(P(C_j/t)) + P(\bar{t})\sum_{j=1}^{m} P(C_j/\bar{t})\, log(P(C_j/\bar{t})) \quad \#\text{score IG}$$

Fin Pour

Fin Pour

2. Ordonner les termes en ordre décroissant selon le score IG

- Stocker les scores IG et les termes correspondants dans une liste (par exemple, une liste de tuples où chaque tuple contient un terme et son score IG).
- Trier la liste de tuples en ordre décroissant selon les scores IG (c'est-à-dire, les termes avec les scores IG les plus élevés apparaîtront en premier).
- Extraire les termes triés de la liste de tuples et stocker la liste de termes triés.

3. Prendre les termes adjacents pour chaque terme t

Pour chaque terme t du vocabulaire V, prendre n termes à gauche et n terme à droite (la liste S des termes adjacents qui sont éventuellement corrélés avec t ➔ redondants).

Dans ce cas spécifique, lorsque nous ordonnons les termes selon leurs scores d'Information Gain (IG) du plus haut au plus bas, les termes adjacents à un terme donné sont définis comme

les k termes qui se trouvent à sa droite et à sa gauche dans cet ordre. En d'autres termes, si nous prenons un terme t et que nous le plaçons dans une liste ordonnée selon les scores IG décroissants, les termes adjacents à t seront les k termes situés immédiatement avant et les k termes situés immédiatement après t dans cette liste.

Par exemple, supposons que nous ayons une liste de termes ordonnés selon leurs scores IG : [terme 1, terme 2, terme 3, terme, terme 5, terme 6, terme 7]. Si nous choisissons t = terme 3 et fixons k = 2, alors les termes adjacents à terme 3 seront terme 1, terme 2, terme 4 et terme 5. Ces termes sont ceux qui se trouvent respectivement à deux positions avant et après terme 3 dans la liste ordonnée.

Lorsque l'on dit que les termes adjacents ont des scores IG proches, cela signifie que leurs scores sont assez similaires, avec une différence minimale entre eux. Par exemple, si les scores IG des termes adjacents sont 0.8, 0.85 et 0.87, on peut considérer ces scores comme étant proches les uns des autres, car la différence entre eux est relativement petite.

- Cette étape consiste à construire une liste de termes adjacents pour chaque terme du vocabulaire V pour chaque terme t dans V, nous sélectionnons les n termes qui se trouvent immédiatement à gauche et à droite de t dans le vocabulaire V.
- Ces termes adjacents sont collectés dans une liste appelée S.
- Cependant, la liste S peut contenir des termes redondants, c'est-à-dire des termes qui apparaissent dans les contextes de plusieurs termes différents.

L'algorithme suivant, intitulé ‘Extraction du sous ensemble des termes adjacents’, indique comment extraire un sous-ensemble de termes adjacents, où N est le nombre des termes adjacents à droite et à gauche du terme t.

Algorithme 2 : Extraction du sous ensemble des termes adjacents

Entrées : V = {vocabulaire des termes uniques} ; t : terme ;

Sorties : S= {liste des termes adjacent};

S = []

i = indice (t, V) # récupérer l'indice du terme t dans la liste V

Pour j dans [max (0, i-N), i-1] **Faire** # Ajouter les termes adjacents à gauche

$S = S \cup V[j]$

Fin pour

Pour j dans [i+1, min (i+N, len (V)] **Faire** # Ajouter les termes adjacents à droite

$S = S \cup V[j]$

Fin pour

4. Calculer l'information mutuelle (MI) entre chaque terme t et ses termes adjacents

Dans la théorie des probabilités et la théorie de l'information, l'information mutuelle de deux variables aléatoires est une quantité mesurant la dépendance statistique de ces variables [2].

L'information mutuelle est un concept important dans la théorie de l'information de Shannon. C'est une méthode pour détecter le degré de corrélation entre les ensembles d'événements [63].

Pour chaque terme adjacent s dans S (liste appelée S contient les termes adjacentes), nous calculons l'information mutuelle entre le terme t et le terme s, qui mesure la dépendance statistique entre ces deux termes dans le contexte du texte en utilisant la formule suivante :

$$MI\ (t,s)\ =\ log\ \frac{p(t,s)}{p(t)p(s)} \qquad (3.4)$$

5. Calculer le nouveau score en fonction de l'information mutuelle.

Calculer le nouveau score de chaque terme t en fonction de l'information mutuelle avec la variable classe C (calculée en 1), et ces termes adjacents.

- Le nouveau score du terme t est alors calculé en soustrayant la somme de ces informations mutuelles sur tous les termes adjacents s de la valeur d'information gain entre t et la variable classe C.

C'est la formule proposée (le nouveau schéma de pondération).

$$W(t) = IG(t, C) - \sum_s I(t, s) \quad (3.5)$$

Où s est un terme adjacent appartient à S. IG (t, C) est l'information gain entre le terme t et la variable classe C. $\sum_s I(t, s)$ est la somme des informations mutuelles entre le terme t et tous les termes adjacent s dans S.

Le nouveau score W(t) permet de mesurer l'importance du terme t dans le contexte du texte en fonction de sa corrélation avec la variable classe C et les termes adjacents S. Les termes avec un score plus élevé sont considérés comme plus importants et peuvent être utilisés pour la classification de texte.

L'algorithme suivant, montre comment calculer le nouveau score :

Algorithme 3 : nouveau score

Entrées : Score IG : dictionnaire ; Adj : $\{s_1, s_2, \ldots s_n\}$;

Sortie : nouveau score

Pour s ∈ S **Faire**

Pour t ∈ dictionnaire **Faire**

IG (C_j,t) ← $-\sum_{j=1}^{m} P(C_j) \log\left(P(C_j)\right) + P(t) \sum_{j=1}^{m} P(C_j/t)) \log(P(C_j/t)) +$

$P(\bar{t}) \sum_{j=1}^{m} P(C_j/\bar{t}) \log(P(C_j/\bar{t}))$ # Score IG

IM (t, s) ← log2 (P (t, s) / (P(t) * P(s))) # Score en fonction de l'information mutuelle

Calculer

W(t) ← IG(t, C) − $\sum_s$ I(t, s) # Nouveau score

Fin pour

Fin pour

Explication :

- Si t a un grand score IG, et faiblement corrélé avec les termes adjacents, il aura un grand score (parce que $\Sigma I(t;s)$ **est faible**)

- Dans ce cas, le terme est à la fois informatif et indépendant des termes adjacents. Il est considéré comme important pour la tâche de classification ou de prédiction, car il apporte une information précieuse tout en étant peu influencé par le contexte. Par conséquent, il aura généralement un grand score.

- Si t a un petit score IG est fortement corrélé avec les termes adjacents, il aura un petit score (parce que $\Sigma I(t;s)$ **est important**).

- Dans ce cas, le terme est moins informatif et fortement lié aux termes adjacents. Sa contribution à la tâche de classification ou de prédiction est limitée et peut être expliquée par les termes voisins. Par conséquent, il aura tendance à avoir un petit score.

3.5 Conclusion

Dans ce chapitre, nous avons expliqué les étapes détaillées de notre approche qui est un schéma de pondération basé sur la combinaison entre les méthodes de filtrage (IG et MI) pour sélectionner et éliminer cette redondance en mesurant la corrélation entre les attributs qui ont des scores similaires ou proches et obtenir un sous-ensemble optimal pour la réduction des coûts de calcul et améliorer la performance de classification globale.

Chapitre 4 : Implémentation

4.1 Introduction

Notre conception détaillée a été expliquée dans le chapitre précédent qui est une méthode de sélection de caractéristiques pour la classification de texte. Ce chapitre présente la mise en œuvre de la méthode proposée.

Nous décrivons d'abord l'environnement de développement, les différentes bibliothèques pour implémenter la méthode proposée, et une explication de l'application et ses différentes fonctionnalités. Nous présentons dans la suite une description des étapes menée dans le processus d'expérimentation, par la suite nous affichons les résultats obtenus.

4.2 Description des ressources logicielles

4.2.1 Enivrements de développement

Lors de la phase d'implémentation de notre méthode, nous avons utilisé Python -Version 3.8.5 comme langage de programmation, Jupyter Notebook comme IDE.

Python

Python est un langage de programmation puissant et facile à apprendre. Il dispose de structures de données de haut niveau et permet une approche simple mais efficace de la programmation orientée objet. Parce que sa syntaxe est élégante, que son typage est dynamique et qu'il est interprété, Python est un langage idéal pour l'écriture de scripts et le développement rapide d'applications dans de nombreux domaines et sur la plupart des plateformes [64].

Python est un langage qui peut s'utiliser dans de nombreux contextes et s'adapter à tout type d'utilisation grâce à des bibliothèques spécialisées à chaque traitement. Il est cependant particulièrement utilisé comme langage de script pour automatiser des tâches simples mais fastidieuses comme par exemple un script qui récupérerait la météo sur internet ou qui s'intégrerait dans un logiciel de conception assistée par ordinateur afin d'automatiser certains enchaînements d'actions répétitives. On l'utilise également comme langage de développement de prototype lorsqu'on a besoin d'une application fonctionnelle avant de l'optimiser avec un

langage de plus bas niveau. Il est particulièrement répandu dans le monde scientifique, et possède de nombreuses extensions destinées aux applications numériques [65].

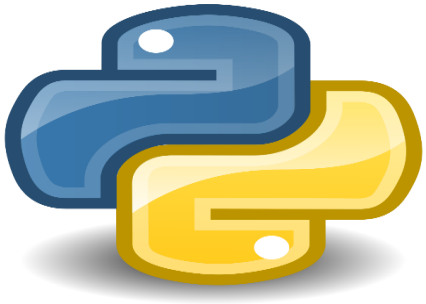


Figure 4. 1 : Logo du langage de programmation Python

Jupyter Notebook

Jupyter Notebook est un outil open source créé à partir de Python en 2014, Jupyter est un notebook de calcul (computational notebook) open source, gratuit et interactif. C'est une application web basée client permettant de créer et de partager du code, des équations, des visualisations ou du texte [66].

En effet, Jupyter Notebook fut créé pour faciliter la présentation de travaux en programmation et pour permettre le codage collaboratif. Il permet de combiner le code, les commentaires, le contenu multimédia et les visualisations dans un document interactif : le notebook. Celui-ci peut être partagé, réutilisé et modifié [66].

Figure 4. 2 : Logo du Jupyter Notebook.

4.2.2 Les bibliothèques nécessaires

Scikit-learn : encore appelé *sklearn*, est la bibliothèque la plus puissante et la plus robuste pour la machine learning en Python. Elle fournit une sélection d'outils efficaces pour l'apprentissage automatique et la modélisation statistique, notamment la classification, la régression et le clustering via une interface cohérente en Python [67].

Matplotlib : est une bibliothèque de traçage disponible pour le langage de programmation Python en tant que composant de NumPy, une ressource de traitement numérique de Big Data. Matplotlib utilise une API orientée objet pour intégrer des tracés dans des applications Python [68].

Pandas : La bibliothèque logicielle open-source Pandas est spécifiquement conçue pour la manipulation et l'analyse de données en langage Python. Elle est à la fois performante, flexible et simple d'utilisation. Grâce à Pandas, le langage Python permet enfin de charger**,** d'aligner, de manipuler ou encore de fusionner des données. Les performances sont particulièrement impressionnantes quand le code source back-end est écrit en C ou en Python. [69]

Natural Language Toolkit (NLTK) : est une plate-forme utilisée pour créer des programmes Python qui fonctionne avec des données en langage humain pour les appliquer au traitement statistique du langage naturel (NLP) [70].

Numpy : Numpy est une bibliothèque pour le langage de programmation Python qui permet plus de stockage de données avec moins de mémoire. Avec un tableau multidimensionnel et d'autres ressources, NumPy permet aux programmeurs Python de stocker efficacement les nombres [72].

4.3 Démarche expérimental

4.3.1 Présentation des Datasets

Pour évaluer notre méthode, nous utilisons des ensembles de données de tailles et de complexité variables. Les deux ensembles de données que nous utilisons dans cette section sont décrits comme suit :

Fake News

Fake News contient deux types d'articles FAKE (12,600 articles) et REAL (12,600 articles). Cet ensemble de données a été collecté à partir de sources du monde réel ; les articles véridiques

ont été obtenus en explorant des articles de Reuters.com (site Web d'actualités) [17]. La figure 4.3 illustre une partie du dataset Fake News [73].

	text	category
0	Says the Annies List political group supports ...	1
1	When did the decline of coal start? It started...	0
2	Hillary Clinton agrees with John McCain "by vo...	0
3	Health care reform legislation is likely to ma...	1
4	The economic turnaround started at the end of ...	0
...	...	...
10235	There are a larger number of shark attacks in ...	0
10236	Democrats have now become the party of the [At...	0
10237	Says an alternative to Social Security that op...	0
10238	On lifting the U.S. Cuban embargo and allowing...	1
10239	The Department of Veterans Affairs has a manua...	1

10240 rows × 2 columns

Figure 4. 3 : Le dataset Fake News

Hotel Reviews

Il s'agit d'une liste de commentaires fournis par la base de données commerciale de Datafiniti. L'ensemble des données comprend l'emplacement de l'hôtel, le nom, la note, les données d'évaluation, le titre, le nom d'utilisateur et plus encore [74].

	Unnamed: 0	User_ID	text	Browser_Used	Device_Used	category
0	0	id47447	nice hotel check good sleep comfortable good q...	InternetExplorer	Desktop	happy
1	1	id27844	many time business want family friendly hotel ...	Google Chrome	Mobile	happy
2	2	id35994	room small share brother parent room tell woul...	Google Chrome	Tablet	happy
3	3	id39336	hotel clean plenty space staff helpful plan la...	Internet Explorer	Mobile	happy
4	4	id15741	stay roger night business trip november really...	IE	Mobile	happy
...	...	...	...	...	...	...
995	995	id47245	give room reserve sleep floor continental brea...	Chrome	Tablet	not happy
996	996	id29202	never expereinced pathetic disgust hotel life ...	Firefox	Tablet	not happy
997	997	id21682	book hotel request double check double queen r...	Chrome	Tablet	not happy
998	998	id31943	stay hotel hotel ever stay arrive later night ...	IE	Mobile	not happy
999	999	id20766	write review save pick hotel deserve star rati...	Firefox	Mobile	not happy

1000 rows × 6 columns

Figure 4. 4 : Le dataset Hotel Reviews.

Prétraitement des données :

Comme mentionné précédemment dans le chapitre 3, le prétraitement des ensembles de données implique plusieurs étapes:

1. Convertir tous les textes en minuscules.
2. Lemmatization
3. Suppression des html tags.
4. Suppression des chiffres
5. Suppression des caractères spéciaux
6. Suppression des mots de moins de 4 caractères.
7. Suppression des espaces entre les mots.
8. Suppression des espaces gauche et droit.
9. Suppression des mots vides.

	category	text
0	0	aarp large reseller insurance country vested i...
1	0	high corporate tax rate world right
2	0	rhode island parole board never receive object...
3	0	approval united state point high hillary clint...
4	0	social security trust fund sound without anyth...
...	...	...
995	1	homosexual behavior cut lifeby year
996	1	recent house special election florida democrat...
997	1	take credit rein state spending governor
998	1	maternal mortality rate woman increase
999	1	say get cons successfully return society impor...

1000 rows × 2 columns

Figure 4. 5 : Fake News après prétraitement.

	Unnamed: 0.1	Unnamed: 0	text	category
0	0	0	room kind clean strong smell dogs generally av...	0
1	1	1	stayed crown plaza april april staff friendly ...	0
2	2	2	booked hotel hotwire lowest price could find g...	0
3	3	3	stayed husband sons way alaska cruise loved ho...	1
4	4	4	girlfriends stayed celebrate th birthdays plan...	0
...	...	...	...	...
38927	38927	38927	arrived late night walked check area completel...	1
38928	38928	38928	positive impression location public parking op...	0
38929	38929	38929	traveling friends shopping show location great...	0
38930	38930	38930	experience ok paid extra view pool got view pa...	0
38931	38931	38931	westin wonderfully restored grande dame hotel ...	1

38932 rows × 4 columns

Figure 4. 6 : Hotel Reviews après prétraitement.

4.3.2 Implémentation de notre méthode

Nous avons implémenté notre méthode et comme décrit dans le chapitre précédent : nous avons fait une combinaison entre IG et MI. En premier lieu nous calculons les scores IG des termes et les trier en ordre décroissant.

Les figures ci-dessous montrent les résultats obtenus des datasets sur les scores affectés par IG.

{'say': 0.006568608636105755, 'top': 0.005609915553914657, 'high': 0.0050308164247800935, 'police': 0.00487670317723099, 'since': 0.004636541111428971, 'away': 0.0043145729355384965, 'across': 0.004176991845240341, 'marijuana': 0.004176991845240341, 'still': 0.004176991845240341, 'within': 0.004176991845240341, 'state': 0.004121348696765836, 'president': 0.00403423015692006, 'voter': 0.003761035215459674, 'job': 0.0037019360521728384, 'year': 0.0036847970985343093, 'gun': 0.003635132418700837, 'bottom': 0.003478298769742927, 'colorado': 0.003478298769742927, 'head': 0.003478298769742927, 'millionaire': 0.003478298769742927, 'option': 0.003478298769742927, 'black': 0.0034702585227174287, 'line': 0.0034702585227174287, 'law': 0.00325715574533314, 'rhode': 0.0031204524585497495, 'barack': 0.0028652833870783656, 'month': 0.002783910994676564, 'abolish': 0.002780620872345429 5, 'defend': 0.002780620872345429 5, 'directly': 0.002780620872345429 5, 'idea': 0.002780620872345429 5, 'invest': 0.002780620872345429 5, 'loan': 0.002780620872345429 5, 'name': 0.002780620872345429 5, 'north': 0.002780620872345429 5, 'opponent': 0.002780620872345429 5, 'terrorist': 0.002780620872345429 5, 'totally': 0.002780620872345429 5, 'water': 0.002780620872345429 5, 'wealthy': 0.002780620872345429 5, 'accept': 0.002774596755717784, 'equal': 0.002774596755717784, 'roughly': 0.002774596755717784, 'plan': 0.0027623106315169954, 'system': 0.0026838341267009236, 'student': 0.0026672620124249535, 'border': 0.002645367455087921 4, 'care': 0.0026413421191417985, 'average': 0.0025531863708591063, 'gov': 0.0025495952945896 46, 'control': 0.002549161426354085, 'domestic': 0.002549161426354085, 'teacher': 0.002549161426354085, 'get': 0.002544206125263737, 'violence': 0.0024509725781383107, 'fund': 0.0024383420057871863, 'clinton': 0.002380174076990871, 'island': 0.002275556306346904, 'abortion': 0.0022106786792536193, 'lose': 0.002183296836526516, 'rate': 0.002163830930030919, 'receive': 0.002161251180732515, 'return': 0.002161251180732515, 'health': 0.0021249167180796835, 'work': 0.0021148362716575386, 'new': 0.0020841634802236664, 'wage': 0.002084010076661702, 'adopt': 0.0020839550891128544, 'agency': 0.0020839550891128544, 'agenda': 0.0020839550891128544, 'alien': 0.0020839550891128544, 'allen': 0.0020839550891128544, 'appoint': 0.0020839550891128544, 'approximately': 0.00208395

Figure 4. 7 : Un aperçu sur les scores IG de chaque terme de Fake News.

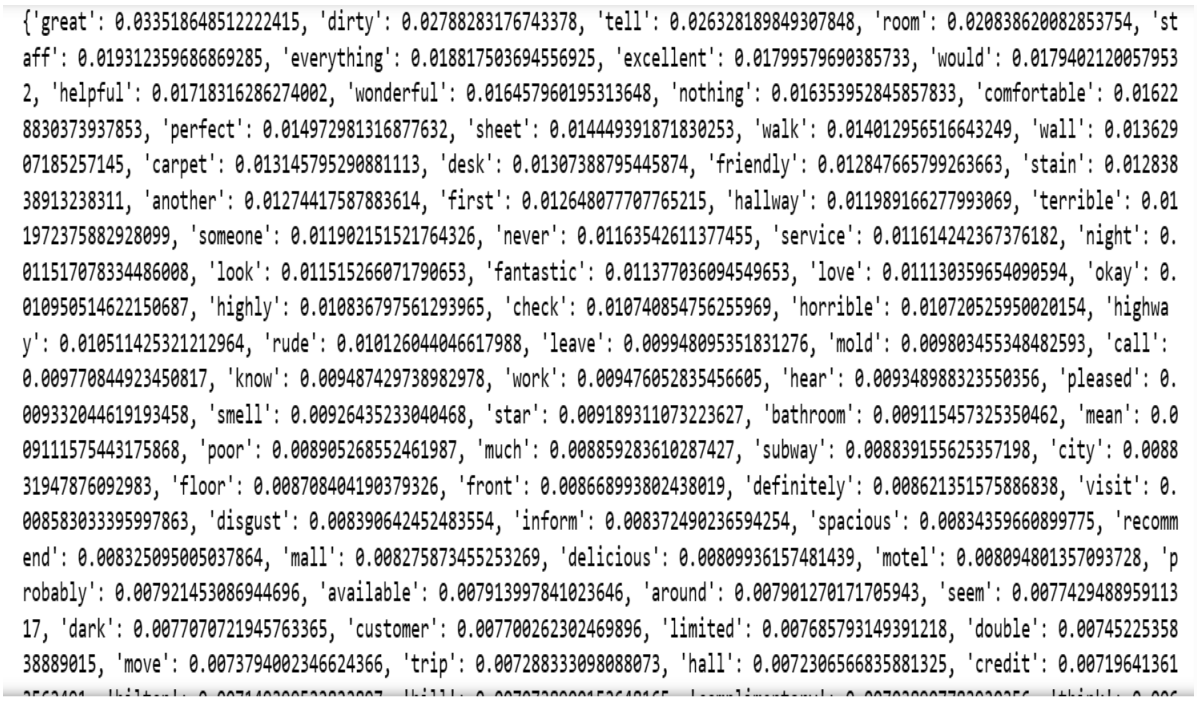


Figure 4. 8 : Un aperçu sur les scores IG de chaque terme de Hotel Reviews.

Après nous avons sélectionné le meilleur sous-ensemble des features qui ont le meilleur score du accuracy, et qui est généré par IG.

Les figures ci-dessous montrent les résultats obtenus des datasets sur les scores affectés par IG.

	Meilleur sous-ensemble	Nb termes	Score (Acc)
0	say - top - high - police - since - away - acr...	454	0.764

Figure 4. 9 : Un aperçu sur meilleur sous-ensemble de Fake News.

	Meilleur sous-ensemble	Nb termes	Score (Acc)
0	great - dirty - tell - room - staff - everythi...	797	0.852

Figure 4. 10: Un aperçu sur meilleur sous-ensemble de Hotel Reviews.

Ensuite, nous cherchons les mots adjacents corrélés (les mots qui ont des scores proches) et éliminer les termes redondants

Les figures ci-dessous montrent un aperçu sur sous ensemble optimale sélectionné.

	Meilleur sous-ensemble	Nb termes	Score (Acc)
0	say - marijuana - still - across - police - aw...	398	0.796

Figure 4. 11: Un aperçu sur ensemble optimale sélectionné de Fake News.

	Meilleur sous-ensemble	Nb termes	Score (Acc)
0	dirty - great - golf - joke - blood - unfriend...	664	0.856

Figure 4. 12: Un aperçu sur ensemble optimale sélectionné de Hotel Reviews.

4.3.3 Classification

Afin d'évaluer l'efficacité de la méthode de sélection des features proposée, il est essentiel de procéder à l'entraînement et au test des algorithmes de classification sur divers ensembles de données. Pour cela, nous utiliserons les métriques les plus couramment utilisées, à savoir l'IGI, le PMI et le CH2, ainsi que les algorithmes couramment utilisés dans la classification de textes : le Support Vector Machine (SVM), Naive Bayes (NB) et la régression logistique (LR). Ces algorithmes sont largement reconnus pour obtenir de bons résultats dans cette tâche. Comme indiqué précédemment dans le deuxième chapitre, plusieurs métriques de sélection des caractéristiques sont disponibles. Pour notre étude, nous avons choisi d'utiliser les métriques suivantes : l'indice de Gini (IGI), l'information mutuelle (PMI) et le test du chi-carré (CH2). Nous avons ensuite appliqué ces métriques à la classification en utilisant les mêmes classifieurs, à savoir le Naive Bayes (NB), le Support Vector Machine (SVM) et la régression logistique (LR), en fonction des résultats d'exactitude (accuracy). Les résultats de classification obtenus sont présentés dans les tableaux suivants. Les tableaux suivants présentent une évaluation de l'efficacité de la méthode proposée par rapport aux autres métriques de sélection. Les résultats des tableaux permettent de comparer les performances de notre méthode par rapport aux autres métriques.

DATASET	*Fake News*		
Classifieur / algorithmes	***NB***	***SVM***	***LR***
IGI	0.568000	0.568000	0.572000
PMI	0.568000	0.568000	0.572000
CH2	0.568000	0.568000	0.572000
Méthode proposée	**0.796000**	**0.792000**	**0.768000**

Tableau 4. 1 : Résultats de classification des méthodes de comparaison et la méthode proposée pour le dataset Fake News.

DATASET	*Hotel Reviews*		
Classifieur / algorithmes	***NB***	***SVM***	***LR***
IGI	0.804000	0.812000	0.804000
PMI	0.804000	0.812000	0.804000
CH2	0.804000	0.812000	0.804000
Méthode proposée	**0.876000**	**0.884000**	**0.844000**

Tableau 4. 2: Résultats de classification des méthodes de comparaison pour le dataset Hotel Reviews.

Selon les résultats des tableaux, il est évident que notre approche proposée surpasse nettement les métriques IGI, PMI, CH2. Les résultats obtenus démontrent clairement la performance supérieure de notre méthode en termes de sélection des features pertinentes dans les deux dataset utilisés. Il est clairement démontré par les résultats obtenus que notre méthode proposée est hautement efficace.

Nous avons réalisé une classification pour chaque ensemble de données en utilisant les algorithmes SVM, NB et LR. Pour cela, nous avons utilisé une approche de seuillage où nous avons ajouté 100 nouvelles features à l'ensemble des caractéristiques déjà sélectionnées à chaque itération. Le nombre de features (termes) qui correspond au meilleur score Accuracy est montré dans les tableaux suivants :

DATASET	***Fake News***					
Classifieur / algorithmes	***NB***		***SVM***		***LR***	
	Nb des features	***Acc score***	***Nb des features***	***Acc score***	***Nb des features***	***Acc score***
IGI	2900	0.568000	800	0.648000	800	0.648000
PMI	800	0.600000	400	0.596000	2800	0.588000
CH2	500	0.784000	400	0.776000	500	0.740000
Méthode proposée	**1600**	**0.800000**	**1400**	**0.792000**	**1700**	**0.772000**

Tableau 4. 3: Résultats de classification montrant les nombres des features des méthodes de comparaison et la méthode proposée pour le dataset Fake News.

DATASET	***Hotel Reviews***					
Classifieur / algorithmes	***NB***		***SVM***		***LR***	
	Nb des features	***Acc score***	***Nb des features***	***Acc score***	***Nb des features***	***Acc score***
IGI	6600	0.808000	6500	0.816000	6600	0.804000
PMI	500	0.812000	6100	0.812000	6700	0.804000
CH2	500	0.860000	900	0.860000	600	0.848000
Méthode proposée	**3200**	**0.884000**	**3000**	**0.888000**	**3100**	**0.852000**

Tableau 4. 4 : Résultats de classification montrant les nombres des features des méthodes de comparaison et la méthode proposée pour le dataset Hotel Reviews.

Notre méthode se distingue par une sélection optimale des sous-ensembles de caractéristiques pertinentes, comme en témoigne le nombre de caractéristiques sélectionnées. Il est clair que notre méthode surpasse les autres méthodes dans l'identification des caractéristiques les plus pertinentes pour la tâche de classification.

Les figures ci-dessous montrent les meilleurs scores obtenus pour chaque dataset :

- ***Dataset Fake News***

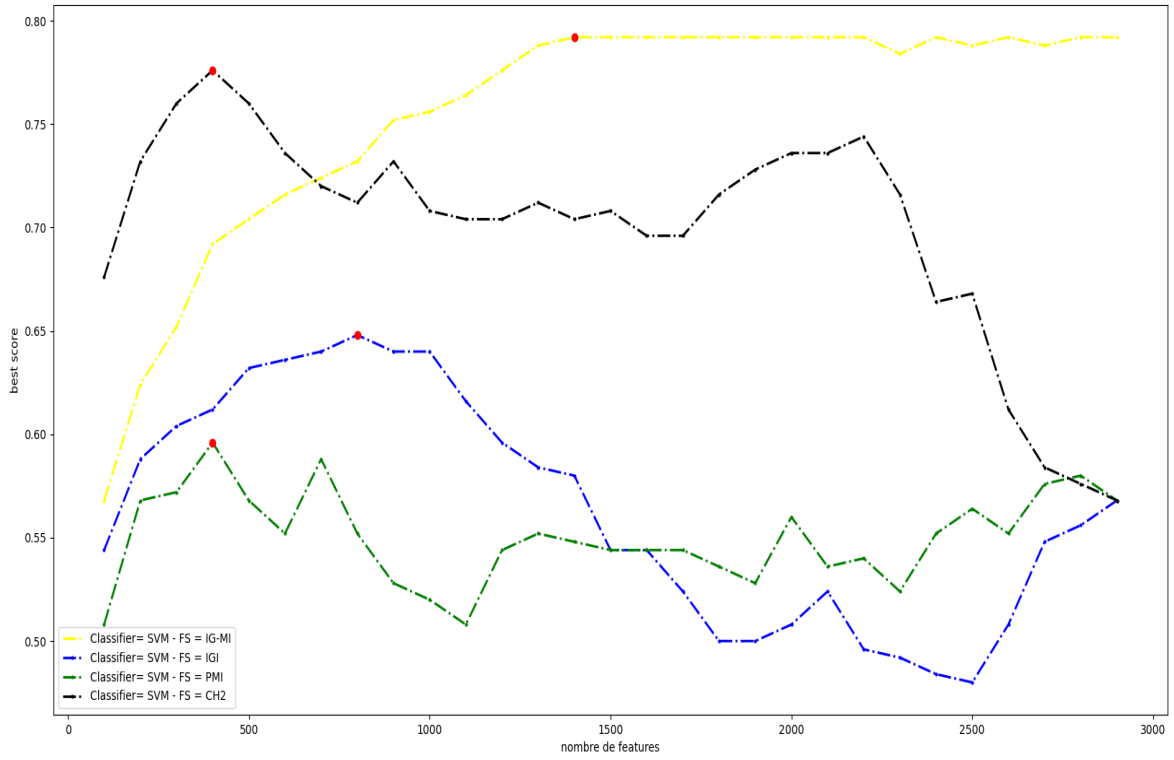


Figure 4. 13 : Accurcy Score par le classifier SVM de Fake News.

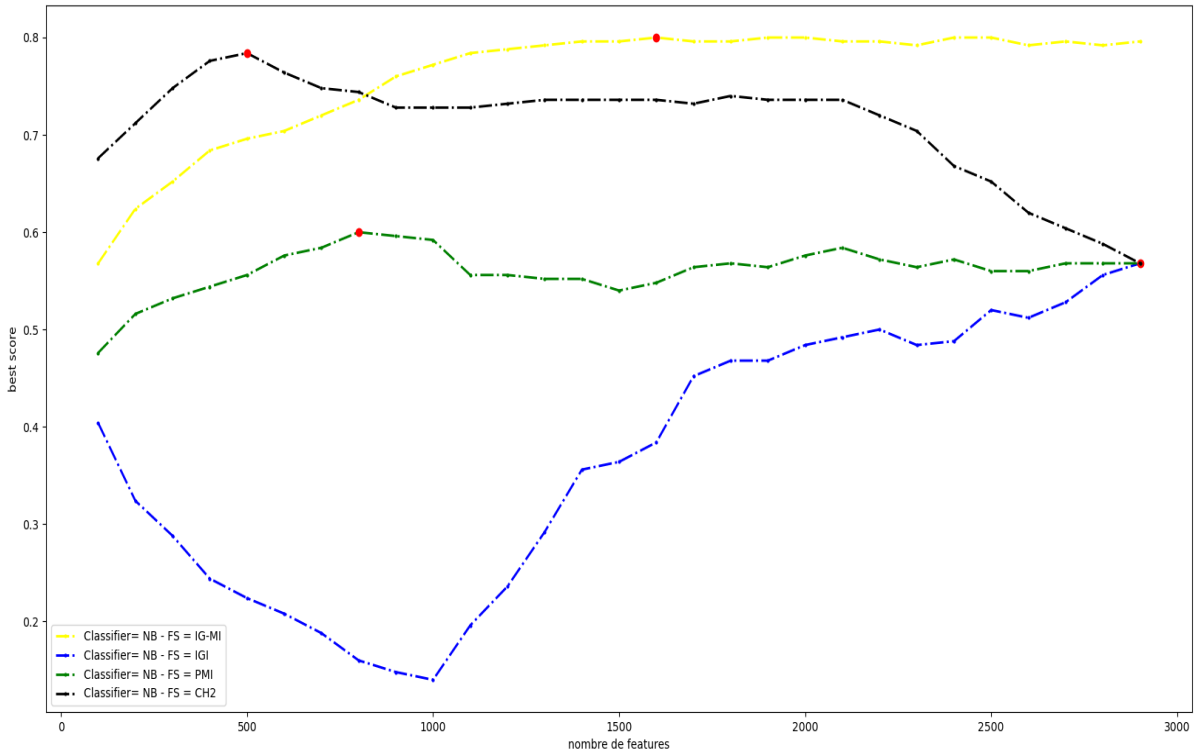


Figure 4. 14: Accurcy Score par le classifier NB de Fake News.

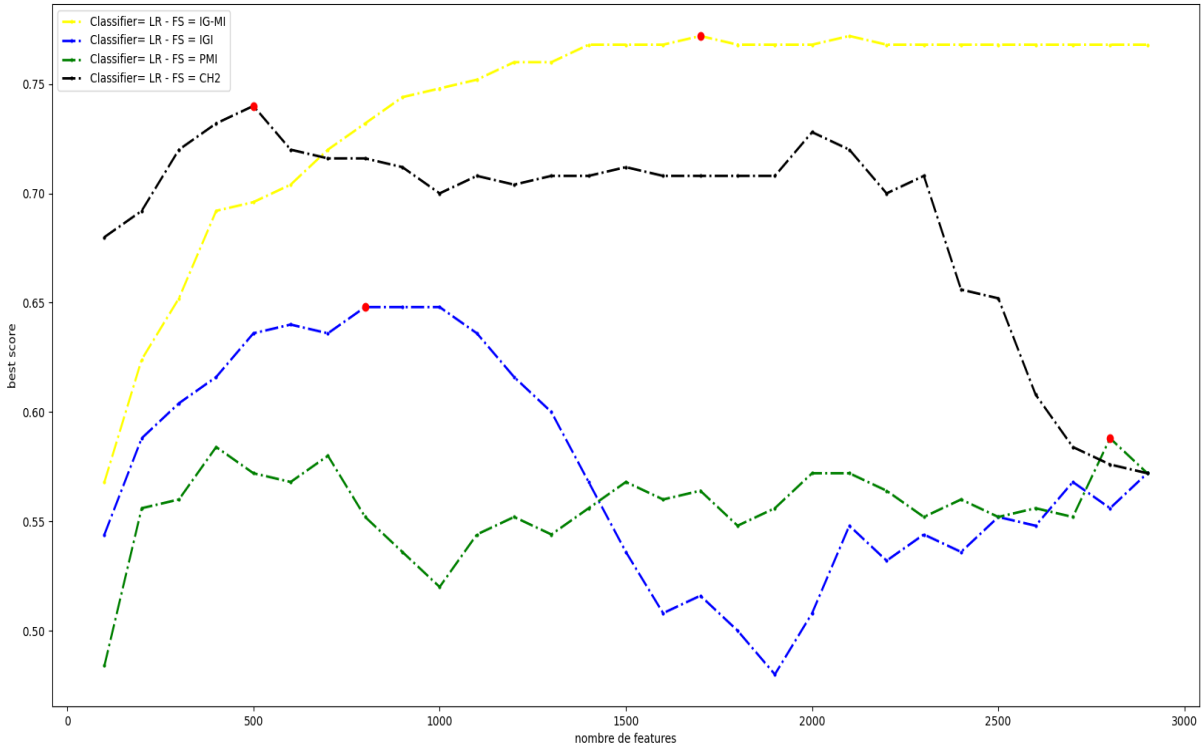


Figure 4. 15 : Accurcy Score par le classifier LR de Fake News.

➢ Dataset Hotel Reviews

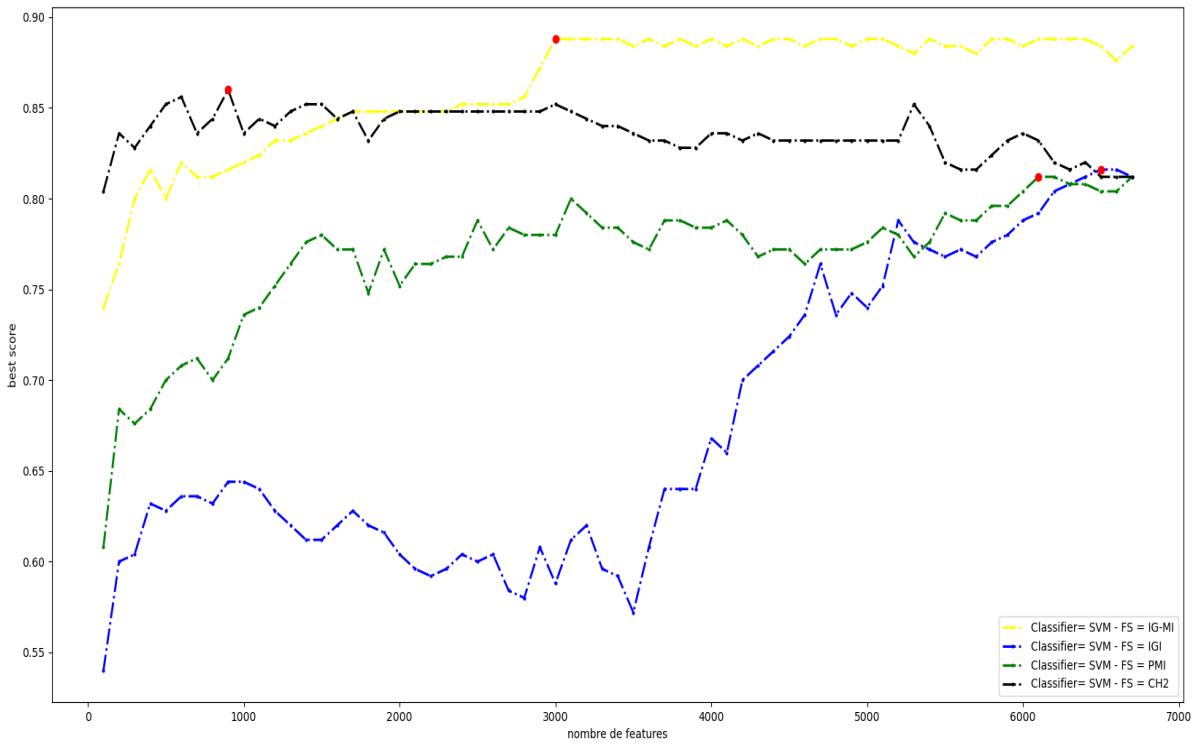


Figure 4. 16 : Accurcy Score par le classifier SVM de Hotel Reviews.

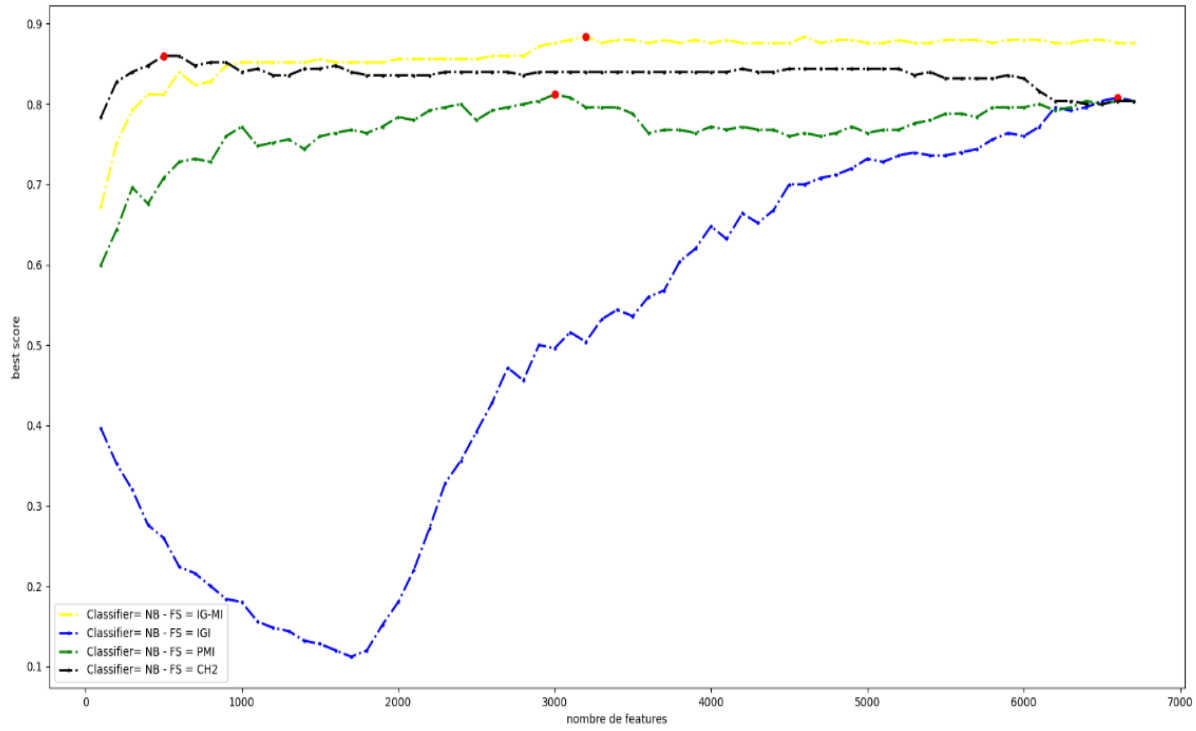


Figure 4. 17: Accurcy Score par le classifier NB de Hotel Reviews .

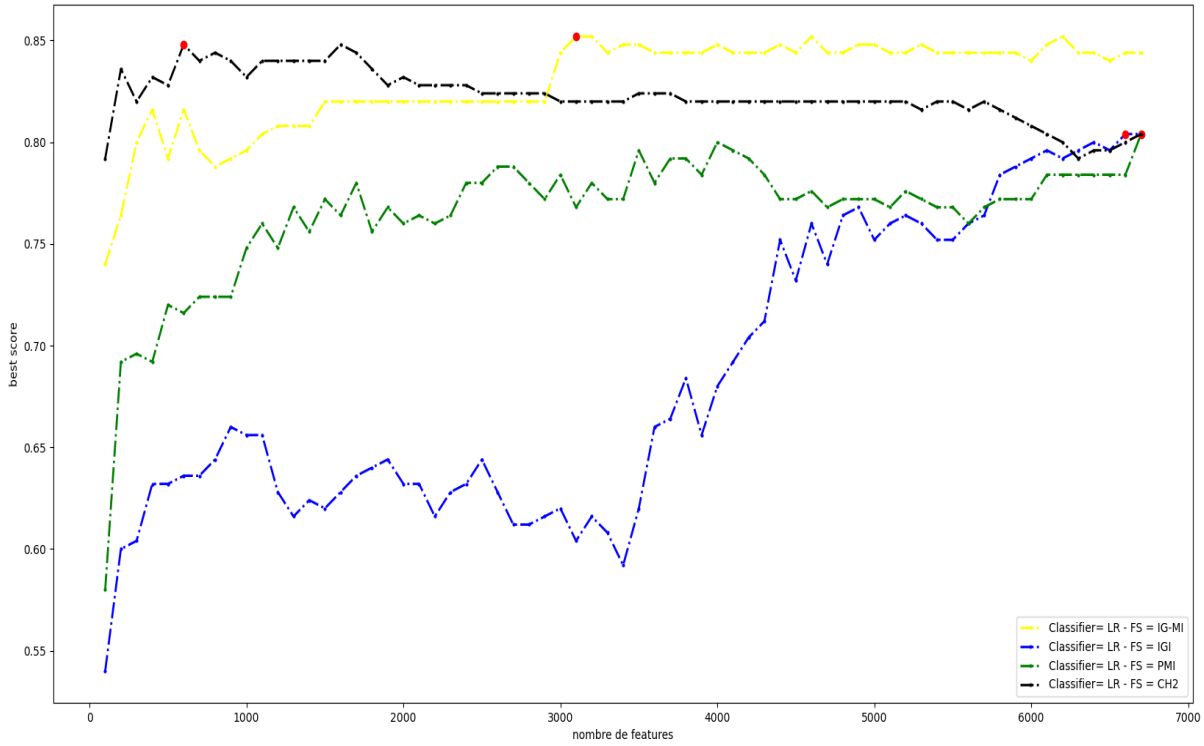


Figure 4. 18: Accurcy Score par le classifier LR de Hotel Reviews.

Les expérimentations ont clairement démontré l'efficacité de notre méthode proposée en termes d'accuracy obtenue et du nombre de caractéristiques sélectionnées. Nos résultats surpassent ceux des méthodes de sélection de données catégorielles les plus répandues, à savoir l'IGI, le PMI et le CH2.

Lorsqu'on évalue un terme à l'aide de méthodes statistiques telles que l'IGI, PMI et CH2), en se basant uniquement sur le nombre de documents dans lesquels le terme apparaît, il existe un risque de sélectionner des termes redondants. Ces approches ne prennent pas en compte le fait que certains termes peuvent être biaisés ou ne pas apporter d'informations distinctives significatives.

La suppression des features redondantes et répétitives a également eu un impact positif sur l'efficacité du processus de classification. En réduisant le nombre de caractéristiques, nous avons pu accélérer le temps d'entraînement du modèle et améliorer l'efficacité des calculs comme nous utilisons dans notre méthode.

4.4 Conclusion

Dans ce chapitre, nous avons mis en place la méthode proposée qui réponde parfaitement aux objectifs fixés au départ. Notre méthode de sélection de caractéristiques proposée est un schéma

de pondération basé sur une combinaison de méthodes de filtrage (information gain et information mutuel) pour sélectionner les features informatifs et éliminer ceux qui sont redondant, en se basant sur la mesure de corrélation entre les attributs. Les résultats montrent que notre méthode a prouvé son efficacité pour éliminer les attributs avec des scores similaires ou proches et obtenir le meilleur sous-ensemble pour réduire les coûts de calcul et augmenter la performance du classificateur.

Conclusion Générale

La classification de texte constitue une tâche du domaine du traitement automatique des langages naturels qui vise à classer de manière automatique des ressources documentaires, souvent issues d'un corpus donné. La classification de texte reste un domaine de recherche dynamique et en évolution, avec de nombreuses opportunités d'amélioration et d'innovation.

Au cours de la dernière décennie, la sélection d'attributs est devenue un sujet de recherche extrêmement actif dans les domaines de l'apprentissage automatique. La sélection d'attributs permet de simplifier et d'accélérer l'apprentissage automatique en réduisant la complexité des données. Elle contribue également à améliorer la généralisation des modèles en réduisant le sur-apprentissage (overfitting) et en améliorant l'interprétabilité des résultats. La sélection d'attributs peut être effectuée selon différentes approches telles que les méthodes de filtrage, les méthodes Wrapper et les méthodes Embedded.

De nombreuses métriques de sélection, telles que l'IG (Information Gain), le MI (Mutual Information), le Chi2 (Chi-Square), le DF (Document Frequency), etc., ont démontré leur efficacité dans la sélection de caractéristiques. Cependant, ces métriques ne prennent pas en compte la corrélation entre les mots ni leur fréquence de cooccurrence dans le même contexte.

Nous avons développé une approche qui combine les méthodes de filtrage IG (Information Gain) et MI (Mutual Information), pour sélectionner et éliminer les attributs non pertinents ou redondants. Cette approche nous permet d'obtenir un sous-ensemble optimal d'attributs, ce qui contribue à réduire les coûts de calcul associés au traitement des données.

En éliminant les attributs non pertinents ou redondants, nous avons réduit le bruit dans les données et permis aux algorithmes de classification de se concentrer sur les attributs les plus informatifs. Cela a conduit à une meilleure précision et à une meilleure capacité de généralisation des modèles de classification.

Nous avons impliqué l'utilisation de trois classifieurs (NB, SVM et LR) sur deux datasets (Fake News et Hotel Reviews), qui ont clairement démontré l'efficacité de notre méthode proposée de sélection des features. En comparant les résultats obtenus avec ceux des métriques les plus populaires, à savoir l'IGI, le MI, et le Ch2, notre méthode s'est avérée supérieure dans la plupart des cas.

Bibliographiques

[1] Billal Belainine «**Classification Supervisée de textes courts et bruités application au domaine des médias sociaux**» thés de document université Québec à montréal, avril 2017

[2] Siafa Aya «**Un schéma de pondération et de sélection des termes pertinents basé sur la distribution des fréquences des documents et des termes entre catégories** » Mémoire de Fin d'études Master, Juin 2022.

[3] M. F. Porter, «**An algorithm for suffix stripping** », Program, pp 130–137, Morgan Kaufmann Publishers Inc, 1980 .

[4] Guillaume Cleuziou «**Une méthode de classification non-supervisée pour l'apprentissage de règles et la recherche d'information**».

[5] https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501309-apprentissage-non-supervise/

[6] S.ABDELOUAHAB, «**Processus de classification supervisée de textes arabes par la méthode K PPV Application aux articles de presse**», Mémoire de Master, Université de M'sila, 2011-2012.

[7] LAHLOU OUCHIHA, « **CLASSIFICATION SUPERVISÉE DE DOCUMENTS ÉTUDE COMPARATIVE »,** UNIVERSITÉ DU QUÉBEC EN OUTAOUAIS, JANVIER 2016.

[8] https://www.researchgate.net/figure/schema-de-la-classification-bayesienne_fig2_330970221.

[9] https://www.xlstat.com/fr/solutions/fonctionnalites/classifieur-bayesien-naif

[10] https://penseeartificielle.fr/tout-pour-bien-debuter-en-machine-learning-4/hyperplan-svm/

[11] https://fr.slideshare.net/AmraneAlik/the-naive-bayesien-classifier

[12] M. Zaiz Faouzi, 15 juillet 2010, « **Les Supports Vecteurs Machines (SVM) pour la reconnaissance des caractères manuscrits arabes**», Thèse Pour obtenir le diplôme de magister, Université Mohamed Khider – BISKRA.

[13] https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

[14] https://www.anakeyn.com/2019/11/28/classification-de-pages-web-via-deep-learning-reseau-de-neurones-a-propagation-avant/.

[15] Ali LABIAD. « **Sélection des mots clés basée sur la classification et l'extraction des règles d'association** ». Thèse de doct. Université du Québec à Trois-Rivières, 2017.

[17] https://www.nikodez.fr/single-post/2018/07/02/For%C3%AAts-Al%C3%A9atoires

[18] https://dataanalyticspost.com/Lexique/random-forest/

[21] Ferdi Dounya « **Sélection des caractéristiques basée sur le plongement lexical pour la classification des textes** » Mémoire de projet de fin d'étude Master, Septembre 2021.

[23] https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

[24] Hassan CHOUAIB « **Sélection de caractéristiques: méthodes et applications** » ,8 juillet 2011.

[25] https://www.jedha.co/formation-ia/algorithme-knn-apprentissage-supervise

[27] https://www.ibm.com/fr-fr/topics/logistic-regression consulté le 05/06/2023.

[28] https://datascience.eu/fr/apprentissage-automatique/matrice-de-confusion/

[29] Bensaada Aridje « **Sélection des termes co-occurrents avec entropie minimale pour la Classification des textes** » Mémoire de Fin d'études Master, Juin 2022.

[31] https://www.techopedia.com/definition/34788/bag-of-words-bow

[32] https://aiml.com/what-are-the-advantages-and-disadvantages-of-bag-of-words-model/

[34] Suhang Wang1, Jiliang Tang2, and Huan Liu1 1Arizona State University, Tempe, AZ, USA 2Michigan State University, East Lansing, MI,USA.

[36] Dash, M. et Liu, H. « **Feature selection for classification** », Intelligent Data Analysis, Volume 1(1-4), pp-pp 131–156 (1997).

[38] Nicole CHALLITA « **Contributions à la sélection des attributs de signaux non stationnaires pour la classification** » le 28 avril 2018.

[39] OUALI Choayb « **Classification automatique de textes** » 2013 /2014

[40] Melle HAFA MÉMOIRE DE FIN D'ÉTUDE pour obtenir le grade de Master en Informatique Spécialité : Modèle Intelligent et Décision (M.I.D) présenté et soutenu publiquement 01 Juillet 2012.

[41] Vipin Kumar et Sonajharia Minz « **Sélection des fonctionnalités revue de littérature** », École des sciences informatiques et des systèmes, Université Jawaharlal Nehru / New Delhi, juin 2014.

[44] D.Alamedine, « **Selection of EHG paramètre characteristics for the classification of utérine contractions** », Université de Technologie de Compiègne, 2015.

[45] N. Zhang, « **Feature selection based segmentation of multi-source images : application to brain tumor segmentation in multi-sequence MRI** » INSA de Lyon, 2011.

[46] N. Zhang, « **Feature selection based segmentation of multi-source images : application to brain tumor segmentation in multi-sequence MRI** » INSA de Lyon, 2011.

[47] J. Yang, Y. Liu, X. Zhu, et al, **« A new feature selection based on comprehensive measurement both in inter-category and intra-category for text categorization** », Information Processing & Management, Volume 48-4 , pp-pp 741-754 , 2012.

[49] https://www.analyticssteps.com/blogs/what-gini-index-and-information-gain-decision-trees.

[50] Heum Park, Soonho,Hyuk-Chul Kwon « **Complete Gini-Index Text (GIT) Feature-Selection Algorithm for Text Classification** » of Computer Science Pusan National University Busan Dept, 23-25 June 2010.

[51] Ameni Bouaziz « **Méthodes d'apprentissage interactif pour la classification des messages courts** », le 19/06/2017.

[54] https://www.talend.com/fr/resources/guide-redondance-donnees/, consulté le 2/02/23.

[55] https://pastel.archives-ouvertes.fr/pastel-00834272/document/ , consulté le 2/02/23.

[56] Mouhoub BELAZZOUG « **Apprentissage statistique pour l'extraction des relations à partir de textes** » Pour l'Obtention du Diplôme de DOCTORAT EN SCIENCES.

[57] MEKHOUKH Wafa, TEHAMI Noura **« Prédiction des protestations publiques à l'aide d'algorithmes de classification** », 2019-2020.

[58] Yasmine YOUSFI, Yasmine BELLAHOUES «**Sélection De Caractéristiques Pour La Classification De Polarité D'Opinion**», Mémoire De Fin D´étude De Master Professionnel.

[59] Jean-Charles RISCH, **«Enrichissement des Modèles de Classification de Textes Représentés par des Concepts** » Charles RISCH, Pour obtenir le grade de DOCTEUR DE L'UNIVERSITÉ DE REIMS CHAMPAGNE-ARDENNE, 27 Juin 2017.

[63] Hongyi GAO, Xi Zeng, Chunhua Yao« **Application of improved distributed naive Bayesian algorithms in text classification**», 2019.

[64] https://docs.python.org/fr/3/tutorial/ , consulté le 10/06/2023.

[65] https://www.techno-science.net/glossaire-definition/Python-langage.html, consulté le 10/06/2023.

[66] https://www.lebigdata.fr/jupyter-notebook , consulté le 10/06/2023.

[67] https://www.data-transitionnumerique.com/scikit-learn-python/, consulté le 10/06/2023.

[68] https://fr.theastrologypage.com/matplotlib , consulté le 10/06/2023.

[69] https://datascientest.com/pandas-python-data-science consulté le 10/06/2023.

[70] https://fr.theastrologypage.com/natural-language-toolkit, consulté le 10/06/2023.

[72] https://fr.theastrologypage.com/numpy,consulté le 10/06/2023.

[73] https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset, consulté le 10/06/2023.

[74] https://www.kaggle.com/datasets/harmanpreet93/hotelreviews, consulté le 10/06/2023.