الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université de 8 Mai 1945 – Guelma -

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

Département d'Informatique

Mémoire de Fin d'études Master

Filière : Informatique

Option : Systèmes Informatiques

Thème :

Sélection des termes co-occurrents avec entropie minimale pour la Classification des textes

---

Encadré par :

Dr. Farek Lazhar

Présenté par :

Bensaada Aridje

Juin 2022

RÉSUMÉ

La sélection de features (attributs) en tant que technique de réduction de la dimensionnalité, vise à choisir un petit sous-ensemble des features pertinents parmi ceux d'origine en supprimant les non pertinents, redondants ou bruyants. La sélection des featuresconduit généralement à de meilleures performances d'apprentissage, c'est-à-dire une plus grande précision d'apprentissage, un coût de calcul inférieur et une meilleure interprétation du modèle. Les méthodes de sélection de features telles que Information Gain (IG), Mutual Information (MI) et Chi-square (Chi2) sont des méthodes statistiques basées sur la fréquence des documents (en, Document-Frequency), mais elle et ne prennent pas en considération la fréquence des termes à l'intérieur des documents, ni considérer leur sémantique.

En se basant sur l'idée que les termes qui coexistent fréquemment peuvent avoir une sémantique commune d'où une capacité de discrimination élevée par rapport aux termes isolés, nous proposons une méthode de sélection des features pour la classification des textes en considérant deux mesures : la fréquence de co-occurrence des termes et leur entropie où un terme qui coexiste fréquemment avec d'autre termes et conduit à minimiser l'incertitude (l'entropie) de la variable classe est considéré comme pertinent.

La performance de notre méthode est comparée aux quatre métriques de sélection les plus couramment utilisées : Information Gain (IG), Mutual Information (MI), Chi-square (Chi2) et Document-Frequency (DF), en utilisant deux classifieursNaïve Bayes (NB) et Support Vector Machine (SVM) et trois datatsets.

Mots clés : sélection , terme , co-occurence , entropie , texte , classification .

ABSTRACT

Feature selection, as a dimensionality reduction technique, aims at selecting a small subset of the relevant features from the original ones by removing the irrelevant, redundant or noisy ones. Feature selection generally leads to better learning performance, i.e. higher learning accuracy, lower computational cost and better model interpretation. Feature selection methods such as Information Gain (IG), Mutual Information (MI) and Chi-square (Chi2) are statistical methods based on document frequency, but they do not take into account the frequency of terms within documents, nor do they consider their semantics.

Based on the idea that terms that frequently co-occur may have a common semantics and thus a high discrimination capacity compared to isolated terms, we propose a feature selection method for text classification considering two measures : term co-occurrence frequency and term entropy, where a term that frequently co-occurs with other terms and leads to minimize the uncertainty (entropy) of the class variable is considered relevant.

The performance of our method is compared to the four most commonly used selection metrics : Information Gain (IG), Mutual Information (MI), Chi-square (Chi2) and Document-Frequency (DF), using two classifiersNaïve Bayes (NB) and Support Vector Machine (SVM) and three datatsets.

Key words : selection , term , co-occurrence , entropy , text , classification .

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

INTRODUCTION GÉNÉRALE

1. Problématique

Actuellement, la classification de textes est un domaine de recherche très actif et l'automatisation de cette opération est devenue un enjeu pour la communauté scientifique. Une bonne classification ne peut se faire sans avoir trouvé un meilleur ensemble de caractéristiques (features) discriminatoires servant pour représenter efficacement les documents. L'utilité majeure de la classification est de déterminer de façon automatique dans quelle classe classer les textes à partir de leur contenu.

En plus des problèmes propres à la discipline de l'apprentissage automatique, notamment la classification, à savoir le sur-apprentissage ou la subjectivité de l'expert et/ou le déséquilibrage, etc. il existe également d'autres types de problèmes induits par la nature des données. Dans la classification des textes, le problème majeur est la haute dimensionnalité qui affecte la qualité de la classification, ce qui entraîne non seulement le coût en temps des algorithmes de classification, mais conduit souvent à ce qu'on appelle un sur-apprentissage (en. Overfitting)[49].

La classification s'apparente alors au problème de l'extraction de la sémantique d'un texte, puisque l'appartenance d'un document à une catégorie est étroitement liée à la signification de ce texte. La sélection des caractéristiques ou des attributs (en. Feature Selection - FS) pour la classification des textes est devenu un sujet de recherche populaire dans les conférences et revues sur l'intelligence artificielle et le data-mining pour le but d'optimiser les performances du système de reconnaissance, par élimination des caractéristiques redondantes ou non pertinentes qui n'aiderons pas à discriminer entre les classes selon un certain critère, et en suivant une certaine stratégie de sélection.

FS est bénéfique pour réduire la dimensionnalité des datasets, elle conduit à minimiser le temps de calcul et à améliorer les performances de la tâche de catégorisation.

Plusieurs algorithmes de sélection des caractéristiques, supervisés et non supervisés pour les données catégoriales et numériques existent dans la littérature, par exemple : Information Gain (IG), Gini Index (GI), Chi-Square (CH2), Mutuel information (MI), etc., ces algorithmes de sélection ne s'utilisent seulement pour les données textuelles mais aussi pour n'importe quel type de données numériques ou

catégoriales.

Le succès de ces algorithmes d'apprentissage repose sur leur capacité de générer des modèles complexes et d'extraire à partir de données des connaissances qui sont non seulement cachées, mais de plus qui sont pertinentes et utiles. Le problème que nous posons dans ce travail est comment faire une sélection de caractéristiques en se basant sur la sémantique des mots ?

Ainsi le premier problème auquel nous nous sommes confrontés est de trouver une méthode optimale de sélection des caractéristiques pertinentes, pour cela nous avons proposé une approche basée sur l'utilisation des matrices des co-occurrences en se basant sur l'idée que les mots qui coexistent fréquemment partagent une sémantique commune d'où une capacité de discrimination plus élevée que les mots traités séparément.

Dans nos expérimentations, nous comparons notre méthode de sélection de caractéristiques avec les méthodes classiques comme IG, CH2 et MI. Nous allons constater que notre méthode s'avère généralement plus performante que les méthodes classiques existantes.

2. Organisation du Mémoire

La suite de ce mémoire est divisée en quatre chapitres :

— **C**hapitre 1. **Sélection des Features pour la Classification des Textes** : dans ce chapitre nous allons présenter un aperçu général sur la sélection de features pour la classification des textes, en mettant l'accent sur les métriques de sélection les plus couramment utilisées telles que : Information Gain (IG), Mutual Information (MI), Chi-square test (Chi2), etc.

— **Chapitre 2** : **Classification supervisée des textes (état de l'art)** : dans ce chapitre nous allons présenter un état de l'art surla classification de textes, nous présentons en détail les algorithmes de classification usuels, leurs avantages et inconvénients, ainsi que les méthodes de pondération.

— **Chapitre 3** : **Sélection des Features pour la Classification des Textes basée sur la Fréquence des Termes Co-occurrents et la mesure d'Entropie** : méthode proposée : ce chapitre décrit en détail l'ensemble des étapes de notre méthode proposée pour la sélection des features.

— **Chapitre 4** :**Implémentation** : dans ce chapitre, nous présentons l'implémentation de notre travail en langage python, ainsi que l'évaluation de la performance de notre approche par comparaison des résultats obtenus à ceux obtenus par d'autres approches ou d'autres métriques .

— **Conclusion Générale** : Le travail se termine par une conclusion générale qui récapitule notre travail.

CHAPITRE 1

SÉLECTION DES FEATURES POUR LA CLASSIFICATION DES TEXTES

1.1 Introduction

La sélection des features (en. Feature Sélection), appelé aussi, sélection des caractéristiques, variables ou attributs, est un thème de recherche très actif en intelligence artificielle, notamment en analyse des textes et en traitement d'images.La sélection des features est un processus qui permet de sélectionner et de choisir un ensemble de grande taille, un sous-ensemble optimal pour étudier le problème.

L'analyse des textes en termes de classification permet de choisir un sous un sous-ensemble de termes (features) qui conduit à une meilleure performance.Dans ce chapitre, nous allons présenter le processus général de sélection des features, les méthodes et les métriques de sélection les plus largement utilisées pour la classification des textes.

1.2 Définition et Objectifs de Sélection des Features

1.2.1 Définition

La sélection des features est un processus qui permet de sélectionner un sous-ensemble d'attributs pertinents de l'ensemble de départ pour le processus de classification, régression ou de clustering. La figure 1.1 représente le principe général de sélection des features.

La définition proposée par est la suivante :
« Étant donnée une fonction permettant de mesurer la qualité d'un sous-ensemble de caractéristiques, la sélection des caractéristiques est réduite au problème de recherche d'un sous-ensemble optimal par rapport à cette mesure »[43].

Dans [54] la définition proposée est la suivante : « Étant donné un ensemble de dimension n, il faut sélectionner le sous-ensemble de dimension m tel que m < n, conduisant à taux d'erreur minimum ».

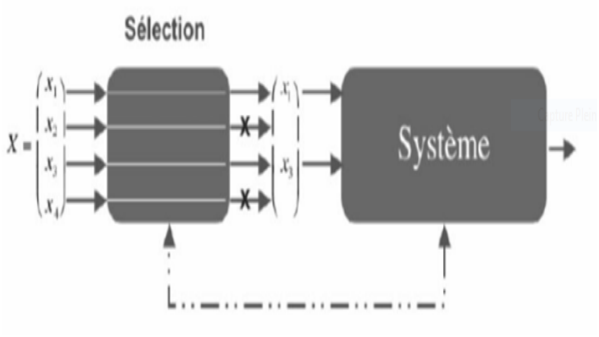

FIGURE 1.1 – Principe de sélection de features
[21].

1.2.2 Objectifs

En effet, les objectifs de sélection des features sont les suivants :

— La sélection des features nous permet de déterminer les features qui sont pertinents.

— La sélection de features nous permet de supprimer le bruit généré par les sous-ensembles non pertinents.

— Réduire les bases d'apprentissage et de test.

— Améliorer les performances et la vitesse de classification.

— Diminuer le temps d'apprentissage.

1.3 Processus de Sélection des Features

Dans un ensemble initial des features, qui forment l'ensemble d'apprentissage du problème étudié, le processus de sélection de features (variables) est illustré sur la figure 1.2.

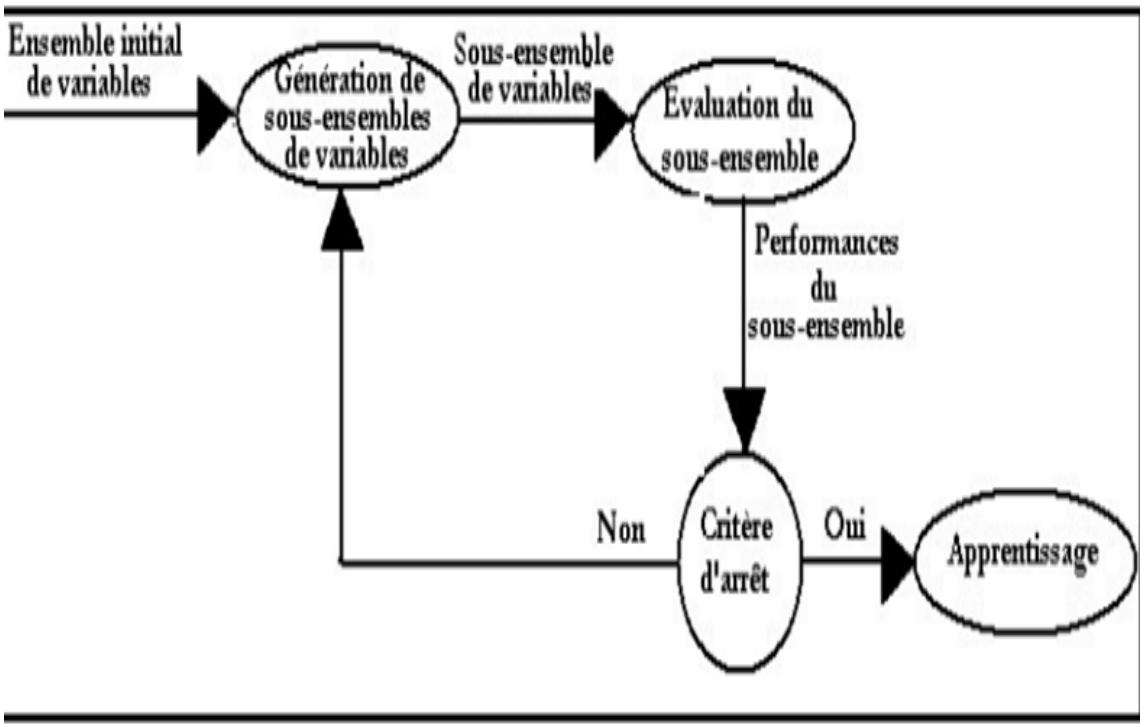

FIGURE 1.2 – Processus de sélection de features [1].

— A partir de l'ensemble initial des variables, le processus de sélection choisi un sous-ensemble de variables qui le considère le plus pertinent.

— Le processus de sélection applique une procédure d'évaluation qui permet s'estimer la pertinence du sous-ensemble.

— Après avoir évaluée la pertinence du sous-ensemble, le processus applique un critère d'arrêt qui teste si le sous-ensemble conduit à une performance meilleure du système, si oui le processus s'arrête, sinon il génère un autre sous ensemble.

— Le processus de « sélection d'attributs » qui a pour but de filtrer le vecteur des features de manière à en extraire l'information discriminante et pertinente en améliorant la qualité du système [24].

1.4 Méthodes de Sélection des Features

Dans cette section, nous présentons les méthodes de sélection de features, qui peuvent être généralement regroupés en deux catégories : méthodes supervisées et méthodes non supervisées.

1.4.1 Méthodes Supervisées

Ces méthodes suppriment les variables non pertinentes, ils se composent par trois approches principales : 'Filter', 'Wrapper' et 'Embedded'.

La figure 1.3 présente l'organigramme des trois approches de sélection des features.

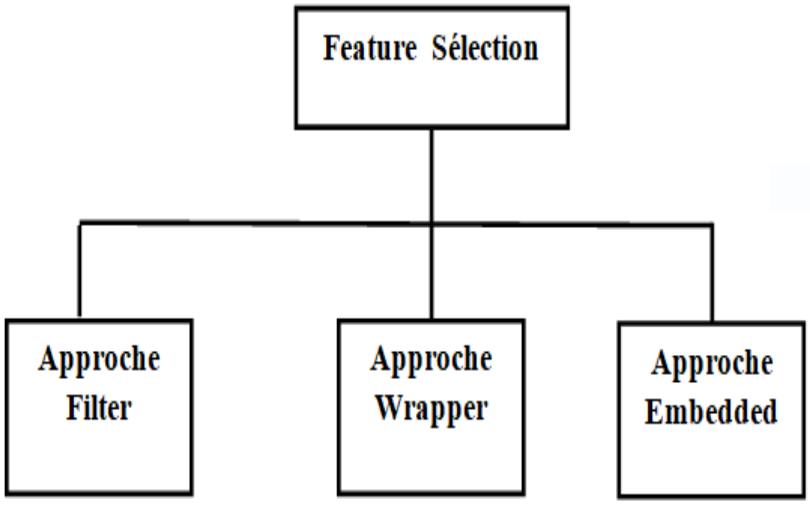

FIGURE 1.3 – Approches supervisées de sélection des features .

A) Approche Filter

L'approche Filter, c'est l'approche la plus utilisée dans la sélection des features. Les features sont sélectionnés sur la base de mesures statistiques. Cette approche ne dépend pas de l'algorithme d'apprentissage et choisit les features comme étape de prétraitement. Elle élimine les features non pertinents en gardant ceux qui augmentent la performance de l'algorithme de classification.
L'avantage d'utiliser des méthodes de filtrage est qu'elles nécessitent peu de temps de calcul et ne sur-ajustent pas les données.

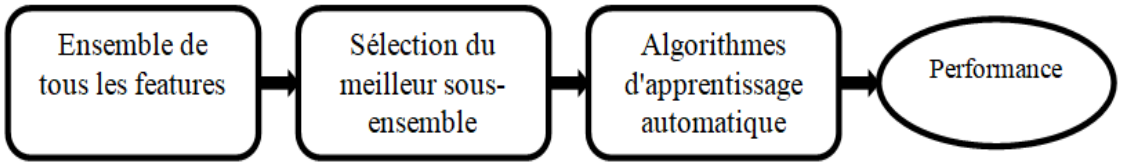

FIGURE 1.4 – L'approche Filtre.

Les principaux avantages des méthodes Filter sont les suivants :

— Les méthodes de filtrage sont indépendantes du modèle, et s'appuient entièrement sur les features de l'ensemble de données.

— Calcul très rapide basé sur différentes méthodes statistiques.
Mais, elles ont aussi les inconvénients suivants :

— La méthode de filtrage examine les features individuels pour identifier leur importance relative. Un feature peut ne pas être utile en elle-même, mais peut avoir une influence importante lorsqu'elle est combinée à d'autres features. Les méthodes de filtrage peuvent manquer de telles fonctionnalités.

— De plus, il n'est pas clair comment déterminer le point seuil pour sélectionnent uniquement les features requis et excluent le bruit.

A) Approche Wrapper

Les méthodes Wrapper choisissent les features pendant la phase d'apprentissage. Dans un premier temps, elles sélectionnent un sous-ensemble de features et évaluent ensuite le sous-ensemble sélectionné avec un modèle de classification. Les méthodes Wrapper ont de meilleurs résultats finaux que les méthodes de filtrage. De plus en évaluant toutes les combinaisons possibles de features par rapport au critère d'évaluation (en. Evaluation Criterion). Le critère d'évaluation est simplement la mesure de la performance qui dépend du type de problème. Enfin, il sélectionne la combinaison de features qui donne les résultats optimaux pour l'algorithme d'apprentissage automatique spécifié [35].
Le principal inconvénient des méthodes Wrapper est le risque de sur-ajustement et la complexité de calcul élevée.

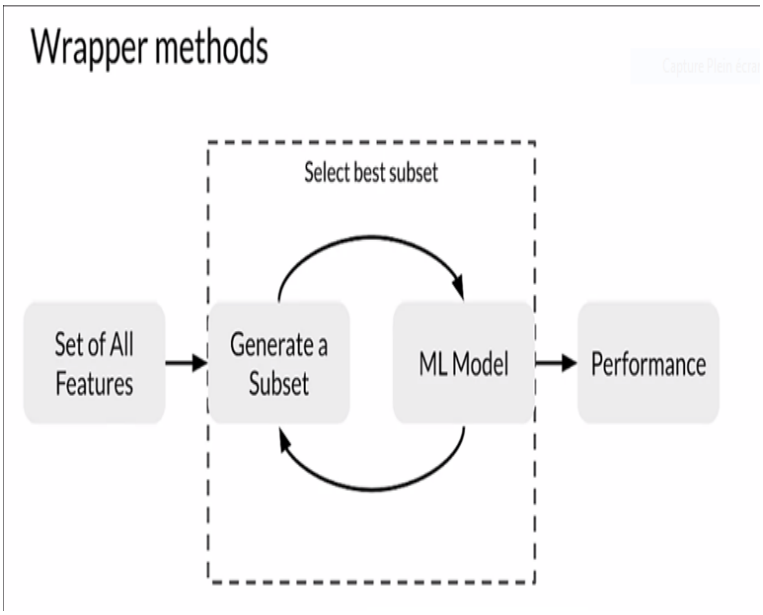

FIGURE 1.5 – L'approche Wrapper .
[2].

C) Approche Embedded

Les méthodes intégrées (en. Embedded) complètent le processus de sélection des features dans la construction de l'algorithme d'apprentissage automatique lui-même. En d'autres termes, elles effectuent une sélection de features lors de la construction du modèle, c'est pourquoi nous les appelons méthodes embarquées.

La méthode intégrée combine les avantages des méthodes Filter et Embedded :

— Elles prennent en considération l'interaction des features comme le font les méthodes Wrapper.

— Elles sont plus rapides et plus précises que les méthodes de filtrage.

— Elles trouvent le sous-ensemble optimal de features pour l'algorithme de classification.

— Elles sont beaucoup plus robustes au problème de sur-ajustement.
Cependant, l'inconvénient majeur de ces méthodes est qu'elles sont propres à l'algorithme de classification.

1.5 Méthodes non Supervisées

Les méthodes non supervise sont généralement utilisé pour les tâches de regroupement (Clustering). La figure 1.6décrit un cadre général de sélection non supervisée de caractéristiques, qui est très similaire à la sélection supervisée de caractéristiques sauf qu'aucune information d'étiquette n'intervient dans la phase de sélection des features et dans la phase d'apprentissage du modèle.

Sans l'information sur les étiquettes pour définir la pertinence des caractéristiques, la sélection supervisée s'appuie sur d'autres critères alternatifs pendant la phase de sélection. Un critère couramment utilisé permet de choisir des features qui peuvent le mieux préserver la structure de la multitude des données d'origine. Une autre méthode fréquemment utilisée consiste à rechercher des indicateurs de regroupement par le biais d'algorithmes de regroupement, puis à transformer la sélection non supervisée de caractéristiques en un cadre supervisé. Il existe deux façons différentes d'utiliser cette méthode. La première consiste à rechercher des indicateurs de regroupement et à effectuer simultanément la sélection supervisée de features dans un cadre unifié. L'autre méthode consiste à rechercher d'abord des indicateurs de grappes, puis à effectuer une sélection de features pour supprimer ou sélectionner certains features, et enfin de répéter ces deux étapes de manière itérative jusqu'à ce que certains critères soient remplis. En outre, certains critères de sélection supervisée des features peuvent encore être utilisés avec quelques modifications.

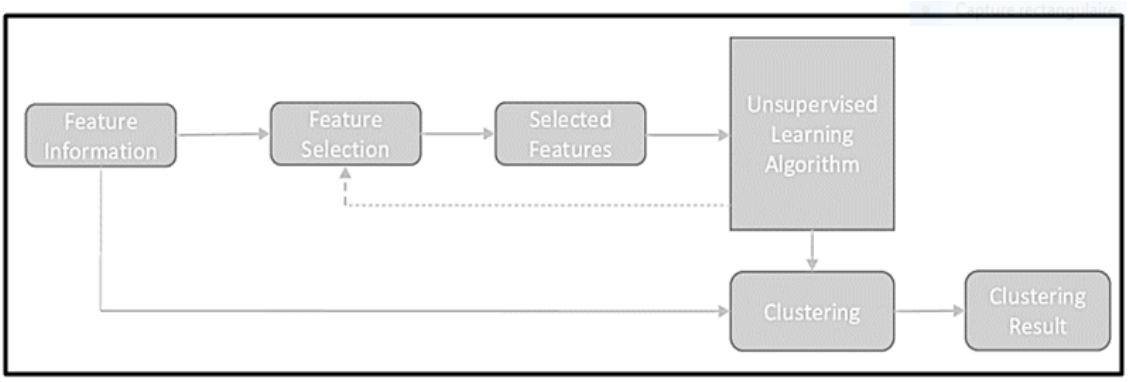

FIGURE 1.6 – Sélection non supervisé des features [36].

1.6 Algorithmes de Sélection des Features

Dans cette partie nous présentons quelques métriques de sélection de caractéristiques existant dans la littérature.

1.6.1 Chi-Square (Chi2)

Chi-square (X^2) [21] [2] est une procédure statistique bien connue utilisée pour tester l'indépendance des variables catégorielles en comparant les données réellement observées avec les données que l'on s'attendrait à obtenir selon une hypothèse précise.
X^2 est connu comme une méthode de sélection des features dans la classification supervisée des textes, qui vise à de trouver une valeur de test en fonction de la relation entre un terme t et une variable cible c, comme montrée par la formule 1.1 :

$$X^2_{(c,t)} = N\frac{(n_{tc}n_{\bar{t}\bar{c}} - n_{\bar{t}c}n_{t\bar{c}})^2}{(n_{tc} + n_{t\bar{c}})(n_{\bar{t}c} + n_{\bar{t}\bar{c}})(n_{tc} + n_{\bar{t}c})(n_{t\bar{c}} + n_{\bar{t}\bar{c}})} \tag{1.1}$$

Où n_tc désigne le nombre de documents appartenant à la catégorie c et contiennent le terme t, $n_{\bar{t}c}$ désigne le nombre de documents appartenant à la catégorie c mais ne contiennent pas le terme t,$n_{t\bar{c}}$ désigne le nombre de documents n'appartenant pas à catégorie c mais contenant le terme t,$n_{\bar{t}\bar{c}}$ désigne le nombre de documents n'appartenant pas à la catégorie c et ne contenant pas le terme t. N désigne le nombre total de documents.

1.6.2 Mutuel Information (MI)

MI est une mesure d'association, largement utilisée comme métrique de sélection, qui représente la corrélation entre les classes et les caractéristiques [53].
MI entre une valeur $x\epsilon X$ et une valeur $y\epsilon Y$ peut être défini par la formule suivante :

$$MI(x,y) = log\frac{p(x,y)}{p(x)p(y)} \tag{1.2}$$

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.

1.6.3 Gini index (GI)

L'indice de Gini est une méthode de sélection des caractéristiques qui mesure la pureté des features par rapport à la classe [47].La pureté fait référence au niveau de discrimination d'un feature pour distinguer les classes possibles [31].Pour la classification des textes, les features sont sélectionnés par la formule suivante :

$$GI(t) = \sum_{j=1}^{m} p(t|c_j)^2 p(c_j|t) \tag{1.3}$$

Ou, m le nombre de classes, $p(c_j|t)$ est la probabilité du terme t en donnant la classe c_j , $p(c_j t)$ est la probabilité de classe c_j étant donné le terme t .

1.6.4 Information Gain (IG)

IG est une méthode de sélection de features qui quantifie la qualité de la division de l'ensemble de données, calculée comme la différence entre l'entropie de l'ensemble de données avant et après la division[30] [46]. IG représente 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 est calculé comme suit :

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

=-$\sum_{j=1}^{m} p(c_j) log(p(c_j)) + p(t) \sum_{j=1}^{m} p(c_j/t) log(p(c_j/t)) + p(t) \sum_{j=1}^{m} p(c_j/t) log(p(c_j/t))$
(1.4)

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,$p(c_j)$ 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. $p(c_j/t)$et $p(c_j/t)$ sont les probabilités conditionnelles de la classe étant donné la présence ou l'absence du terme t.

1.6.5 Document Frequency (DF)

La fréquence des documents [20] [27] est une méthode simple et efficace de sélection des features. Elle fait référence au nombre de documents contenant un certain élément. L'idée de base est que les termes de basse fréquence sont inutiles pour la prédiction de catégorie et peut même influencer l'effet de classement final. Donc, ces mots à basse fréquence doivent être supprimés de l'espace initial des features, en gardant uniquement les termes dont la fréquence des documents est supérieure au seuil spécifié, La fréquence documentaire d'un terme est calculée comme suit :

$$DF(t) = \text{NOMBRE DE DOCUMENT CONTENANT t}$$

(1.5)

1.7 Conclusion

Dans ce chapitre, nous avons fourni un aperçu général et structuré sur la sélection des features. En premier lieu, nous avons donné quelques définitions et les objectifs de sélection des features. En second lieu, nous avons présenté les méthodes de sélection supervisés et non supervisée, ainsi que les différentes approches associées à la sélection supervisée Filter, Wrapper et Embedded et les avantages et les inconvénients de chacune.
Enfin, nous avons présenté le principe de fonctionnement des métriques les plus largement utilisées pour la sélection des données catégoriales qui sont CH2, IG, MI, GI et DF.
Dans le chapitre qui suit, nous allons présenter un état de l'art sur la classification des textes en raison de son lien étroit avec la sélection des features.

CHAPITRE 2

CLASSIFICATION SUPERVISÉE DES TEXTES (ÉTAT DE L'ART)

2.1 Introduction :

La classification de textes est un domaine où les algorithmes sont appliqués sur des documents de texte. La classification automatique de texte est traitée comme une technique d'apprentissage automatique supervisée. Elle est généralement effectuée sur la base de mots ou de features significatifs extraits du document texte. Les ensembles de données textuelles contiennent des unités de textes, documents. Un document se compose d'un nombre de phrases de sorte que chaque phrase comprend une suite de mots séparés évidement par des espaces. Dans la classification supervisée, i.e., catégorisation automatique, chaque document est étiqueté avec une valeur qui est la classe, étant donné que les classes sont prédéfinies, la classification automatique de texte a des applications importantes dans la gestion de contenu, la recherche contextuelle, l'exploration d'opinions, l'analyse des avis sur les produits, le filtrage des spams et l'exploration des sentiments de texte, elle a pour but de déterminer si un document donné appartient à la catégorie donnée ou non en regardant les mots ou termes de cette catégorie.
Dans ce chapitre, nous présentons une définition du concept d'apprentissage automatique. Nous introduisons les concepts de la classification supervisée et les différents types de classification ainsi qu'un état de l'art des algorithmes usuels en apprentissage automatique, les méthodes de pondération.

2.2 Classification Automatique

La classification automatique est un domaine dans l'intelligence artificielle, pour comprendre et de reproduire la faculté de l'apprentissage humain dans des systèmes artificiels. Il s'agit de concevoir des algorithmes capables de distinguer entre les objets à partir un ensemble d'apprentissage. L'objectif de ce domaine est de déterminer la relation entre les objets et leurs catégories pour la prédiction et la découverte des connaissances [48]. L'apprentissage automatique est divisé en trois types : l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage semi supervisé [22].

2.2.1 Définition

L'apprentissage supervisé consiste à construire un modèle basé sur un jeu de données étiquetés [22], Cette technique est utilisée dans plusieurs applications telles que les diagnostics médicaux et la prédiction des pannes.

2.2.2 Types de Classification Supervisée

Actuellement, la classification de textes est un domaine de recherche très actif et l'automatisation de cette opération est devenue un enjeu pour la communauté scientifique, les travaux évoluent considérablement depuis une vingtaine d'années et plusieurs modèles ont vu le jour comme le filtrage (classification supervisée bi-classe), le routage (classification supervisée multi-classe) ou le classement ordonnée (classement des textes par ordre de pertinence pour chaque catégorie).

A) Classification Binaire

Dans la classification binaire, les données sont étiquetées en utilisant deux classes. Elle consiste distinguer si un objet appartient ou non à une catégorie (classe).Par exemple, dans l'analyse des sentiments, un classificateur prend en compte les avis des clients pour classer un document à « positif » ou « négatif ».

B) La classification Multi-Classes

Dans la classification multi-classes, on peut se grouper le texte à une ou plusieurs classes voire à aucune classe. Le système répond donc à la question : « A quelles classes appartient le document ? ». C'est le cas le plus général dans la classification.

C) La classification Multi-Labels

La classification multi-labels si seulement si chaque document peut être groupe simultanément à deux ou plusieurs classes (ou labels).

2.3 Algorithmes de Classification

Il existe plusieurs algorithmes et techniques utilisés pour la classification supervisée. Ces derniers sont de différents types mais ayant tous la même intention d'avoir une bonne performance tôt en étant efficace. Chacun d'eux a ces propres avantages et inconvénients.

2.3.1 Classification Bayésienne

Naïve Bayes [38] C'est une méthode de classification statistique probabiliste qui est basée sur le théorème de Bayes. Elle est très utilisée dans plusieurs applications de classification des textes. Cette méthode est très utilisée dans le cadre du Machine Learning et plusieurs applications, elle consiste à connaitre le futur à partir du passé,

en plus les hypothèses (attributs) sont mutuellement exclusives et il y a une indépendance conditionnelle dans les hypothèses (attributs).
La règle de Bayes permet alors de calculer la probabilité a posteriori de la classe « C_k » quand x est observé :

$$p(C_i|X) = P(x|C_i)P(C_i)/P_jP(x|C_j)P(C_j)$$

(2.1)

Avantages L'hypothèse d'indépendance des descripteurs du classificateur Naïve Bayes le rend simple et efficace. Son entraînement ne nécessite pas beaucoup de documents, il a fait ses preuves dans la classification de documents courts, notamment les courriels (Ham/Spam) [40].

Inconvénients Contrairement aux documents courts, les documents longs posent un grand problème pour le classificateur Naïve Bayes, un riche vocabulaire favorise les dépendances entre les descripteurs (termes)[40].

2.3.2 Réseaux de Neurones

Un réseau de neurones est un ensemble de neurones connectés entre eux. En classification, les réseaux de neurones [25] permettent d'introduire de la non-linéarité dans la séparation entre les classes grâce au choix de la fonction d'activation .Ce réseau réalise un ou plusieurs fonctions d'activation de ses entrées.
Les réseaux de neurones ont été développés comme un modèle mathématique générique afin de modéliser les neurones biologiques. Ils comportent un certain nombre d'éléments de traitement d'information appelés neurones. Chaque neurone a son propre état interne interprété par la fonction d'activation. Il envoie son activation aux autres neurones sous forme de signaux. La connexion entre les neurones est réalisée via des liens orientés et pondérés. [33].
Un réseau de neurones est organisé sous forme de couches, contient des maillages reliés entre eux par des liaisons synaptiques, Il est basé sur l'expérience qui se constitue une mémoire lors de la phase d'apprentissage [19], ainsi que les neurones sont organisés en trois couches ou plus : les cellules d'entrée associées aux données, chaque neurone de sortie est associé à une classe, et les neurones cachés qui sont entre les neurones d'entrée et les neurones de sortie. La Figure 2.1 représente le principe fonctionnement des réseaux de neurone.

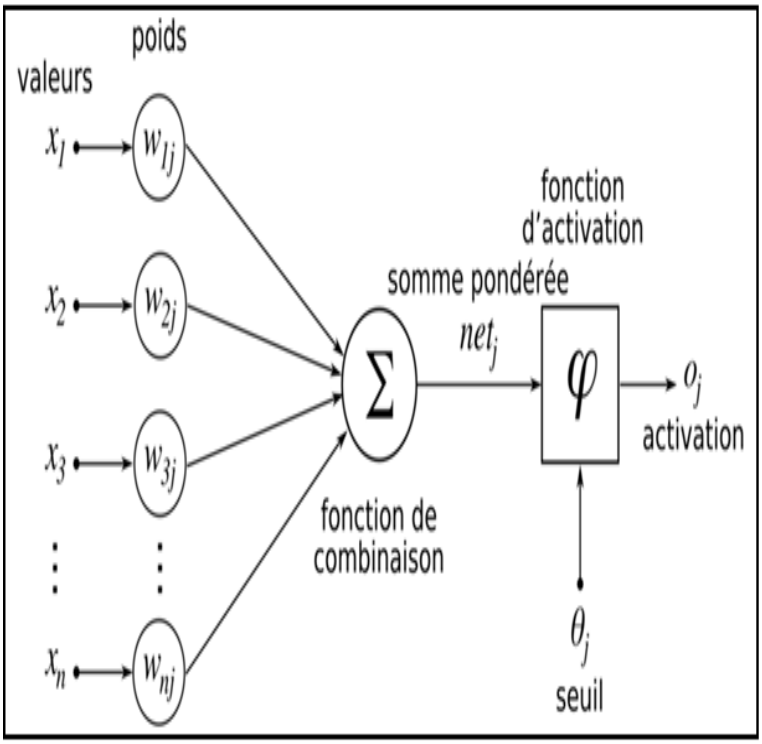

FIGURE 2.1 – Modèle de neurones formels.

La capacité de traitement d'un réseau est stockée sous forme de poids d'interconnexions obtenus par un processus d'apprentissage à partir d'un ensemble exemples d'apprentissage [29].

Avantages [xo]

— Les réseaux de neurones sont souples et génériques. Ils peuvent résoudre différents types de problèmes dont le résultat peut être : une classification, analyse de données, etc.

— Ils traitent des problèmes non structurés sur lesquels aucune information n'est disponible à l'avance.

— Les réseaux de neurones fonctionnent sur des données incomplètes ou bruitées. Cette lacune d'information peut être complétée par l'ajout d'autres neurones à la couche cachée.

Inconvénients

— La lenteur d'apprentissage [32].

— En cas d'erreur dans les résultats de sorties, l'utilisateur n'a aucune information sur le fonctionnement interne [32].

2.3.3 Support Vector Machine (SVM)

Les Machines à Vecteur Support (SVMs), ou en anglais Support Vector Machines, ont été développés par Cortes et Vapnik en 1995 . Elles sont des techniques d'apprentissage supervisé destinées à résoudre des problèmes de classification et de régression. Elles reposent sur deux notions principales : la notion de marge maximale et la notion de fonction noyau [32].
SVM est un classificateur linéaire, ça veut dire que, dans le cas idéal, les données doivent être linéairement séparables. En classification des textes, le corpus est représenté comme étant un espace vectoriel, où chaque terme est représenté par un point dans ce dernier, Il s'agit d'un ensemble de techniques destinées à résoudre des problèmes de discrimination (prédiction d'appartenance à des groupes prédéfinis) et de régression (analyse de la relation d'une variable par rapport à d'autres) [22].
Leur principe est de trouver le meilleur hyperplan optimal, qui sépare notre corpus en deux phases en cherchant la plus grande marge possible pour séparer les données de classes opposées. La marge est l'espace entre les 2 phases qui sont définis par les points (vecteurs des supports) entre l'hyperplan optimal. Le but essentiel est de maximiser cette marge.
Toutefois, si les données ne sont pas linéairement séparables, la SVM peut être modifiée pour accepter un minimum d'erreurs. Cette fois, le but est de maximiser la marge et de minimiser l'erreur de classification. La figure 2.2 présente un exemple d'hyperplan linéairement séparable, elle choisit la marge maximale.
Pour résoudre le problème du non linéarité séparatrice, l'idée des SVM est d'augmenter la dimension d'espace de données. Dans ce cas, il est alors probable qu'il existe un séparateur linéaire. En effet, la chance de trouver un hyperplan séparateur augmente proportionnellement avec la dimension d'espace de données [32].

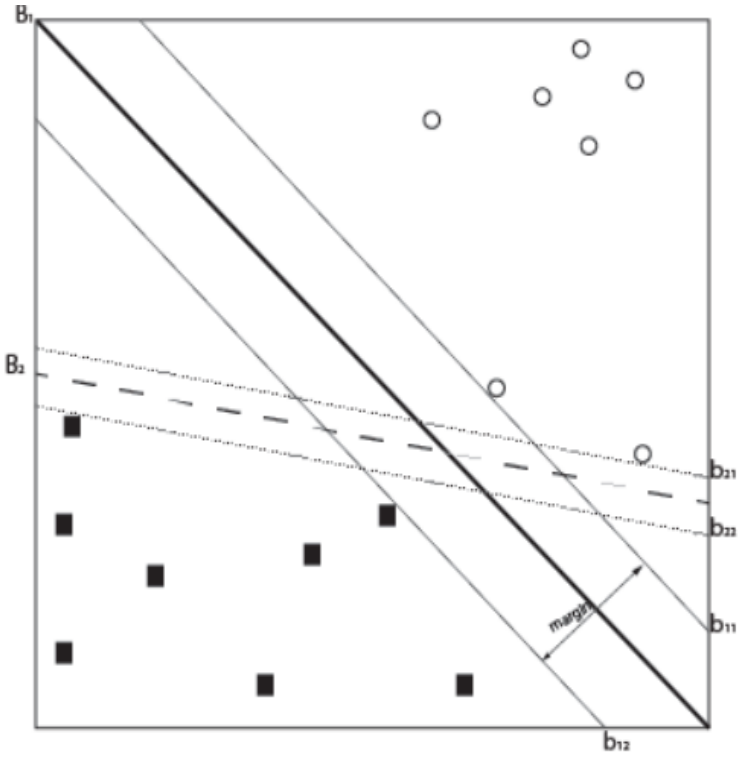

FIGURE 2.2 – Exemple d'hyperplan optimal.
[40].

Avantages

— C'est un classificateur qui se fonctionne très bien avec les nouvelles données, puisque les vecteurs de support sont les points les plus proches du séparateur est non les plus loin. Toute fois, les valeurs anormales n'affectent en aucun cas le classificateur.

— Sa flexibilité quant au traitement des cas non-linéairement séparables.

— Différentes fonctions noyau peuvent être spécifiées [32].

Inconvénients

— Son temps d'apprentissage est supérieur, comparativement aux arbres de décision.

— Un SVM est considéré comme classificateur bi-classe (binaire), mais des solutions ont été proposées dans ce sens, notamment, la transformation du problème de classification multi-classes en sous problème bi-classes [40]
.
— Elles utilisent des fonctions mathématiques complexes pour la classification [32]
.
— Les machines à support de vecteurs demandent un temps énorme durant les phases de test [3].

2.3.4 Arbre de Décision (AD)

Un arbre de décision (eng. Decision Tree) est constitué d'un ensemble de nœuds liés par des branches, il s'étend du haut vers le bas [34]. Le nœud racine est situé au sommet de l'arbre par contre les nœuds feuilles sont placés au bas de l'arbre. L'arbre de décision peut classer les textes en plusieurs catégories.
En commençant par le haut, nous devons mettre dans le nœud racine le descripteur qui discrimine le meilleur possible les textes de notre corpus, pour obtenir de nouveaux sous nœuds, ainsi de suite, on répète cette étape pour chaque nœud, jusqu'à ce que la séparation des textes n'est plus possible. À la fin, les nœuds feuilles sont constitués d'ensemble de textes de même catégorie. Le critère d'arrêt est le suivant : si un sous-ensemble donné dans lequel on trouve que toutes les instances appartiennent à la même classe alors on va créer une feuille d'arbre, et la fin du processus pour ce chemin. Pour le cas contraire, un sous-ensemble avec des instances de différentes étiquettes, le processus de construction du nœud est repris.
L'avantage majeur de ce type est qu'il est très semblable au raisonnement humain, de ce fait il est très facile d'expliquer les résultats obtenus et son comportement.

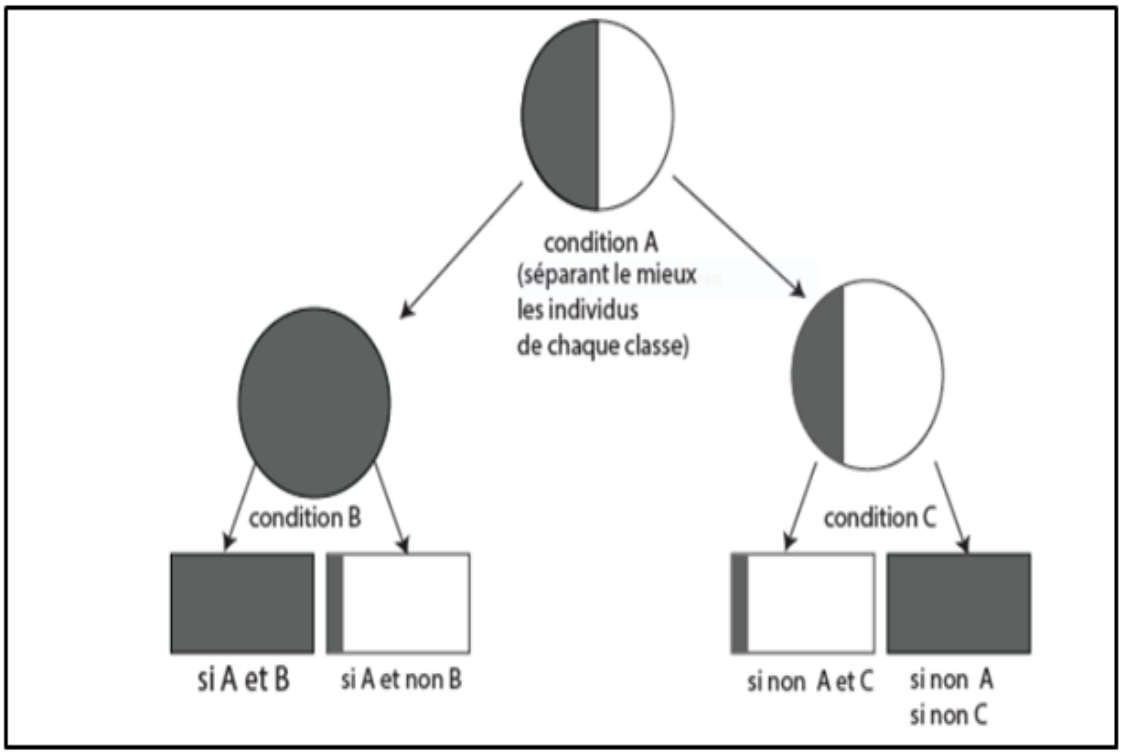

FIGURE 2.3 – Arbre de décision.
[50].

Avantages

Les arbres de décision sont bâtis selon un ensemble de règles très explicites ce qui rend les résultats bien compris par l'utilisateur [40]. En générale, ils demandent peu de ressources, et leur temps d'apprentissage et de test sont relativement courts.

Inconvénients

Quand les arbres deviennent assez grands, ça veut dire, ils comportent beaucoup de feuilles, ils perdent leur pouvoir explicatif. De plus le fait que les sous-nœuds dépendent directement du nœud racine rend la présence ou l'absence d'un seul descripteur dans un texte déterminant sur le sort de son classement. D'autant plus que la modification d'un seul nœud, s'il est près du sommet, modifie entièrement l'arbre [40].

2.3.5 Forêts d'Arbres Décisionnels (eng. Random Forest Classifier - RFC)

C'est un algorithme de classification qui a pour but de réduire la variance des prévisions des arbres de décision et améliorer leur performance. C'est une application de graphe en arbres de décision permettant ainsi la modélisation de chaque résultat sur une branche en fonction des choix précédents. On prend ensuite la meilleure décision en fonction des résultats qui suivront. On peut considérer ceci comme une forme d'anticipation [22]. La figure 8 illustre un aperçu sur le principe du fonctionnement de RFC :

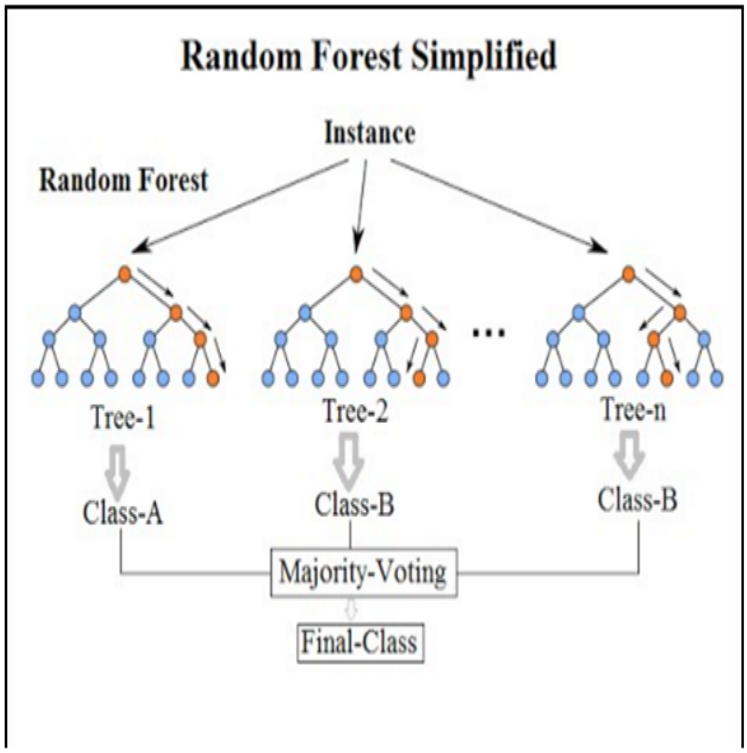

FIGURE 2.4 – Structure de l'algorithme Random Forest.

2.3.6 Le boosting

Le boosting est une méthode proposé pour combiner un ensemble d'apprenants faibles en un apprenant fort afin de minimiser les erreurs d'apprentissage. Il est utilisé pour résoudre divers problèmes de classification tels que la classification des textes, traitement du langage naturel, etc .L'algorithme le plus pratiqué s'appelle AdaBoost [26] le plus connu utilisant cette technique. La critique majeure de ces méthodes est qu'elles sont très sensibles aux données bruitées, vu qu'ils visent à pondérer les exemples mal classés dans chaque itération, cela peut conduire évidemment à une situation de sur-apprentissage.

2.3.7 k-Nearest Neighbor (k-NN)

Le k-plus proches voisins (k-NN)a vu son origine avec (Fix et Hodges 1951) est un type d'algorithme d'apprentissage supervisé utilisé pour la classification. C'est une méthode simple qui essaie de prédire la classe correcte pour les données de test en calculant la distance entre les données de test et tous les points d'apprentissage. Sélectionnez ensuite le nombre K de points qui est le plus proche des données de test.
L'algorithme k-NN calcule la probabilité que les données de test appartiennent aux classes de données d'apprentissage « k » et que la classe détient la probabilité la plus élevée sera sélectionnée. Il est également intéressant de noter que l'augmentation la valeur de k ne dégrade pas significativement les performances.
Cette méthode utilise principalement deux paramètres : une fonction de similarité pour comparer les individus dans l'espace de features et le nombre k qui décide

combien de voisins influencent la classification [32].

Avantages :

— L'algorithme KNN est robuste envers des données bruitées [52].

— La méthode des k plus proches voisins est efficace si les données sont larges et incomplètes [4].

— Cette méthode est l'une des plus simples de tous les algorithmes d'apprentissage automatique [5].

Inconvénients :

— Le besoin de déterminer la valeur du nombre des plus proches voisins (le paramètre k) [6].

— . Le temps de prédiction est très long puisqu'on doit calculer la distance de tous les exemples [7].

— Cette méthode est gourmande en espace mémoire car elle utilise une grande capacité de stockage pour le traitement des corpus [32].

2.4 Évaluation des modèles de classification

2.4.1 Matrice de Confusion

La matrice de confusion en classification du texte est un tableau utilisé pour décrire les performances d'un modèle de classification sur un ensemble de données de test dont les vraies valeurs sont connues. En d'autres termes, elle indique si un classificateur fonctionne bien et à quel degré de fiabilité. Chaque classe est représentée par une colonne et une ligne.

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

TABLE 2.1 – Matrice de Confusion.

Où :
VP (Vrais Positifs) : Les instances de la classe Positive correctement prédites.
FP (Faux Positifs) : Les instances de la classe Négatives prédites comme Positives.
VN (Vrais Négatifs) : Les instances de la classe Négatives correctement prédites.
FN (Faux Négatifs) : Les instances de la classe Positive prédites comme Négatives.
Notons bien que c'est nous calculons ses paramètres, nous pouvons calculer la précision, le rappel et le score F1.

2.4.2 Précision

La précision est un rapport entre les observations correctement prédites et le nombre total d'observations. Son but est d'évaluer les performances des modèles de classification a deux classes ou plus.

$$\textbf{Précision (C)} = \frac{\textbf{VP}}{\textbf{VP+FP}} \tag{2.2}$$

2.4.3 Rappel (eng. Recall)

Le rappel (sensibilité) est le rapport entre les observations positives correctement prédites et toutes les observations de la classe réelle.

$$\textbf{Rappel (C)} = \frac{\textbf{TP}}{\textbf{TP+FN}} \tag{2.3}$$

2.4.4 Taux d'erreur et de Succès

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

$$\textbf{Taux de succès} = \frac{\textbf{VP+VN}}{\textbf{VP+FP+VN+FN}} \tag{2.4}$$

D'autre part le taux d'erreur est le rapport entre les observations mal classés sur le nombre total des observations.

$$\textbf{Taux d'erreur} = \frac{\textbf{FP+FN}}{\textbf{VP+FP+VN+FN}} \tag{2.5}$$

2.4.5 F1-Mesure

F1-Mesure est un indicateur qui arrange le rappel et la précision, calculé par la formule suivante :

$$\textbf{F1-Mesure} = \textbf{2} \times \frac{\textbf{precision * rappel}}{\textbf{precision+rappel}} \tag{2.6}$$

2.5 Méthodes de Pondérations

Pour toute représentation de textes, et notamment le modèle sac de mots (eng. Bag of Words –BOW), il faut coder les termes d'un vecteur dans lequel un document est représenté. Évidemment, la formule la plus simple est de pondérer les différents mots apparaissant dans les textes par des valeurs binaires égales à 1 ou 0 qui expriment leur présence et leur absence, respectivement. Une autre méthode est de les représenter à l'aide de leurs fréquences désignant les nombres d'occurrences des

termes pour un tel document. En général, dans le modèle de sac de mots, un vecteur comporte généralement les poids w_i de chaque mot i pour le document k. Soit la notation suivante :
La fréquence du mot i dans le document k.

— N : le nombre de documents dans la collection.

— n_i le nombre total de fois que le mot i se produit dans la collection.

2.5.1 TF-IDF

La pondération TF-IDF acronyme de (Term Frequency Inverse Document Frequency) est la formule la plus répondue dans le codage de textes. Elle est issue du monde de la recherche d'information [45]. Le codage TF-IDF prend en compte deux critères importants à la fois pour un terme : le premier (TF : Term Frequency) exprime l'importance locale est souvent mesurée par la fréquence brute du terme dans le document ou par son log tandis que le deuxième (IDF : Inverse Document Frequency) exprime l'importance globale, qui est l'inverse du nombre des documents dans lesquelles le terme est apparu. La mesure IDF est définie comme suit :

$$IDF(t) = log\frac{N}{df(t)} \tag{2.7}$$

Où t est un terme, N est le nombre total de documents dans la collection, et df(t) est le nombre de documents qui contiennent le terme t. Le poids du terme t dans un document d est alors défini comme suit dans la pondération TFI-DF :

$$W(d,t) = TF(d,t) \times IDF(t) \tag{2.8}$$

Ainsi, un terme qui a une valeur de TFI-DF élevée doit être à la fois important dans le document auquel ce terme est associé, et doit apparaitre peu souvent dans les autres documents.
Cette méthode peut également être considérée comme une forme de modèle de sac de mots, car elle ne prend pas en compte la grammaire ou l'ordre.

2.5.2 Bag-Of-Words (BOW)

Sac de mots (Bag of Words) est l'une des méthodes de représentation les plus connues pour la catégorisation de textes. Le nom Bag of Words fait référence au fait que ce modèle ne tient pas compte de l'ordre des mots. Au lieu de cela, on peut imaginer que chaque mot est mis dans un sac, où l'ordre des mots se perd. Bien qu'il existe quelques variantes différentes de ce modèle, la plus courante consiste simplement à compter le nombre d'occurrences de chaque mot dans un document et à conserver le résultat dans un vecteur [19]. De cette façon, les fréquences des termes restent intactes, bien que la grammaire et l'ordre soient perdus [37].

2.6 Conclusion

La classification est une tâche très importante dans le data-mining, et qui nécessite beaucoup de recherches pour son optimisation. La classification supervisée est l'une des techniques les plus utilisées dans l'analyse des données. Elle permet d'apprendre des modèles de décision qui permettent de prédire les catégories des exemples futurs.
Dans le prochain chapitre, nous exposons notre propre méthode pour la sélection des features basée sur la corrélation sémantique des mots.

CHAPITRE 3

SÉLECTION DES FEATURES POUR LA CLASSIFICATION DES TEXTES BASÉE SUR LA FRÉQUENCE DES TERMES CO-OCCURRENTS ET LA MESURE D'ENTROPIE : MÉTHODE PROPOSÉE

3.1 Introduction :

Classer les textes automatiquement nécessite une phase de sélection des features pour réduire l'espace vectoriel et augmenter la performance du modèle construit. Cependant,la difficulté majeure de la catégorisation de texte est la grande dimensionnalité de l'espace de features. Les méthodes de sélection automatique de features telles que le seuillage de fréquence de document (DF), le gain d'information (IG), l'information mutuelle (MI), Chi-Square (Chi2) etc , sont couramment appliquées dans la catégorisation de texte, mais elles ne prennent pas en compte le nombre d'occurrences du terme dans le document ni la corrélation entre les termes. Selon notre analyse, l'utilisation des termes fréquents et qui co-existants avec d'autre termes peuvent avoir une capacité de discrimination plus élevée que termes traité de façon individuelle.

Dans ce chapitre, nous allons présenter l'inconvénient des métriques existantes de sélection des features, ainsi que l'utilité des termes fréquents et co-occurents, puis nous présentons notre une nouvelle approche de sélection de features en prenant en compte les fréquences des termes et la corrélation sémantique entre eux.

3.2 Inconvénient majeur des métriques existantes

quatre méthodes sont incluses dans cette étude DF, IG ,Chi2 et MI qui sont couramment utilisées dans la sélection de features pour la catégorisation de texte. Le seuillage DF est la technique la plus simple pour la réduction du vocabulaire. Il s'adapte facilement à de très grands corpus avec une complexité de calcul approximativement linéaire dans le nombre de documents de d'apprentissage.

Le gain d'information (IG) est couramment utilisé comme critère de qualité du terme dans l'apprentissage automatique [44, 39] Il mesure la quantité d'information

obtenue pour la prédiction de catégorie en connaissant la présence ou absence d'un terme dans un document.

L'inconvénient majeur des métriques existantes est qui ne prennent pas la fréquence des termes (Term-Frequency) et la corrélation (dépendance) sémantique entre les termes, et traite chaque mot de façon séparée des autres, alors que la co-existence des mots peut avoir une capacité de discrimination plus élevés que les mots évalués individuellement. En plus, elles ne sont non fiables pour les termes de basse fréquence, c'est-à-dire que les termes de basse fréquence seront filtrés en raison de leurs poids et ils ne comptent que si un terme apparaît dans un document et ignorent la fréquence des termes à l'intérieur des documents. En fait, les termes à haute fréquence (à l'exception des mots vides) présents dans peu de documents sont souvent considérés comme des discriminants dans le corpus réel.

3.3 Utilité de fréquence des termes (Term-Frequency)

La fréquence des termes (TF) signifie le nombre d'occurrences du terme dans un document. La fréquence des termes est couramment utilisée dans les tâches d'exploration de texte, d'apprentissage automatique et de recherche d'informations.

Comme les documents peuvent avoir des longueurs différentes, il est possible qu'un terme apparaisse plus fréquemment dans des documents longs que dans des documents courts. Pour cette raison, il semblera qu'un terme est plus important dans un document long que dans un document court.

Soit D : un document et T : un terme, la fréquence du terme T dans le document D, et on la note par $TF_{(T,D)}$ est calculée par cette formule :

$$tf_{T,D} = \frac{n_{(T,D)}}{\sum_{terme} n_{terme,D}} \tag{3.1}$$

Où : $n_{(T,D)}$: le nombre d'occurrence de terme T dans le document D.
$\sum_{terme} n_{terme,d}$: La somme des occurrences de tous les termes qui apparaissent dans le document D [8].

3.4 Méthode proposée

Notre travail proposé vise à étudier l'impact de la sélection des features basée sur la fréquence des termes co-occurrents qui prend en compte les fréquences des termes et la corrélation sémantique entre eux sachant les termes co-occurrents ont une capacité de discrimation plus élevée que les termes isolés. Les étapes de notre méthode sont les suivantes :

3.4.1 Prétraitement et Extraction du Vocabulaire

La phase de prétraitement joue un rôle décisif où la performance du classificateur est influencée de façon directe et considérable. C'est-à-dire, un bon nettoyage améliore le résultat de la classification peu-importe la capacité du classificateur appliqué et l'inverse est vrai. Le nettoyage permet éventuellement de dégager tous ce qui est bruit, mot inutile, ainsi que tout mot contribuant de façon négative dans un contexte de catégorisation malgré son importance et son apport dans le texte . D'après la loi de ZipF [55] les termes importants sont les termes qui ne sont ni trop fréquents et ni trop rares. En plus des termes fréquents, on en trouve un autre type très particulier qui est les mots vides, ou les mots outils. Ce sont donc, et sans aucun doute, utiles pour toute sorte de langue mais malheureusement sans intérêt dans un contexte de catégorisation. Généralement, dans la phase de prétraitement, une liste préétablie pour chaque langue est utilisée à la suppression de ces mots. À titre d'exemple, une liste de mots vides en français contient les termes suivants : 'le, la, les, ce, ceux, donc, mais, ou . . ., etc.Ces mots sont donc jugés bruit, non porteurs d'information, peu importe leurs qualités statistiques ou sémantiques dans les textes. Il est à noter que ce genre de mots subit un traitement spécifique appelé, l'élimination des mots vides.

Nous construisons le corpus prétraité et le vocabulaire des termes unique à partir de corpus documentaire. Dans cette phase nous considérons que les étapes de prétraitement courantes 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, seront, etc.).Élimination des mots dont la taille est inférieure à un seuil donné.

Le but de ces méthodes de prétraitement est de transformer les données dans un format plus agréable pour l'extracteur des features, et également de supprimer les informations superflues.

Soit $D = d1, \ldots, dn$ le corpus des textes du document, les étapes de prétraitement et d'extraction du vocabulaire sont présentées dans l'Algorithme 1 :

Algorithme 1 Prétraitement et extraction du Vocabulaire

Entrées: D : document Corpus ;
Sorties: P : corpus prétraité ; V : vocabulaire des termes uniques ;

Pour tout document $d \in D$ **Faire**
    $Lower\_case(d)$
    $Remove\_html\_tags(d)$
    $Remove\_special\_chars(d)$
    $Remove\_digits(d)$
    $Remove\_stop\_words(d)$
    $Lemmatize(d)$
Fin Pour
Pour chaque document $d \in P$ **Faire**
    Pour tout terme $t \in d$ **Faire**
        Si $t \notin V$ **Alors**
            $V = V \cup t$
        Fin Si
    Fin Pour
Fin Pour
Renvoyer P,**T**

3.4.2 Construction de la matrice de cooccurrence

La matrice de cooccurrence est une matrice carrée TT, où T est le nombre total de mots considérés. On note $n_{(i,j)}$ le nombre de fois où le mot j se trouve à l'intérieur d'un paragraphe qui contient le mot i. Dans cette notation, i est le pôle, j est l'un de ses co-occurrents. Notons que cette matrice est symétrique, puisque $n_{(i,j)} = n_{(j,i)}$ Cette matrice de cooccurrence permet de définir un graphe pondéré et orienté où les mots sont les sommets et les valeurs $n_{i,j}$ sont les poids des arêtes [23].

Supposons que nous ayons un grand corpus composé d'un total de N mots et soit $w_1, w_2, , ..., w_{N_w}$ désignent les différents mots dans le corpus. Soit $N_i, i1, 2, ...N_w$ dénotons le nombre de fois que w_i apparaît dans le corpus et soit $N_{i,j}, i, j1, 2, ..., N_w$ désignent le nombre de fois que w_i se produit au voisinage de w_j dans le corpus. Le voisinage d'un mot, w_i , est typiquement les mots les plus proches de w_j devant et derrière dans le texte. On suppose la symétrie tel que $N_{i,j} = N_{j,i}$ [9].

Avantages de la matrice de cooccurrence :

— Elle préserve la relation sémantique entre les mots.
— Elle utilise la factorisation qui est un problème bien défini et peut être résolu efficacement.
— Elle doit être calculée une fois et peut être utilisée à tout moment une fois calculée.

Inconvénients de la matrice de cooccurrence : Il nécessite une énorme mémoire pour stocker la matrice de cooccurrence.Mais, ce problème peut être contourné en factorisant la matrice hors du système, par exemple dans les clusters Hadoop, etc. et peut être enregistrée.

La matrice de cooccurrence terme-terme est une matrice carrée symétrique largement utilisée dans les fouilles des textes, qui capture la fréquence des mots apparaissant ensemble dans l'espace latent et trouve la relation sémantique entre les mots en fonction du contexte dans lequel les mots apparaissent. Le contexte du mot (terme) peut être le document entier dans lequel il apparaît, une phrase ou une fenêtre coulissante fixe.

Soient D le corpus de documents et $V = t1, \ldots, tn$ le vocabulaire des termes uniques extraits de D après prétraitement (Algorithme 1). La matrice de cooccurrence terme-terme T construite avec D et V a la forme générale illustrée par la formule suivante :

$$\begin{matrix} t_1 & \cdots & t_n \end{matrix}$$

$$\mathrm{T}=\begin{bmatrix} f_{11} & \cdots & f_{1n} \\ \vdots & f_{ij} & \vdots \\ f_{n1} & \cdots & f_{nn} \end{bmatrix}$$

FIGURE 3.1 – Matrice de cooccurrence.

Où f_{ij} indique la fréquence à laquelle un terme t_i coexiste avec un autre terme t_j. Notons que la valeur f_{ij} peut changer lors de la modification de la taille de la fenêtre contextuelle.
L'algorithme suivant montre comment construire une matrice de cooccurrence en utilisant une fenêtre contextuelle glissante de taille w_s, où w_s signifie que l'on prend w_s termes à gauche et w_s termes à droite du terme courant.

Algorithme 2 Matrice de cooccurrence

Entrées: D : Corpus ; ws : Window Size ;
Sorties: T : Matrice de cooccurrence ;

V, P = Prétraitement et extraction du vocabulaire() ;
Pour tout document $d \in P$ **Faire**
    Pour $i = 1 \rightarrow taille(d)$ **Faire** '
        $current\_term = d[i]$
        $left\_context = max(i - ws, 0)$
        $right\_context = min(i + ws, len(text) - 1)$
        $text[left\_context : right\_context]$
    Fin Pour
Fin Pour
Pour tout $context\_term$ **dans** $context$ **Faire**
    $j = index(context\_term)$
Fin Pour
$T[i, j] += 1$
Renvoyer T

3.4.3 L'entropie conditionnelle en théorie de l'information

a)l'entropie

Nous introduisons d'abord le concept d'entropie, qui est une mesure de l'incertitude d'une variable aléatoire. Soit X une variable aléatoire discrèteavec une fonction de masse de probabilité $p(x) = PrX = x, x \in X$.
L'entropie de X est définie par la formule suivante :

$$H(X) = -\sum_{x} p(x) log_2 p(x) \tag{3.2}$$

L'entropie mesure l'incertitude attendue dans X. Nous disons également que $H(X)$ est approximativement égale à la quantité d'information que nous apprenons en moyenne à partir d'une instance de la variable aléatoire X. Notez que la base de l'algorithme n'est pas importante puisque la modification de la base ne change la valeur de l'entropie que par une constante multiplicative [18].

b)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, moyennée sur la variable aléatoire conditionnant [51].

On peut définir **l'entropie conditionnelle** $H(X|Y)$, qui est l'entropie d'une variable aléatoire X conditionnelle à la connaissance d'une autre variable aléatoire Y.

La réduction de l'incertitude due à une autre variable aléatoire s'appelle l'information mutuelle [51].

L'entropie conditionnelle est une mesure de l'incertitude qui reste sur la variable aléatoire Y lorsque nous connaissons la valeur de X [18].

Si $(X, Y) \approx p(x, y))$, l'entropie conditionnelle $H(Y|X)$ est défini comme [51].

$$H(Y|X) = -\sum_{x \in X} p(x) H(YX)$$

$=\sum_{x \in X} p(x) \sum_{y \in Y} p(y|x) log p(y|x) = -\sum_{x \in X} \sum_{y \in Y} p(x, y) log p(y|x)$(3.3)

Elle peut être interprétée comme l'incertitude sur Y lorsque X est connue.

Propriétés de l'entropie [18]

— **Non négativité** : $H(X) \geq 0$, l'entropie est toujours non négative. H(X) = 0 si X est déterministe.

— **Mono-tonicité** le conditionnement réduit toujours l'entropie :

$$H(Y|X) \leq H(Y) \tag{3.4}$$

— **Sous fonctions non croissantes** Soit X une variable aléatoire et soit $g(X)$ une fonction déterministe de X. Nous avons que :

$$H(Y) \geq H(g(X)) \tag{3.5}$$

3.5 Schéma de pondération proposé

Pour chaque catégorie c_k de l'ensemble des catégories C=c1,...,cm, une matrice de cooccurrence Terme-Terme T est construite. Maintenant, pour chaque terme t_i du vocabulaire V=t1,t2,...tn, la fréquence de cooccurrence est calculée avec une catégorie c_k comme dans l'équation suivante :

$$CF(t_i, c_k) = \frac{F(t_i, c_k)}{\sum_{l=1}^{m} F(t, c_l)}$$

(3.6)

Où, $F(t_i, c_k)$ est la somme des fréquences de cooccurrence f_{ij} de t_i avec les autres termes t_j, comme indiqué dans l'équation :

$$F(t_i, c_k) = \sum_{i=1}^{n} f_{ij} \tag{3.7}$$

Le score global de cooccurrence pour chaque terme ti est calculé comme suit :

$$CF(t_i) = max_k(CF(t, c_k)) \tag{3.8}$$

Lorsqu'un terme apparaît fréquemment dans une catégorie et rarement dans les autres catégories, la fréquence de cooccurrence du terme est élevée, ce qui signifie que le terme a une bonne distinction sémantique lorsqu'il est inclus avec d'autres termes dans la même catégorie.

Après prétraitement du corpus textuel, extraction du vocabulaire des mots uniques, calcul de la matrice de cooccurrence, l'algorithme suivant, intitulé **MCEC (Sélection des features par Matrice de Cooccurrence et Entropie Conditionnelle)** résume l'ensemble des étapes du schéma de pondération :

Algorithme 3 MCEC

Entrées: $V = \{mot\ uniques\}; C = \{c_1, c_2, \ldots ...c_n\};$
Sorties: mots pondérés ;

Pour $c_j \in c$ **Faire**
  Pour $t \in v$ **Faire**
    Calculer :
    $f(t) \leftarrow CF(t)$ # fréquence de co-occurrence
    $H \leftarrow H(t, c_j)$ # entropie conditionnelle
    $W(t) = f(t) \times (H(c) - H(c \mid t))$
  Fin Pour
Fin Pour

3.6 Conclusion

Dans ce chapitre, nous avons expliqué les étapes détaillées de notre approche, où nous avons fait une combinaison de deux mesures pour la pondération des termes :la fréquence de cooccurrenceet l'entropie conditionnelle.
Le chapitre suivant exposera la partie expérimentale de notre projet, où les résultats obtenus seront analysés et interprétés.

‘

CHAPITRE 4
IMPLÉMENTATION

4.1 Introduction

Après avoir terminé le chapitre de conception de notre approche, nous présentons dans ce chapitre deux parties :

La première partie est consacrée à définir 'environnement de développement, et les bibliothèques utilisées. La deuxième partie montre les différentes expérimentations effectuées la discussion des résultats obtenus en comparant la performance de notre approche avec les métriques de sélection des features les plus courantes.

4.2 Description des ressources logicielles

4.2.1 Enivrements de développement

Dans la phase d'implémentation de notre approche nous avons utilisé Python version 3.8.5 comme langage de programmation, et Jupyter Notebook comme IDE, avec lesquels nous avons implémenté notre projet :

python : est un langage de programmation open source, puissant et facile à apprendre. Python est devenu ces dernières années le langage de programmation le plus employé par les informaticiens. Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. Python permet notamment aux développeurs de se concentrer sur ce qu'ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. Ainsi, développer du code avec Python est plus rapide qu'avec d'autres langages [10].
Les principales utilisations de Python par les développeurs sont :

— La programmation des applications .

— La création de services web.

— La génération de code.

Jupyter : est le dernier environnement de développement interactif basé sur le Web. Son interface flexible permet aux utilisateurs de configurer et d'organiser des flux de travail en science des données, en informatique scientifique, et en apprentissage automatique. Une conception modulaire invite les extensions à étendre et enrichir les fonctionnalités. [11]

4.2.2 Les bibliothèques nécessaires

Itmos-fs : de sélection de features écrite en Python [12].

Scikit-learn : est une bibliothèque clé pour le langage de programmation Python qui est généralement utilisée dans les projets d'apprentissage automatique. Scikit-learn se concentre sur les outils d'apprentissage automatique, y compris les algorithmes mathématiques, statistiques et à usage général qui constituent la base de nombreuses technologies d'apprentissage automatique. En tant qu'outil gratuit, Scikit-learn est extrêmement important dans de nombreux types de développement d'algorithmes pour l'apprentissage automatique et les technologies associées[13].

Pandas : est un outil d'analyse et de manipulation de données open source rapide, puissant, flexible et facile à utiliser, construit sur le langage de programmation Python [14].

Matplotlib : est une bibliothèque complète pour créer des visualisations statiques, animées et interactives en Python. Matplotlib rend les choses faciles et les choses difficiles possibles [15].

Nltk : est une plate-forme leader pour la création de programmes Python pour travailler avec des données du langage naturel. Il fournit des interfaces faciles à utiliser à plus de 50 corpus et ressources lexicales telles que WordNet, ainsi qu'une suite de bibliothèques de traitement de texte pour la classification, la tokenisation, la radicalisation, le balisage, l'analyse et le raisonnement sémantique, etc [16].

4.3 Démarche expérimental

Après avoir défini les grandes lignes de la démarche méthodologique que nous avons nommée CTME (co-occurrents termes avec minimum entropie) au chapitre 3, nous allons maintenant aborder le processus de sa mise en pratique :

4.3.1 Base d'apprentissage et de test

Pour évaluer notre approche, nous avons utilisé des ensembles de données avec des tailles et des niveaux de complexité variables. Les trois datasets que nous avons utilisés dans cette section sont décrits comme suit :

1) Movie Reviews : : contient 1 000 critiques positives et 1 000 critiques négatives collectées sur imdb.com et utilisées pour l'analyse des sentiments. La première version a été publiée en 2002 et la version mise à jour a été publiée en 2004, appelée "polarity dataset v2.0" [41]. La figure 4.1 montre un aperçu du dataset en utilisant la bibliothèque Pandas :

	text	category
0	plot : two teen couples go to a church party ,...	0
1	the happy bastard ' s quick movie review damn ...	0
2	it is movies like these that make a jaded movi...	0
3	" quest for camelot " is warner bros . ' first...	0
4	synopsis : a mentally unstable man undergoing ...	0
...	...	...
1995	wow ! what a movie . it ' s everything a movie...	1
1996	richard gere can be a commanding actor , but h...	1
1997	glory -- starring matthew broderick , denzel w...	1
1998	steven spielberg ' s second epic film on world...	1
1999	truman (" true - man ") burbank is the perfe...	1

2000 rows × 2 columns

FIGURE 4.1 – Le dataset Movie Reviews.

2) SMS Spam : est un ensemble de messages SMS étiquetés, collectés pour la recherche. Il contient 5 574 messages en anglais, étiquetés selon qu'ils sont Ham (légitime) ou Spam (illégitime) [28].). La figure 4.2donne un aperçu sur le contenu de le dataset SMS Spam.

	category	text
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...
...	...	...
5567	spam	This is the 2nd time we have tried 2 contact u...
5568	ham	Will Ì_ b going to esplanade fr home?
5569	ham	Pity, * was in mood for that. So...any other s...
5570	ham	The guy did some bitching but I acted like i'd...
5571	ham	Rofl. Its true to its name

5572 rows × 2 columns

FIGURE 4.2 – Le dataset SMS SPAM.

3) 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.

	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 – dataset Fake News

4.3.2 Prétraitement

Comme nous l'avons déjà expliqué dans le chapitre 3, le prétraitement des datasets passe par plusieurs étapes. Nous expliquerons ces étapes dans ce qui suit :

1. **Tokenization :** À l'aide de la boîte à outils de traitement du langage naturel (NLTK) pour Python, nous tokenisons les textes afin d'extraire les mots uniques du vocabulaire.
2. **Suppression des ponctuations**
3. **Suppression des Tags HTML**
4. **Suppression des mots vides**
5. **Suppression des stop-words**
6. **Suppression des chiffres**
7. **Transfert des majuscules en minuscules**

Les figures suivantes présentent les trois datasets après prétraitement.

	text	category	text_clean
0	plot : two teen couples go to a church party ,...	0	plot two teen couple church party drink drive ...
1	the happy bastard ' s quick movie review damn ...	0	happy bastard quick movie review damn bug get ...
2	it is movies like these that make a jaded movi...	0	movie like make jaded movie viewer thankful in...
3	" quest for camelot " is warner bros . ' first...	0	quest camelot warner bros first feature length...
4	synopsis : a mentally unstable man undergoing ...	0	synopsis mentally unstable man undergo psychot...
...	...	...	...
1995	wow ! what a movie . it ' s everything a movie...	1	wow movie everything movie funny dramatic inte...
1996	richard gere can be a commanding actor , but h...	1	richard gere commanding actor always great fil...
1997	glory -- starring matthew broderick , denzel w...	1	glory star matthew broderick denzel washington...
1998	steven spielberg ' s second epic film on world...	1	steven spielberg second epic film world war un...
1999	truman (" true - man ") burbank is the perfe...	1	truman true man burbank perfect name jim carre...

2000 rows × 3 columns

FIGURE 4.4 – Le dataset Movie Reviews après prétraitement.

	category	text	text_clean
0	ham	Go until jurong point, crazy.. Available only ...	jurong point crazy available bugis great world...
1	ham	Ok lar... Joking wif u oni...	lar joking wif oni
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	free entry wkly comp win cup final tkts may te...
3	ham	U dun say so early hor... U c already then say...	dun say early hor already say
4	ham	Nah I don't think he goes to usf, he lives aro...	nah think usf live around though
...	...	...	...
5567	spam	This is the 2nd time we have tried 2 contact u...	time try contact win pound prize claim easy ca...
5568	ham	Will Ì_ b going to esplanade fr home?	esplanade home
5569	ham	Pity, * was in mood for that. So...any other s...	pity mood suggestion
5570	ham	The guy did some bitching but I acted like i'd...	guy bitching act like interested buy something...
5571	ham	Rofl. Its true to its name	rofl true name

5572 rows × 3 columns

FIGURE 4.5 – Le dataset SMS SPAM après prétraitement.

	text	category	text_clean
0	Says the Annies List political group supports ...	1	say annies list political group support third ...
1	When did the decline of coal start? It started...	0	decline coal start start natural gas take star...
2	Hillary Clinton agrees with John McCain "by vo...	0	hillary clinton agree john mccain vote give ge...
3	Health care reform legislation is likely to ma...	1	health care reform legislation likely mandate ...
4	The economic turnaround started at the end of ...	0	economic turnaround start end term
...	...	...	...
10235	There are a larger number of shark attacks in ...	0	large number shark attack florida case voter f...
10236	Democrats have now become the party of the [At...	0	democrat become party atlanta metro area black
10237	Says an alternative to Social Security that op...	0	say alternative social security operate galves...
10238	On lifting the U.S. Cuban embargo and allowing...	1	lift cuban embargo allow travel cuba
10239	The Department of Veterans Affairs has a manua...	1	department veteran affair manual tell veteran ...

10240 rows × 3 columns

FIGURE 4.6 – Le dataset FAKE NEWS après prétraitement.

4.3.3 classification sans et avec métriques de sélection IG, MI, CH2 et DF

L'objectif des expérimentations est de tester la performance de notre approche de sélection des features CTME (co-occurrents matrice avec minimum entropie) avec celles des métriques les plus courantes (IG, MI, CH2, et DF) en termes du score F1 obtenu à partir des algorithmes de classification SVM et NB.

1) Classification sans sélection de features

Dans le cadre de notre travail, et pour voir l'effet de sélection des features sur la performance des algorithmes de classification, nous avons d'abord effectué une classification sans sélection. Les résultats obtenus en termes du score F1 sont montrés sur le Tableau 4.1. Les algorithmes de classification choisis sont NB et SVM car ils sont reconnus comme ayant de bons résultats dans la tâche de classification des textes [42].

DATASET	F1 score	
	SVM	NB
Movie reviews	0.818182	0.795455
SMS Spam	0.980424	0.957586
Fake News	0.578107	0.606213

TABLE 4.1 – Résultats de classification sans sélection de features.

2) Classification avec sélection des features par IG, MI, CH2 et DF

Comme nous l'avons déjà vu au premier chapitre, il existe plusieurs métriques de sélection des features, nous avons choisi Information Gain (IG), Mutual Information (MI), CH-Square (CH2),et Document-Frequency (DF) pour faire la classification avec les mêmes classifieurs NB et SVM. Les résultats de classification sont illustrés dans les tableaux suivants :

DATASET	**Movie reviews**			
Classifieur	**SVM**		**NB**	
	Features	**best score**	**Features**	**best score**
IG	800	0.863636	600	0.850000
MI	21800	0.836364	29400	0.800000
CH2	6600	*0.881818*	6200	*0.887879*
DF	2600	0.836364	1800	0.807576

TABLE 4.2 – Résultats classification avec sélection de features pour le dataset Movie Reviews.

DATASET	**SMS Spam**			
Classifieur	**SVM**		**NB**	
	Features	**best score**	**Features**	**best score**
IG	6200	0.979880	*1000*	*0.972268*
MI	5600	0.980424	3600	0.960305
CH2	6600	*0.978793*	2200	*0.970092*
DF	*2400*	*0.980968*	2200	0.970092

TABLE 4.3 – Résultats de classification avec sélection de features pour le dataset SMS Spam.

DATASET	**Fake News**			
Classifieur	**SVM**		**NB**	
	Features	**best score**	**Features**	**best score**
IG	1900	0.671006	*1900*	*0.681953*
MI	8200	0.583136	8400	0.610355
CH2	*4100*	0.689645	*2200*	*0.691420*
DF	*200*	*0.595858*	1400	0.615089

TABLE 4.4 – Résultats de classification avec sélection de features pour le dataset Fake News .

— Pour Movie Reviews, Le tableau 4.2 montre que les meilleurs scores ont été réalisés par CH2, où SVM a achevé un score F1 est égal à **0.881818** et NB a achevé un score F1 est égal à **0.887879**.

— Pour SMS Spam, comme indiqué sur le Tableau 4.3, NB a réalisé un meilleur score est égal **0.972268** pour IG. Tandis que SVM a réalisé un meilleur score est égal à **0.980968** pour DF.

— Pour Fake News, CH2 a prouvé aussi son efficacité, où un score est égal à **0.689645** a été réalisé par SVM. Tandis que NB a réalisé un score un peu plus

élevé est égal à **0.691420**.

4.3.4 Implémentation de CTME et comparaison des résultats

Nous avons implémenté notre approche CTME, où nous avons calculé la matrice de cooccurrence, l'entropie conditionnel correspondante à chaque terme du vocabulaire, après nous avons calculé le score de chaque terme en utilisant les fréquences de cooccurrence des termes et leurs entropies conditionnelles comme mentionné au chapitre précédent.

La méthode CTME (co-occurrents terms avec minimum entropie) que nous avons proposée est basée sur la combinaison entre la matrice de cooccurrence et l'entropie conditionnelle. Pour tester la performance de CTME, les scores F1 des classifieurs obtenus après sélection des features avec CTME doivent être comparés à ceux obtenus par les mêmes classifieurs après sélections des features avec les méthodes communes IG, MI, CH2 et DF.

Pour cela, nous avons effectué une classification en utilisant les mêmes algorithmes de classification SVM et NB sur les trois datasets pour évaluer les performances de la méthode proposée.

Les résultats de classification avec la méthode proposée CTME sont indiqués sur le Tableau 4.6 .

	CTME			
Classifieur	**SVM**		**NB**	
	Features	**best score**	**Features**	**best score**
Movie Reviews	6000	0.906061	*6800*	*0.903030*
SMS Spam	6100	0.979880	1100	0.970636
Fake News	*3400*	0.682840	*1600*	*0.689053*

TABLE 4.5 – Résultats de classification obtenus après sélection de features par CTME.

D'après les résultats du tableau 4.5, il est évident de constater que notre approche CTME est nettement meilleure comparée aux métriques IG, MI, CH2 et DF. Les résultats obtenus montrent clairement la performance de notre approche en termes de sélection des features pertinents où les meilleurs scores F1 de SVM et NB ont été réalisés pour notre approche.

En conclusion, nous peut dire que l'approche est très effective en matière de sélection de termes pour la catégorisation de textes.

Pour les trois datasets, les figures suivantes présentent les scores F1 obtenus par SVM et NB. Les features (termes) sont ordonnés selon les scores de chaque méthode IG, MI, CH2, DF et notre méthode CTME, de plus haut au plus bas, où les algorithmes de classifications sont exécutés itérativement avec un seuil de 100, c'est-à-dire 100 termes sont ajoutés à chaque itération et le score F1 est recalculé.

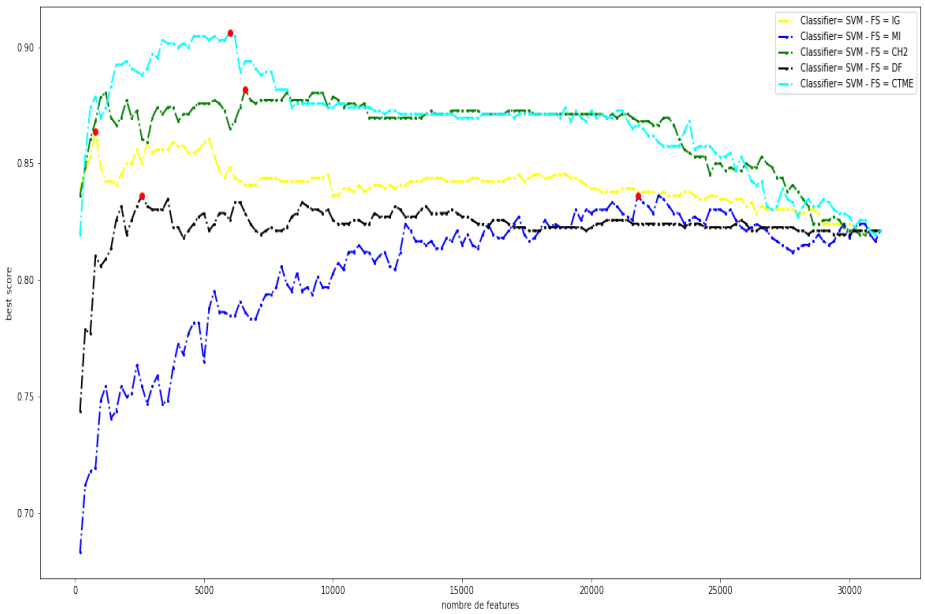

FIGURE 4.7 – Résultats de classification de Movie Reviews avec SVM.

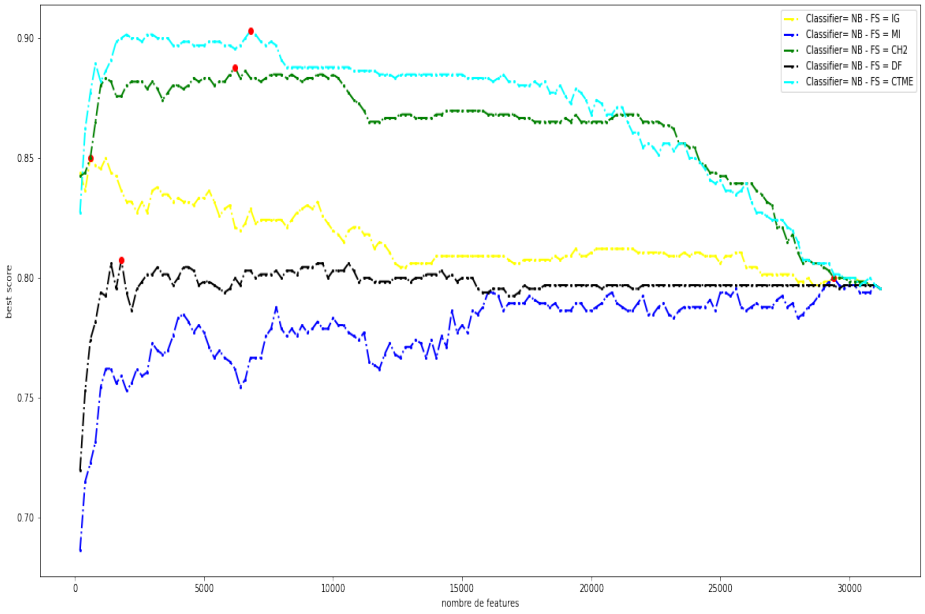

FIGURE 4.8 – Résultats de classification de Movie Reviews avec NB.

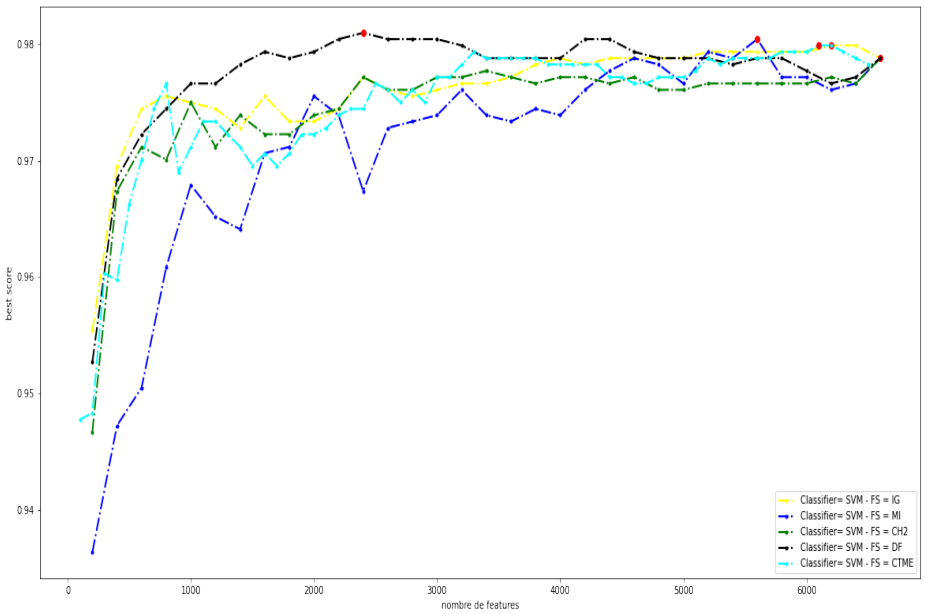

FIGURE 4.9 – Résultats de classification de SMS Spam avec SVM.

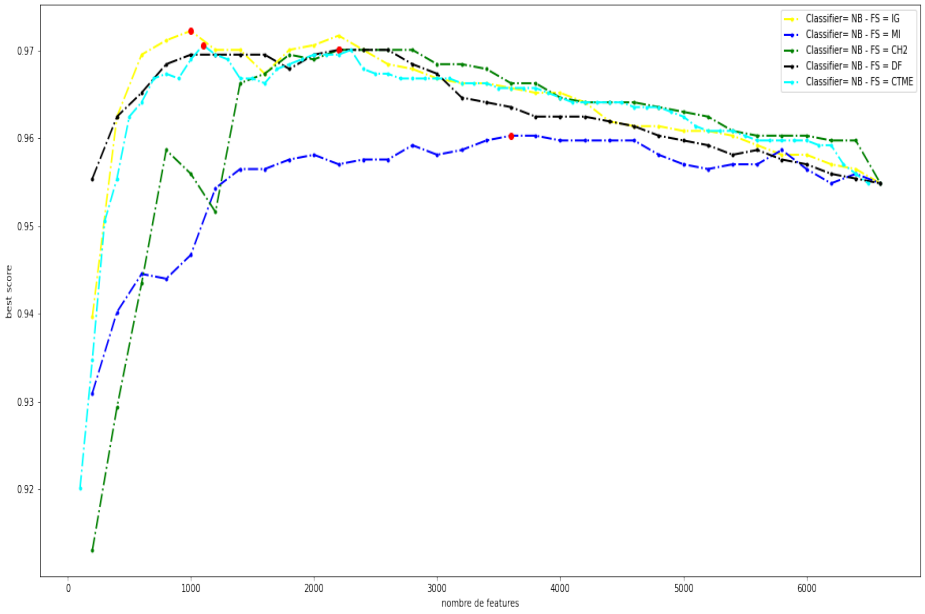

FIGURE 4.10 – Résultats de classification de SMS Spam avec NB.

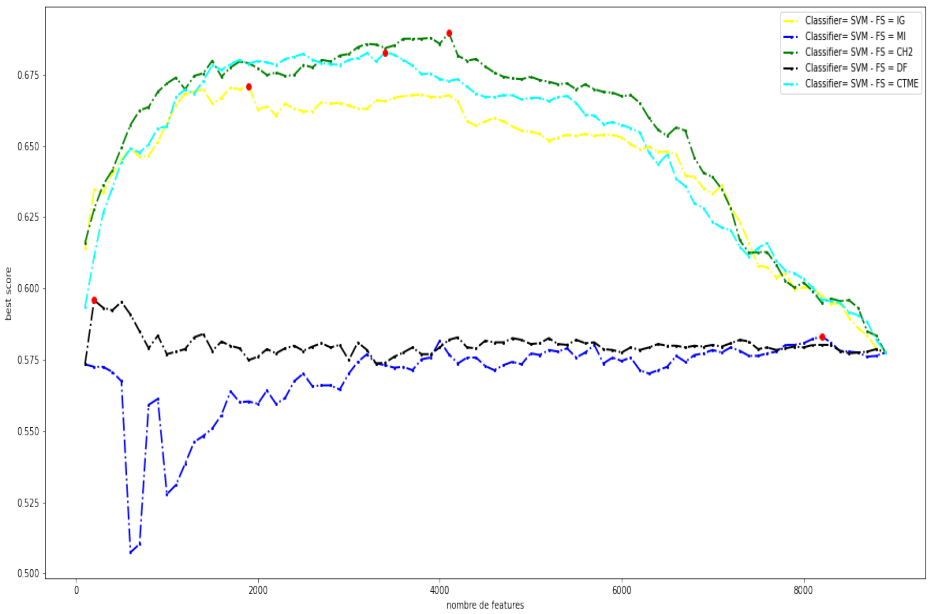

FIGURE 4.11 – Résultats de classification de Fake News avec SVM.

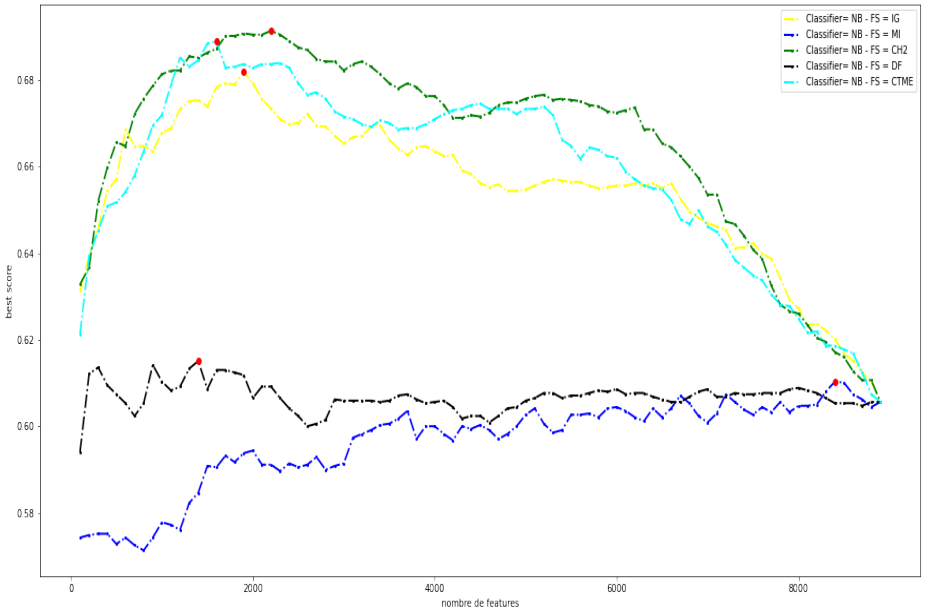

FIGURE 4.12 – Résultats de classification de Fake News avec NB.

Les expérimentations ont montré que notre méthode proposé CTME a obtenu des performances considérables en termes des scores F1 score achevées par les deux classifieurs SVM et NB, avec un nombre minimal de features (termes) comparée avec

IG, MI, CH2 et DF.

La performance de notre méthode peut être interprétée comme suit :

— Les termes qui apparaissent fréquemment dans le même contexte peuvent avoir une corrélation sémantique très élevée, d'où une capacité de discrimination élevée par rapport aux termes évalués séparément.

— Les fréquences de cooccurrence sont calculées au niveau de chaque catégorie pour prendre en considération le biais des termes, où un terme est considéré comme pertinent s'il apparaît fréquemment avec d'autres termes dans la même catégorie, et rarement dans les autres catégories, qui devrait avoir une entropie conditionnelle minimale.

— L'évaluation des termes par les méthodes statistiques comme IG, MI, CH2 et DF en se basant sur le nombre de documents dans lesquels apparaît le terme, peut conduire à sélectionner des termes redondants, parce qu'elles ne prennent pas en considération si les termes sont biaisés ou non.

4.4 conclusion

Dans le dernier chapitre de notre projet, nous avons implémenté notre approche qui répond parfaitement aux objectifs fixés au début.

Notre méthode de sélection de features CTME est basée sur la combinaison de la matrice de cooccurrence et la mesure d'entropie conditionnelle, pour capturer la sémantique des termes les plus représentatifs qui conduisent à une meilleure performance de classification. Les résultats ont montré que CTME a prouvé son efficacité en capturant les termes les plus informatifs pour la classification des textes, comparée aux méthodes de sélection les plus largement utilisées dans littérature IG, MI, CH2 et DF.

CONCLUSION GÉNÉRALE

La classification des textes est un sujet de recherche populaire qui a attiré l'attention des chercheurs où plusieurs travaux ont été publié dans conférencesetrevues scientifiques de l'intelligence artificielle et l'exploration des données.

Dans les problèmes de classification de textes, la sélection des features est la recherche d'un sous-ensemble de mots les plus pertinents, qui permet de réduire la dimension de l'ensemble complet des features et conduit à une meilleure performance des algorithmes de classification.

Beaucoup de métriques de sélection qui ont prouvé leur efficacité pour la sélection des features comme IG, MI, Chi2, DF, etc., ne prennent pas la corrélation entre les mots, ni leur fréquence de cooccurrence dans le même contexte et traitent les mots comme des entités isolées où l'information sémantique pertinente partagée par les mots coexistants n'est pas considérée.

Pour remédier à ce problème, et en vue de prendre en considération l'information sémantique, nous avons proposé une méthode de sélection des features basée sur l'idée que les termes qui existent fréquemment dans le même contexte sont plus informatifs que les termes isolés.
Le schéma de pondération des mots (termes) prend en considération la fréquence de cooccurrence du mot et son entropie conditionnelle par rapport à la variable classe, où les termes pertinents sont ceux qui existent fréquemment avec d'autres termes et capables de réduire l'entropie de la variable classe.

Les expérimentation en utilisant deux classifieurs NB et SVM sur trois datasets, ont prouvé l'efficacité de notre méthode dans la plupart des cas en la comparant avec les métriques les plus populaire : IG, MI, Chi2 et DF.

BIBLIOGRAPHIE

[1] http://theses.univ-lyon2.fr/documents/lyon2/2004/legrand_g/pdfAmont/legrand_g_chapitre02.pdf.. (dernier accès : 02/03/2022.)

[2] https://bookstack.apmechev.com/books/machine-learning/page/feature-engineeringtransformationselection.

[3] http://scikit-learn.org/stable/modules/svm.html..

[4] https://www.researchgate.net/figure/260397165_fig7_pseudocode-for-knn-classification,.

[5] https://saravananthirumuruganathan.wordpress.com/2010/05/17/a-detailed-introduction-to-k-nearest-neighbor-knn-algorithm/.

[6] https://en.wikipedia.org/wiki/k-nearest_neighbors_algorithm.

[7] http://www-ia.lip6.fr/~tollaris/articles/these/node7.html.

[8] https://fr.wikipedia.org/wiki/tf-idf. (dernier accès : 02/05/2022.)

[9] Co_location_final_V3.pdf(oslomet.no). (dernier accès : 02/05/2022.)

[10] https://scikit-learn.org/stable/index.html.. (dernier accès : 03/05/2022.)

[11] https://jupyter.org/. (dernier accès : 02/05/2022.)

[12] https://pypi.org/project/ITMO-FS/. (dernier accès : 02/05/2022.)

[13] https://www.techopedia.com/definition/33860/scikit-learn. (dernier accès : 02/05/2022.)

[14] https://pandas.pydata.org/. (dernier accès : 02/05/2020.)

[15] https://matplotlib.org/. (dernier accès : 02/05/2022.)

[16]]https://www.nltk.org/. (dernier accès : 02/05/2022.)

[17] Hadeer AHMED, Issa TRAORE et Sherif SAAD. « Detecting opinion spams and fake news using text classification ». In : *Security and Privacy* 1.1 (2018), e9.

[18] Ahmad Abu AL HAIJA et Mai VU. « An asymptotically capacity-achieving scheme for the Gaussian relay channel with relay-destination cooperation ». In : *2013 47th Annual Conference on Information Sciences and Systems (CISS)*. IEEE. 2013, p. 1-6.

[19] Mehdi ALLAHYARI et al. « A brief survey of text mining : Classification, clustering and extraction techniques ». In : *arXiv preprint arXiv :1707.02919* (2017).

[20] Nouman AZAM et JingTao YAO. « Comparison of term frequency and document frequency based feature selection metrics in text categorization ». In : *Expert Systems with Applications* 39.5 (2012), p. 4760-4768.

[21] Soraya AZZOUG. « Sélection automatique des caractéristiques pour la reconnaissance des chiffres manuscrits par la méthode F-score ». Thèse de doct. Faculté d'Electronique et d'Informatique, 2013.

[22] Billal BELAININE. « Classification supervisée de textes courts et bruités : application au domaine des médias sociaux ». In : (2017).

[23] Nicolas BOURGEOIS et al. « Search for meaning through the study of co-occurrences in texts ». In : *International work-conference on artificial neural networks*. Springer. 2015, p. 578-591.

[24] Manoranjan DASH et Huan LIU. « Feature selection for classification ». In : *Intelligent data analysis* 1.1-4 (1997), p. 131-156.

[25] WE DIETZ, EL KIECH et Moonis ALI. « Classification of data patterns using an autoassociative neural network topology ». In : *IEA/AIE-89*. 1989.

[26] Kai-Yan FENG, Yu-Dong CAI et Kuo-Chen CHOU. « Boosting classifier for predicting protein domain structural class ». In : *Biochemical and biophysical research communications* 334.1 (2005), p. 213-217.

[27] George FORMAN et al. « An extensive empirical study of feature selection metrics for text classification. » In : *J. Mach. Learn. Res.* 3.Mar (2003), p. 1289-1305.

[28] José María GÓMEZ HIDALGO et al. « Content based SMS spam filtering ». In : *Proceedings of the 2006 ACM symposium on Document engineering*. 2006, p. 107-114.

[29] Bernard GOSSELIN. « Application de réseaux de neurones artificielsa la reconnaissance automatique de caracteres manuscrits ». In : *Faculté polytechnique de Mons* 231 (1996).

[30] Isabelle GUYON et André ELISSEEFF. « An introduction to variable and feature selection ». In : *Journal of machine learning research* 3.Mar (2003), p. 1157-1182.

[31] Veneta HARALAMPIEVA et Gavin BROWN. « Evaluation of Mutual information versus Gini index for stable feature selection ». In : (2016).

[32] Hassane HILALI. « Application de la classification textuelle pour l'extraction des règles d'association maximales ». Thèse de doct. Université du Québec à Trois-Rivières, 2009.

[33] 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.

[34] Daniel T LAROSE. « An introduction to data mining ». In : *Traduction et adaptation de Thierry Vallaud* (2005).

[35] Jundong LI et al. « Feature selection : A data perspective ». In : *ACM computing surveys (CSUR)* 50.6 (2017), p. 1-45.

[36] Jundong LI et al. « J. Tang ja H. Liu,"Feature Selection" ». In : *ACM Computing Surveys* 50 (), p. 1-45.

[37] Joseph LILLEBERG, Yun ZHU et Yanqing ZHANG. « Support vector machines and word2vec for text classification with semantic features ». In : *2015 IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*. IEEE. 2015, p. 136-140.