République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de Bordj Bou Arréridj

Faculté des mathématiques et d'informatiques

Département Informatique

MÉMOIRE

Présenté en vue de l'obtention du diplôme

Master en informatique

Spécialité : Technologie de *l'Information et de la communication*

Thème :

Représentation Hybride pour la classification des documents

Présenté par :

- **BELAZZOUG Louiza**
- **SAOUD Nour el houda**

Président : Mr : Abdelhamid SAIFI MAA à L'U.El Bachir El Ibrahimi-BBA.

Examinatrice : Mm : Ramla BLALTA MAA à L'U.El Bachir El Ibrahimi-BBA

Encadrant : Dr : BELAZZOUG Mouhoub MCA à L'U.El Bachir El Ibrahimi-BBA

Promotion : 2020/ 2021.

REMERCIEMENTS

On remercie dieu le tout puissant de nous avoir donné la force et la patience et la volonté d'entamer et de terminer ce mémoire.

Je tiens à remercie très chaleureusement Dr. BELAZZOUG Mouhoub qui nous a permis de bénéficier de son encadrement. Les conseils qu'il nous a prodigué, la patience, la confiance qu'il nous a témoignés ont été déterminants dans la réalisation de notre travail de recherche.

Nos remerciements s'adressent également à tous les membres du département d'Informatique, enseignants et administratifs, pour leurs générosités et la grande patience dont ils ont su faire preuve malgré leurs charges académiques et professionnelles.

Merci pour tous ceux qui, m'ont aidé de près ou de loin à réaliser ce travail.

DÉDICACES

Je dédie ce mémoire

A la mémoire de mon oncle BENBOUZID Djameleddine, paix à son âme

A mes chers parents ma mère Naima et mon père Kameleddine
Pour leur patience, leur amour, leur soutien et leurs Encouragements.

A mes sœurs et mes frères

A ma cousine Kaouther et mes nièces Doua et Radja

A mon cher mari

A tous mes amis et a tous mes camardes Sans oublier tous mes professeurs

Et à tous ceux qui m'ont aidé dans l'élaboration De ce travail

Louiza

DÉDICACES

Je dédie ce mémoire

A mes chers parents ma mère et mon père Pour leur patience, leur amour, leur soutien et leurs Encouragements.

A ma sœur Khaoula et mes frères Ahmed et Yousef

A mon cher mari BENDAOUED Ali

A ma nièce Rim

A tous mes amis et a tous mes camardes Sans oublier tous mes professeurs

Et à tous ceux qui m'ont aidé dans l'élaboration De ce travail

Nour

Résumé

Le travail effectué dans ce mémoire se situe dans le domaine de la fouille de texte et en particulier la classification automatique de textes. Le projet étudié le problème de catégorisation de textes de sports comme étant objectifs ou subjectifs. Pour cela nous avons suivi une représentation hybride des documents textuels qui contient le modèle sac de mots et morpho-syntactique à la fois. Pour la phase de classification nous avons appliqué ‘algorithme Naive Bayes combiné avec des algorithmes de recherche tel que Gras et PSO pour en servir à la réduction de la dimensionnalité.

Abstract

The work carried out in this thesis is in the field of text mining and in particular the automatic classification of texts. The project investigated the problem of categorizing sports texts as objective or subjective. For this we have followed a hybrid representation of textual documents which contains both the word bag and the morph-syntactic model. For the classification phase we applied ‘Naive Bayes algorithm combined with search algorithms such as Gras and PSO to be used for dimensionality reduction.

ملخص

العمل الذي تم تنفيذه في هذه الرسالة هو في مجال التنقيب عن النص وخاصة التصنيف الآلي للنصوص. حقق المشروع في مشكلة تصنيف النصوص الرياضية على أنها موضوعية أو ذاتية. لهذا، اتبعنا تمثيلًا هجينًا للوثائق النصية التي تحتوي على كل من حقيبة الكلمات والنموذج الصرفي النحوي. بالنسبة لمرحلة التصنيف، طبقنا "خوارزمية Naive Bayesجنبًا إلى جنب مع خوارزميات البحث مثل Gras و PSO لاستخدامها في تقليل الأبعاد.

Table des matières :

Liste des figures

Liste des tableaux

Liste d'abréviations

TC : Text Categorization

FS : Feature selection

IG : Information Gain

GA : Genetical algorithms

LSA : Latent semanticanalysis

PCA : principal component analysis

TF : Term Frequency

IDF : Inverse Document Frequency

TF-IDF : Term Frequency- Inverse Document Frequency

PSO : Particle swarm optimization

SVM : Support Vector Machines

SFS : Sequential Forward Selection

SBS : Sequential Backward Selection

TS : Tabou Search

SCA : Sine Cosine Algorithm

ACO : Ant Colony Optimisation

MFO : Moth-Flame optimization

Introduction générale

Introduction Générale

- ✓ **Contexte de travail et problématique:**

Grâce à la généralisation d'utilisation du réseau mondial INTERNET dans les différents secteurs de la vie, et suite à l'expansion rapide du web et des réseaux sociaux, nous disposons actuellement de gros volumes de données de différents types qui sont représentés sous forme électronique. Ces données renferment souvent des connaissances précieuses qui peuvent être d'une grande utilité pratique. Cependant la nature son structurée et le volume important de ces données rendent l'accès à ces connaissance par les moyens classiques une tâche très difficile, voire impossible. Le besoin de développer des méthodes intelligentes pour l'accès à de telles connaissances représente donc un défi majeur et constitue le domaine appelé fouille de donnée.

Nous nous plaçons dans ce cadre et nous nous intéressons en particulier aux données textuelles. Plus précisément, nous considérons le problème de classification ou de catégorisation de textes écrits en langage naturel, c'est-à-dire d'associer à un texte une classe ou une catégorie parmi un ensemble de catégories prédéfinies. L'approche suivieest fondée sur l'apprentissage supervisé. Dans cette approche, on s'appuie sur une collection de textes (échantillon d'apprentissage) préalablement étiquetés où à chaque texte est associée sa catégorie pour construire un modèle qui permet d'expliciter la relation cachée entre les entrées (textes) et les sorties du système (catégorie). Ce modèle est ensuite validé en l'appliquant sur une deuxième collection de textes préalablement étiquetés. L'idée étant de vérifier si le modèle permet de prédire les bonnes catégories pour des textes qui n'ont pas servi à son apprentissage.

Il se trouve que les textes bruts ne sont pas adaptés directement à une utilisation comme entrée aux algorithmes de classification. Une étape de prétraitement des textes est donc nécessaire avant de passer à l'étape de classification proprement dite. Cette étape permet d'effectuer des nettoyages et des traitements lexicaux et syntaxiques sur les mots du texte. Elle comporte aussi souvent une phase de réduction de dimensionnalité qui permet de ne fournir à l'étape de classification qu'un nombre réduit de termes qui sont jugés pertinents par rapport à l'objectif visé. Plusieurs approches ont été proposées pour la réduction de

dimensionnalité. Nous nous intéressons ici à celles dites de type Wrapper et qui sont basées sur des algorithmes de recherche.

- ✓ ***Contribution :***

Dans ce mémoire nous proposons d'effectuer une classification automatique d'articles sportifs où le but de la classification est de juger si un article donné est de caractère subjectif ou objectif. Pour cela nous avons suivi une représentation hybride des documents textuels qui contient le modèle sac de mots et morpho-syntactique à la fois. Pour la phase de classification nous avons appliqué 'algorithme KNN combiné avec des algorithmes de recherche tel que Gras et PSO pour en servir à la réduction de la dimensionnalité.

Organisation de la thèse :

La présente thèse se structure comme suit :

Le chapitre 1 : dans ce chapitre, on a présenté un aperçu sur les feuilles de données, ainsi que le processus de la classification de textes. Les algorithmes d'apprentissages. Il contient aussi une explication détaillée sur les collections de textes les plus utilisé dans la catégorisation automatique.

Le chapitre 2 : dans ce chapitre, nous présenterons tout d'abord la phase de prétraitement de textes(le processus du nettoyage), ensuite on a cité les méthodes de représentation de textes et ainsi que les mesures de pondération de termes. Comme Il présente un aperçu sur les approches et les méthodes de réduction de la dimensionnalité.

Le chapitre 3 nous allons introduire les différentes stratégies de recherche. Après on a représenté quelque modèles des algorithmes de recherche.

Le Chapitre 4 : est consacré à l'implémentation de notre application et à l'analyse des résultats obtenus des expérimentations que nous avons effectuées. Ces expérimentations concernent la combinaison

Enfin, le mémoire est terminé par une conclusion générale et des idées de perspectives pour des travaux futurs.

Chapitre I

Classification automatique de textes

1.1 Introduction

La surabondance des document (éléctronique, page, web)pas de nouveaux problèmes vis – a-vis de l'utilisateur final(entrprise, orgnisme, individu …etc.) qui n'est donc plus capable d'analyser ou d'appréhender ces information dans leurs globalité, l'information utile étant enfouie dans le texte, il devient indispensable de proposer de nouveaux systèmes permettant l'analyse l'organisation et la représentation des différents contenus textuels la fouille de textes est la solutions actuelle au problème de la surcharge informationnelle de type textuel. Le domaine de la fouille de textes(fidelia,2007)réunit et intègre dans ces applications des méthodes d'extractions d'informations de recherche d'informations de questions-réponses de résumé automatique ,de catégorisation de textes, de classification et de routage de documents textuels ainsi que le recours a des techniques de fouille de textes (CT) et l'extraction d'information (EI).

1.2 Présentation de l'exploration de données

Le développement des technologies de l'information a généré une grande quantité de bases de données et d'énormes données dans divers domaines. La recherche dans les bases de données et les technologies de l'information a donné lieu à une approche de stockage et manipuler ces précieuses données pour une prise de décision ultérieure. L'exploration de données est un processus d'extraction de des informations et des modèles utiles à partir de données énormes. Il est également appelé processus de découverte des connaissances, l'exploration de connaissances à partir de données, l'extraction de connaissances ou l'analyse de données/modèles.

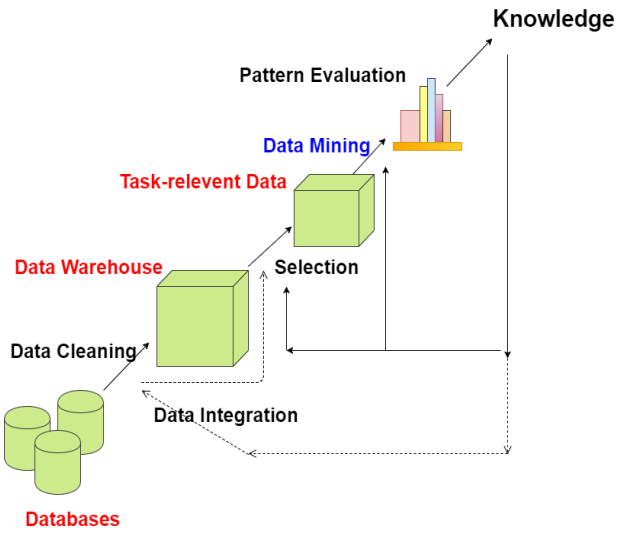

Figure1. 1 Processus de fouille de textes

L'exploration de données est un processus logique qui est utilisé pour rechercher dans une grande quantité de données afin de trouver données utiles. Le but de cette technique est de trouver des modèles qui étaient auparavant inconnus. Une fois ces des modèles sont trouvés, ils peuvent en outre être utilisés pour prendre certaines décisions pour le développement de leurs entreprises. Trois étapes impliquées sont • Exploration • Identification de modèle • Déploiement Exploration : Dans la première étape de l'exploration des données, les données sont nettoyées et transformées sous une autre forme, et les variables importantes et la nature des données en fonction du problème sont déterminées. Identification des modèles : une fois les données explorées, affinées et définies pour les variables spécifiques, la deuxième étape est de former une identification de modèle. Identifiez et choisissez les modèles qui font la meilleure prédiction. Déploiement : les modèles sont déployés pour le résultat souhaité.

1.2.1 Modèles d'exploration de données

De nombreuses industries telles que la fabrication, le marketing, la chimie et l'aérospatiale tirent parti de l'exploration de données. Ainsi, la demande de processus d'exploration de données standard et fiables augmente considérablement. Les modèles d'exploration de données importants comprennent:

Processus standard intersectoriel pour l'exploration de données (CRISP-DM) CRISP-DM est un modèle d'exploration de données fiable composé de six phases. Il s'agit d'un processus cyclique qui fournit une approche structurée du processus d'exploration de données. Les six phases peuvent être mises en œuvre dans n'importe quel ordre, mais cela nécessiterait parfois de revenir aux étapes précédentes et de répéter les actions.

Les six phases de CRISP-DM comprennent :

1. **Compréhension de l'entreprise** : Dans cette étape, les objectifs des entreprises sont définis et les facteurs importants qui aideront à atteindre l'objectif sont découverts.
2. **Compréhension des données :** cette étape collectera toutes les données et les remplira dans l'outil (si vous utilisez un outil). Les données sont répertoriées avec leur source de données, leur emplacement, la manière dont elles sont acquises et si un problème est rencontré. Les données sont visualisées et interrogées pour vérifier leur exhaustivité.

3 **Préparation des données :** cette étape implique la sélection des données appropriées, le nettoyage, la construction d'attributs à partir des données, l'intégration des données de plusieurs bases de données.

4 **Modélisation :** la sélection de la technique d'exploration de données telle que l'arbre de décision, générer une conception de test pour évaluer le modèle sélectionné, construire des modèles à partir de l'ensemble de données et évaluer le modèle construit avec des experts pour discuter du résultat est effectuée dans cette étape

5 **Évaluation :** Cette étape déterminera dans quelle mesure le modèle résultant répond aux exigences de l'entreprise. L'évaluation peut se faire en testant le modèle sur des applications réelles. Le modèle est examiné pour toute erreur ou étape qui devrait être répétée.

6 **Déploiement :** dans cette étape, un plan de déploiement est élaboré, une stratégie pour surveiller et maintenir les résultats du modèle d'exploration de données pour vérifier son utilité est formée, des rapports finaux sont établis et un examen de l'ensemble du processus est effectué pour vérifier toute erreur et voir si toute étape est répétée.

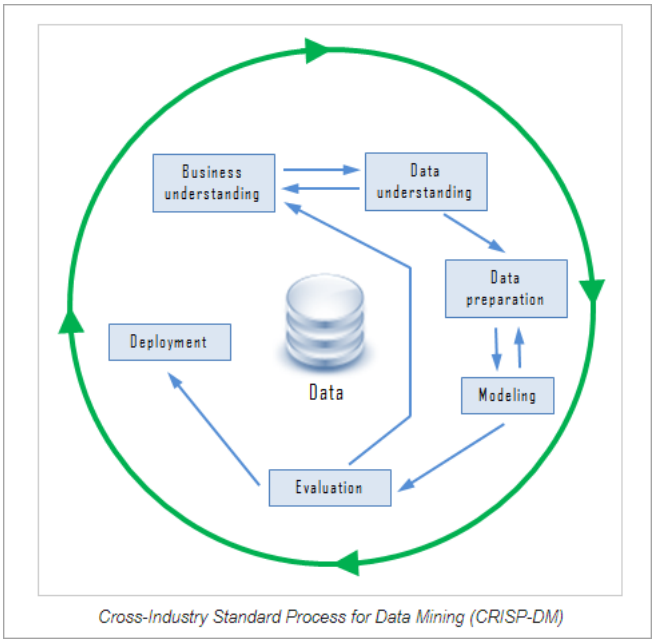

Figure1. 2 Les modèles de Data Mining

1.2.2 Étapes du processus d'exploration de données

Le processus d'exploration de données est divisé en deux parties, à savoir le prétraitement des données et l'exploration de données. Le prétraitement des données implique le nettoyage des données, l'intégration des données, la réduction des données et la

transformation des données. La partie d'exploration de données effectue l'exploration de données, l'évaluation des modèles et la représentation des connaissances des données.

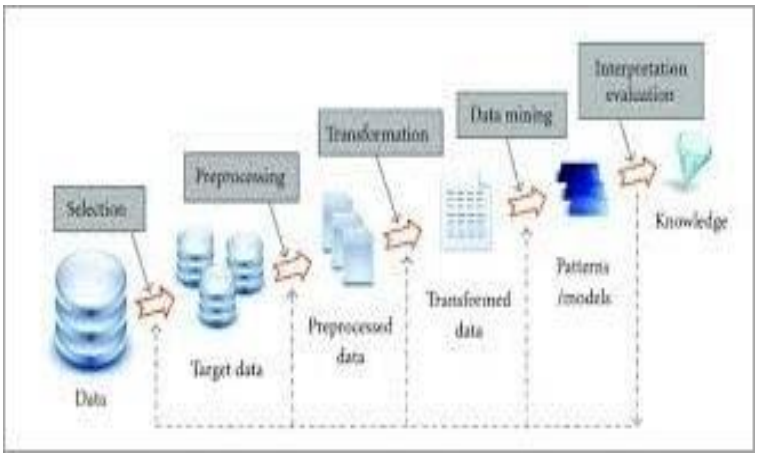

Figure1. 3 Les méthodes de Data Mining

De nombreux facteurs déterminent l'utilité des données, tels que l'exactitude, l'exhaustivité, la cohérence et l'actualité. Les données doivent être de qualité si elles répondent à l'objectif visé. Ainsi, le prétraitement est crucial dans le processus d'exploration de données. Les principales étapes du prétraitement des données sont expliquées ci-dessous.

1) Nettoyage des données Le nettoyage des données est la première étape de l'exploration de données. Cela a de l'importance car des données sales, si elles sont utilisées directement dans l'exploitation minière, peuvent entraîner une confusion dans les procédures et produire des résultats inexacts. Fondamentalement, cette étape implique la suppression des données bruitées ou incomplètes de la collection. De nombreuses méthodes qui nettoient généralement les données par elles-mêmes sont disponibles, mais elles ne sont pas robustes.

2) Intégration des données Lorsque plusieurs sources de données hétérogènes telles que des bases de données, des cubes de données ou des fichiers sont combinées pour l'analyse, ce processus est appelé intégration de données. Cela peut aider à améliorer la précision et la vitesse du processus d'exploration de données. Différentes bases de données ont différentes conventions de nommage des variables, en provoquant des redondances dans les bases de données. Un nettoyage supplémentaire des données peut être effectué pour supprimer les redondances et les incohérences de l'intégration des données sans affecter la fiabilité des données. L'intégration de données peut être effectuée à l'aide d'outils de migration de données tels qu'Oracle Data Service Integrator et Microsoft SQL, etc.

3) Réduction des données Cette technique est appliquée pour obtenir des données pertinentes pour l'analyse à partir de la collecte de données. La taille de la représentation est beaucoup plus petite en volume tout en maintenant l'intégrité. La réduction des données est effectuée à l'aide de méthodes telles que Naive Bayes, les arbres de décision, le réseau de neurones, etc. Certaines stratégies de réduction des données sont :

- Réduction de la dimensionnalité : réduire le nombre d'attributs dans l'ensemble de données.
- Réduction de la numérotation : remplacement du volume de données d'origine par des formes plus petites de représentation des données.
- Compression de données : représentation compressée des données d'origine

4) Transformation des données Dans ce processus, les données sont transformées en une forme adaptée au processus d'exploration de données. Les données sont consolidées afin que le processus d'extraction soit plus efficace et que les modèles soient plus faciles à comprendre. La transformation des données implique le mappage des données et le processus de génération de code. Les stratégies de transformation des données sont:

- Lissage : suppression du bruit des données à l'aide de techniques de clustering, de régression, etc.
- Agrégation : les opérations récapitulatives sont appliquées aux données.
- Normalisation : mise à l'échelle des données pour qu'elles se situent dans une plage plus petite.
- Discrétisation : Les valeurs brutes des données numériques sont remplacées par des intervalles. Par exemple, l'âge.

5) Exploration de données L'exploration de données est un processus permettant d'identifier des modèles et des connaissances intéressants à partir d'une grande quantité de données. Dans ces étapes, des modèles intelligents sont appliqués pour extraire les modèles de données. Les données sont représentées sous forme de modèles et les modèles sont structurés à l'aide de techniques de classification et de regroupement.

6) Évaluation de modèle Cette étape consiste à identifier des modèles intéressants représentant les connaissances sur la base de mesures d'intérêt. Des méthodes de synthèse et de visualisation des données sont utilisées pour rendre les données compréhensibles par l'utilisateur.

7) Représentation des connaissances La représentation des connaissances est une étape où des outils de visualisation des données et de représentation des connaissances sont utilisés pour représenter les données extraites. Les données sont visualisées sous forme de rapports, de tableaux, etc.

1.3 Classification de textes

La catégorisation de texte est le processus de classification des textes et d'attribution de balises aux textes en langage naturel dans l'ensemble prédéterminé de catégories. Outre les API de classification manuelles et automatisées, elles sont également utilisées pour catégoriser les textes clés d'un document afin d'utiliser les mots importants. Cependant, l'API peut faire ce travail automatiquement, mais si vous cherchez à classer les textes pour l'apprentissage automatique ou l'IA, vous devriez utiliser manuellement les experts. Comme, l'API peut ne pas fonctionner correctement, ou ne pas réussir à classer les différents types de mots-clés dans un document. Normalement, la classification de texte peut être classée en différents types tels que supervisé et non supervisé. Dans le cas de l'exploration de texte, le processus de classe prédéfinie pour former un texte de traitement du langage naturel « inconnu » a été proposé par Moreno et Redondo, il a été représenté comme la fonction TC en une seule étiquette. La plupart des recherches présentées sur la classification incluent les algorithmes SVM, Naïve Bayes et KNN. Chaque classe ck C pendant la phase de test de la classification des textes. Afin de définir chaque document qui est représenté par dj où dj est le numéro du document. D représente le nombre total de documents. Cependant, afin de justifier plusieurs clusters prédéfinis à un texte « inconnu » tel que présenté par Feng et al. (2005). Cependant, il est très courant de la présenter comme la tâche TC multi-label, alors que tout nombre $0 < nj \leq |C|$ des classes peuvent être prédites pour chaque document $dj \in D$. « TC représente la classification de texte et son utilisation pour une étiquette unique » (Allahyari et al. 2017 qui est une classe distincte ni une classe prédéfinie ni son supplément à une » (Sebastiani, 2006). Dans le passé, plusieurs types de recherche ont été étudiés. Certaines difficultés peuvent s'opposer au processus de classification des textes, on site quelques difficultés sont :

- **Ambiguïté :** un mot peut avoir plusieurs synonymes différents et quelque définition, À cause de l'ambigüité, les mots sont parfois de mauvais descripteurs.
- **Sur apprentissage :** Le grand nombre des mots générés par l'étape de prétraitement entraînera un sur apprentissage pour les algorithmes de classification. C'est à dire

que les algorithmes sont bien classer dans la phase d'apprentissage, mais dans la phase de test il mal classer.

- **Synonymes :** Les synonymes sont des terme sou des phares qui signifient la même chose. Ne pas prendre en compte ces synonymes dans l'étape de représentation du document étendra davantage notre vecteur de mots et nous aurons donc une précision de performance médiocre.
- **Le graphe :** Un mot peut contenir une faute de frappe ou une erreur typographique, car il peut être écrit de plusieurs façons en majuscules. Cela affectera la qualité des résultats. Car si un terme est graphe de deux façons dans le même document (Ghelizane, Relizane), la simple recherche de ce mot avec une seule forme graphique ignorera la présence du même terme dans d'autres graphes.
- **Mots composés :** Ne vous souciez pas des mots composés comme : comme Arc-en-ciel, can, save, etc. Dont le numéro est très important dans toutes les langues et traiter le mot Arc-en-ciel comme 3 termes distincts par exemple réduit considérablement les performances cependant, un système de classification utilisant la technique n-gram pour l'encodage de texte a considérablement réduit ce problème de mot composé.

1.4 Les algorithmes de classification

Il existe plusieurs algorithmes dans l'apprentissage automatique et notamment la classification. 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. Cependant, et quant à la construction d'algorithmes on trouve généralement deux phases durant le processus d'apprentissage. Une qui est la phase d'entrainement dans laquelle on essaye de générer un modèle ou d'estimer une fonction pour relier une catégorie avec un document. La deuxième permet de décider pour chaque document à quelle catégorie il appartient. Par ailleurs, les algorithmes de classification pourraient avoir plusieurs inspirations ou types, à savoir : mathématiques, probabilistes, ensemblistes, floue, règles de décisions…etc. Dans les sous-sections suivantes on va présenter quelques algorithmes de classification qui sont appliqués dans la classification de textes ainsi très reconnus dans plusieurs domaines d'application de la classification.

1.4.1 Algorithme de Rocchio

Cette méthode est mise en œuvre pour la catégorisation et l'efficacité, et textes ne peuvent pas être appartenant à une seule catégorie. Toutefois, si le texte peut appartenir à plusieurs catégories et certains documents du corpus d'apprentissage appartenant à une catégorie Ci initialement ne seraient pas classé dans Ci par le classificateur, le classificateur d'origine s'est classé. La méthode Rocchio est basée sur la génération de profils de catégorie. Le poids du terme est calculé lors de l'apprentissage de l'apparence avec des documents appartenant au document de catégorisation.

$$w_{ki} = \alpha . \sum_{t_{j \epsilon pos_i}} \frac{w_{kj}}{|pos_i|} - \beta . \sum_{t_{j \epsilon NEG_i}} \frac{w_{kj}}{|NEG_i|} \quad \textbf{(1. 1)}$$

Pour POSi tous les documents Tr sont dans la catégorie Ci et POSi tous les documents Tr ne sont pas dans la catégorie Ci. Les valeurs réelles α et sont définies arbitrairement. Généralement α > β.

1.4.2 Naïve Bayes :

Cette méthode calcule probabilité conditionnelle basée sur le nettoyage de BAIZ. Pour CT, la méthode naïf Bayes est utilisée comme suit : Trouvez une classification qui optimise la probabilité d'observer le mot du document. Lors de la phase de formation, le classificateur calcule probabilités qu'un nouveau document appartient à telle catégorie à partir de la proportion des documents d'entraînement appartenant à cette catégorie. . En outre, il est également connu que ce texte appartient à une telle catégorie, et le mot donné est également calculé comme le mot donné présent dans le texte. Si vous devez catégoriser un nouveau document, la probabilité de chaque catégorie de règles de baie est calculée.

$$P(A/B) = \frac{P(A \cap B)}{p(A)} \quad \textbf{(1. 2)}$$

1.4.3 Les Machines à Vecteur Support (SVM)

Les séparateurs à vaste marge, connus par nom : les Machines à Vecteur Support (SVMs), ou en anglais Support Vector Machines, ont été développés par Cortes et Vapnik en 1995 (Cortes &Vapnik, 1995). SVMs sont conçues au début pour la classification binaire, les problèmes de discrimination. Contrairement aux autres stratégies de certains algorithmes qui visent à trouver juste une séparation, peu importe la qualité, entre les deux classes lors

de la génération de leurs modèles, SVMs non seulement tendent vers trouver une séparation mais aussi elles essayent de générer un séparateur optimal dans un sens de tracer un hyperplan, qui maximise la distance (la marge) entre les bornes ou les frontières des nuages de points positifs et négatifs à la fois.

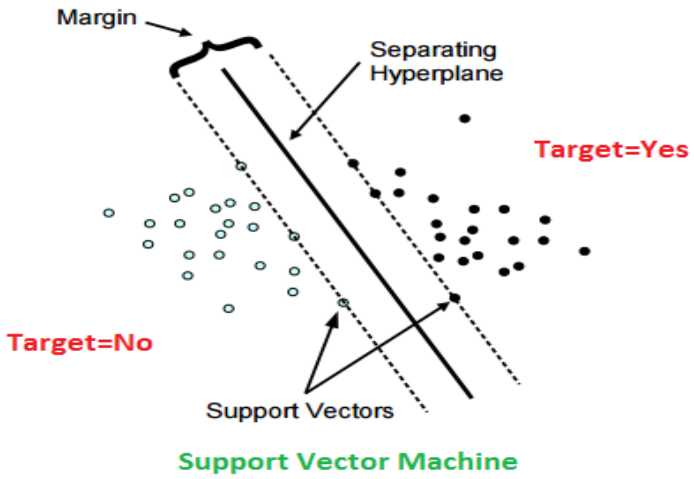

Figure1. 4 classification linière par SVM

Dans la figure ci-dessus on montre un cas de classification binaire. Les points d'extrémités des deux classes s'appellent Vecteurs Supports sur lesquelles on construit l'hyperplan qui occupera la position du milieu entre eux. Plusieurs extensions ont été développées par la suite tel que la régression, et la classification multi-classe dans laquelle les SVMs utilisent la technique un-contre-un en ajustant toutes les sous-classes binaires. La décision de classification d'une instance donnée est obtenue par un mécanisme de vote.

Pour les cas non-linéairement séparables les SVMs utilisent la technique des noyaux (Kernels) afin de transformer le problème initial de son espace d'origine vers un autre de plus grande dimensions.

1.4.4 Les arbres de décision

Les arbres de décision jouent un rôle dans la classification des observations futures de l'observation des observations d'observations. C'est le cas de nos exemples de plantes qui ont 100 commentaires qui ont déjà été classés en espèces. L'arbre commence par la racine (où nous avons toutes nos observations) (où nous avons notre observation), puis l'intersection suivante appelle le nœud, chacun de l'ensemble de divers types de classes diverses. Nous parlons au nœud maximal la profondeur de l'arbre avant d'atteindre la feuille. Chaque nœud d'arbre représente la règle (par exemple, la longueur des pétales de plus de 2,5 cm). Accédez à l'arbre Figure 1.3 pour voir une série de règles.

L'arborescence est créée que chaque nœud est la règle (type de mesure et type de seuil) à laquelle chaque nœud est le meilleur partitionnement (type de mesure et type de seuil) à laquelle toutes les observations de démarrage sont les meilleures.

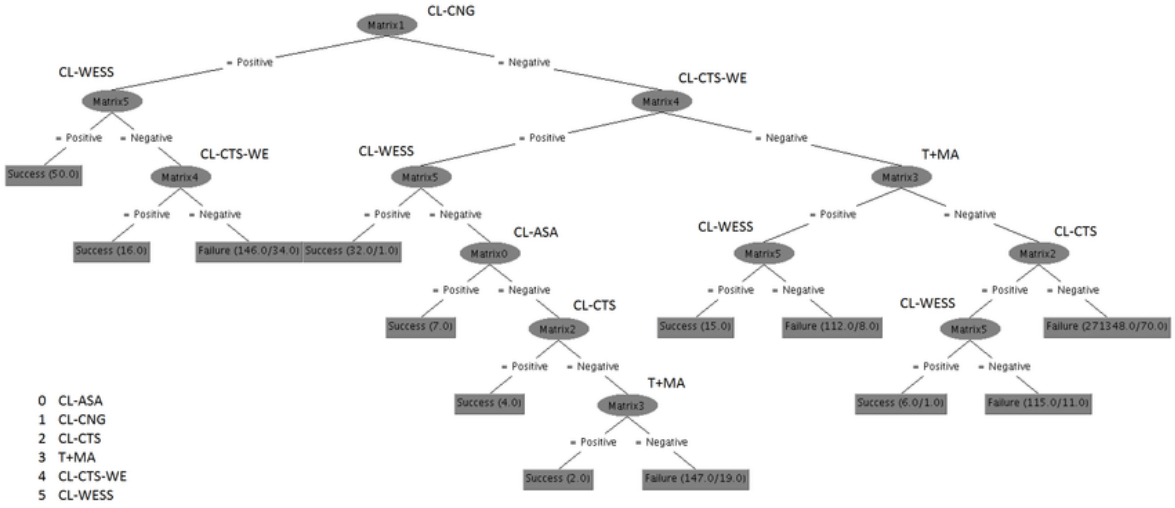

Figure1. 5 Arbre de décision appliquée sur la Base Reuters avec l'outil Weka.

1.4.5 L'algorithme K-plus proches voisins

Le k le plus proche voisin est une approche très connu dans la classification du texte. KPLUS Fermer les résultats du quartier représentent chaque texte de l'espace vectoriel, dont chacun des axes représente un élément textuel (peut être un mot sous sa forme brute ou sous une forme lemmatisée).

Les algorithmes de classification K plus proche voisins sont affichés comme suit :

Algorithme KNN (DATA, k, distance)

---

1. Charger les données
2. Initialiser la valeur de k
3. Parcourir et Calculer la distance entre le point teste et l'ensemble des points d'apprentissage
4. Trier dans une liste les documents (points) d'apprentissage en ordre croissant en fonction des valeurs de distance.
5. Obtenir le top k documents à partir de notre liste triée
6. déterminer la classe la plus fréquente de ces documents
7. Retourner la classe trouvée Fin d'algorithme

Le choix du paramètre K est primordial pour le bon fonctionnement de cette méthode

$$\boldsymbol{Eucleidien}\ (\boldsymbol{X},\boldsymbol{Y})\ =\ \sqrt{\boldsymbol{\Sigma}}\ (\boldsymbol{Xi}\ -\ \boldsymbol{Yi})^2\boldsymbol{ki}=\boldsymbol{1} \qquad \textbf{(1. 3)}$$

$$\boldsymbol{Manhattan}\ (\boldsymbol{X},\boldsymbol{Y})\ =\ \boldsymbol{\Sigma}\ /\boldsymbol{Xi}\ -\ \boldsymbol{Yi}/\ \boldsymbol{ki}=\boldsymbol{1} \qquad \textbf{(1. 4)}$$

$$\boldsymbol{Minkowski}\ (\boldsymbol{X},\boldsymbol{Y})=\sqrt{\boldsymbol{\Sigma}}/\boldsymbol{Xi}-\boldsymbol{Yi}/^{\boldsymbol{p}}\ \boldsymbol{ki}=\boldsymbol{1}\boldsymbol{p} \qquad \textbf{(1. 5)}$$

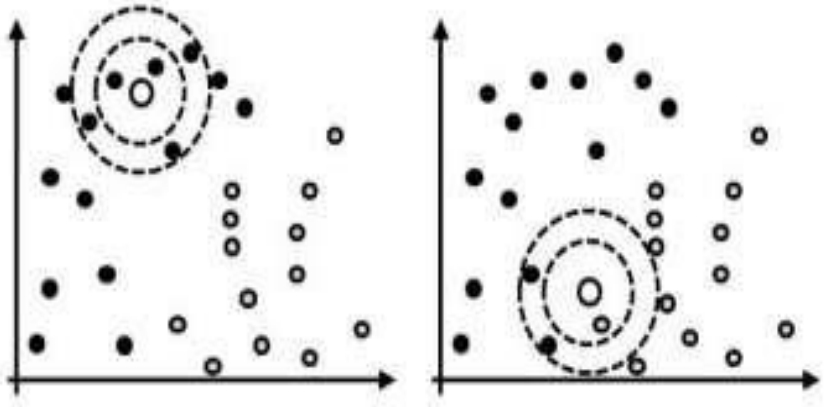

Figure1. 6 Exemple de classification avec les K-nn cas de deux classe A et B.

Toutes les mesures de similarité ou distance peuvent être utiles Pour déterminer les plus proches documents. En particulier, la distance Eucleidien, Manhattan et la similarité Cosinus parmi d'autres. Les expériences ont montré que l'augmentation de k n'augmente pas forcément la performance du classificateur K-nn, (Uğuz, 2011).

1.4.6 Bagging

Bagging (Breiman, 1996) est généralement conçu pour réduire l'erreur due à l'augmentation du processus d'apprentissage. Ces méthodes sont basées sur petits modèles de sous-minez, techniques d'échantillonnage (sélectionnant à l'aide de réductions) pour créer classificateurs indépendants (sélectionnant à l'aide de réductions). Par conséquent, chaque classificateur est donc différent de l'échantillon et est différent des autres classifications. Ensuite, combiner et / ou fusionner les résultats de la classification de l'échantillon pour obtenir le résultat final. Figure Ci-dessous décrit le concept de méthode de sac. Celles-ci peuvent généralement être, généralement utilisées avec tous les types de classifications et en particulier des cristaux. Le principal inconvénient de cette

méthode peut entraîner une faible précision en raison de la faible taille d'un échantillonnage d'apprentissage. La méthode de l'ensachage est remarquablement apprise par l'apprentissage, mais elle n'est utile que si le classificateur est instable aux petits détails de l'étape d'apprentissage. Un petit changement de données génère un changement important dans le modèle d'apprentissage généré. Un exemple de l'un de ces algorithmes est un arbre de décision très sensible qui est très sensible à la manière dont les nœuds supérieurs de l'arbre sont composés d'un espace de propriété très élevé (Grandvalet, 2004).

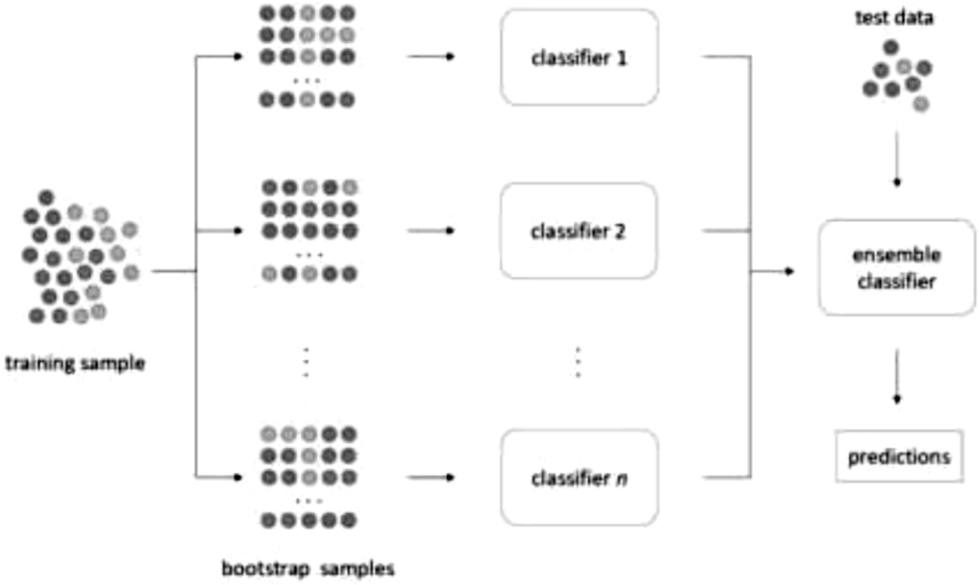

Figure1. 7 le schéma général de Bagging

1.5 Conclusion

Nous allons parler dans ce chapitre sur la fouille de données ou bien l'exploration de données et ces modèles et après nous avons vu que les algorithmes d'apprentissage des machines servaient deux choses: classer et prédire et ont été divisés en algorithmes supervisés et non supervisés. Il y a de nombreux algorithmes possibles, nous avons voyagé d'eux dont la régression logistique et les forêts aléatoires pour classer une observation et un regroupement pour apporter des groupes homogènes des données. Nous avons également constaté que la valeur d'un algorithme dépendait de la fonction de coût ou de perte associée, mais que son pouvoir prédictif dépendait de divers facteurs liés à la qualité et au volume des données.

Chapitre II

Prétraitement et représentation de texte

2.1 Introduction

Durant ces dernières décennies, nous sommes entrés en contact avec une croissance importante d'un grand nombre de documents électroniques textuels non structurés. Donc il est très difficile d'obtenir des informations. De plus, parcourir ces documents à partir du Web est devenu une tâche de plus en plus fastidieuse d'une part, et coûteuse en termes de temps d'autre part. De ce fait, le développement automatique de ces bases de données documentaires est devenu une nécessité majeure. De ce point de vue, la structuration de ces fichiers est nécessaire, pour que la technologie informatique puisse être appliquée, notamment la classification automatique. En général, la structure de ces fichiers est une préparation informatisée. On peut distinguer deux grandes étapes dans leur structuration ou indexation, à savoir : le nettoyage et la présentation. De plus, il existe plusieurs méthodes de représentation et différentes techniques de nettoyage, dont il existe plusieurs étapes. De plus, chaque type de données, à savoir : vidéo, image, la parole et/ou texte spécifique, possède son propre processus de prétraitement et un mode de représentation approprié. Dans ce chapitre, nous montrons les dernières technologies dans différentes méthodes de représentation de texte, de terminologie pondérée et de méthodes de codage. Ensuite, nous avons présenté les étapes de nettoyage et décrit les méthodes et méthodes de réduction de la dimensionnalité.

2.2 Le prétraitement :

Le prétraitement du texte est une étape clé, et il joue un rôle déterminant dans tout processus de classification, qu'il soit automatique ou textuel. Lorsque les performances du classificateur sont directement et significativement affectées. Quelle que soit la capacité du classificateur, un bon nettoyage améliorera la classification et vice versa. Ce dernier élimine tous les mots et bruits inutiles, ainsi que les mots qui ont un impact négatif sur le contexte de classification. Après la première opération d'un système de classification, nous avons besoin d'autant de mots redondants et inutiles que possible. Les mots non liés à la classe sont également omis. Utiliser des mesures statiques et précises pour déterminer la validité du terme et la gravité de son association avec une classe donnée. Les mots très peu fréquents et rares avec des taux d'utilisation très faibles dans l'ensemble du corpus sont considérés comme des mots inutiles.
Selon la loi de ZipF (Zhu, Zhang, Wang, Li, & Cai, 2018), malheureusement les termes importants qui ne sont ni trop fréquents ni trop rares ne resteront pas dans les critères de classement, même si les mots vides et les outils de mots sont importants et utiles pour toutes les langues. Dans l'étape de prétraitement, chaque langue a un vocabulaire pré-construit, et ses mots sont supprimés pour les distinguer des mots outils et des mots vides.Bien qu'ils aient une

importance sémantique et un contexte, ils ne sont pas inclus dans la classification dans le processus. Par exemple, nous avons trouvé dans la liste des mots vides français : "le, la, les, ce, ceux, donc, mais, ou... .Etc.". Il est automatiquement évalué à autant de mots parasites. Il convient de noter que ces mots subissent un traitement spécial, appelé suppression des mots vides. Le prétraitement est généralement effectué en cinq étapes séquentielles :

2.2.1 Suppression des caractères inutiles :

La reconnaissance des termes utilisés cera la première opération que doit l'effectuer un système de classification. La segmentation (tokenization) permet de représenter un document par un ensemble de mots, termes à partir de l'ensemble d'entités linguistiques qui sont apparus dans ce dernier. Pour le faire, avant tout il faut déterminer les délimiteurs des mots (reconnaissance des débuts et fins des mots et paragraphes). Habituellement, cette étape permet de découper les séquences de caractères en fonction la présence ou l'absence de caractère de séparation (délimiteurs) suivants : espace, tabulation, virgules, retour à la ligne… etc. Généralement, dans cette phase en utilise une liste contenant les signes de ponctuation même que les chiffres pour former des nombres (reconnaissance éventuelle des dattes), par exemple, toutes les caractères figurants dans cette suivante : {.,;:'''()?!&-#0123456789+/<>$^%[]≠'} seront supprimés, cette dernière aide à ressortir et de représenter les mots (Token). Aux principes, c'est un traitement assez simple, mais pour les documents ayant des représentations variées et beaucoup de bruit, la réalisation exacte de ces derniers doit être notamment difficile.

In all the times and societies, it was very beneficial for playing sports., Sports and games give needed competitive nature and a strong desire to win. Moreover, when competing with opponents, it is easier to gain proper organizational ; decision-making and strategy building skills. Thus, participation in sports was always aimed at bringing numerous benefits for participants.	In all the times and societies it was very beneficial for playing sports Sports and games give needed competitive nature and a strong desire to win. Moreover when competing with opponents it is easier to gain proper organizational decision making and strategy building skills Thus participation in sports was always aimed at bringing numerous benefits for participants

Figure2. 1 Exemple de suppression des caractères inutiles

2.2.2 L'élimination des mots vides (stop words) :

Une fois les documents textes découpé en token, dans cette étape on élimine les mots inutiles. Dans un corpus, on trouve les mots fréquents sont les mots vides (stop words) : article ; déterminant, adjectif, préposition, conjonction…etc. Par exemple on peut citer quelques mots outilles pour l'Anglais : 'the', 'for', 'and', 'a' et 'it', la présence de ces mots n'apporte aucune différence tant sur le plan sémantique ou lexicale, donc ces mots-là sont non discriminants dans la base textuelle vu leurs fréquences, et du coup leur utilisation s'avère inutile pour une tache de classification. En contrepartie, la suppression de ces mots va assurer une amélioration des performances des classificateurs, ainsi qu'un avantage en termes de taille de vocabulaire et sa réduction. Par conséquence l'élimination de ces mots réduit considérablement le temps du traitement, mais il n'affecte pas les performances de la classification textuelle.

In all the times and societies, it was very beneficial for playing sports., Sports and games give needed competitive nature and a strong desire to win. Moreover, when competing with opponents, it is easier to gain proper organizational ; decision-making and strategy building skills. Thus, participation in sports was always aimed at bringing numerous benefits for participants.	times societies, beneficial playing sports. Sports games needed competitive nature strong desire win. competing opponents, easier gain proper organizational, decision-making strategy building skills. participation sports aimed bringing numerous benefits participants

Figure2. 2 L'élimination des mots vides

State-level case, laboratory, and hospital data updated daily by 5:00 p.m. MT, excluding holidays. Data are current as of 9/17/2021. Case data are based on surveillance system records provided by the public health districts.	State-level case, laboratory, and hospital data updated daily by : p.m. MT, excluding holidays. Data are current as of / / . Case data are based on surveillance system records provided by the public health districts.

Figure2. 3 La suppression des numéros

2.2.3 Traitement des lettres majuscules :

Les lettres majuscules sont utilisées aux débuts des phrases, dans les noms propres et même dans les abréviations. On l'est trouve souvent utilisé dans paragraphe (corpus textuels).
Dans cette phase, on élimine les majuscules. La méthode la plus évidente de les traiter est de les convertir en minuscules. Cette technique possède des inconvénients, notamment le risque d'ambigüité ou de perte de sens.

Sports helps us to live a Fit and healthy life. Playing sports and games make our body and mind fresh. Most important thing, our heart is strengthen after playing sports and games. ... As sports have our body's physical involvement, sports make our blood vessels clean and it reduces the fat from our body.	sports helps us to live a fit and healthy life. playing sports and games make our body and mind fresh. most important thing, our heart is strengthen after playing sports and games. ... as sports have our body's physical involvement, sports make our blood vessels clean and it reduces the fat from our body.

Figure2. 4 Le traitement des majuscules

2.2.4 La désuffixation (stemming) :

La désuffixation ou le (steming en anglais) est un traitement lexical appliqué sur les mots selon leurs variations morphologiques, à savoir les flexions, dérivations et compositions. Donc elle remplace, ou, regroupe un ensemble de mots de différentes morphologies qui expriment le même sens, par un seul terme qui est leur racine (stem). Le principe de cette opération s'articule sur des règles de troncatures qui servent à éliminer les suffixes des mots, à savoir : 'ED', 'ING, 'ION 'IONS'. A ce titre, en langue anglaise, les termes : ''consult '', ''consultant '', ''consulting '', ''consultantative '' et''consultants '' seront regroupés par le stem ''consult''. Par ailleurs, et pour d'autres termes, l'application de la désuffixation nécessite un traitement exceptionnel. A titre d'exemple, les mots 'business' et 'busy' se réduisent, en supprimant leurs suffixes, au même mot 'busi' tandis qu'ils ont des sens différents. En plus, un autre inconvénient de cette technique est qu'elle peut fournir des stems qui ne sont pas des morphèmes, c'est-à-dire n'existent pas dans le dictionnaire, comme c'est le cas du mot 'computer' qui devient 'comput'. Par ailleurs, chaque langue a ses propres algorithmes de désuffixation, c'est-à-dire que les règles appliquées en Français ne sont pas les même pour une autre langue. En revanche, les avantages de cette phase sont comme suit : elle ne nécessite pas trop de connaissances linguistiques comparée à la lemmatisation. Aussi, elle ne demande pas un dictionnaire. Les algorithmes de désuffixation sont basés sur des règles de

troncature de mots afin de produire des racines, et ils sont, en conséquence, trop rapides. L'algorithme de PORTER (Porter, 1980) est l'un des algorithmes de racinisation les plus reconnus. Il est conçu pour l'anglais et il contient plus de cinquante règles. [1]

"\n\ncomputer terminal systems cpml completes sale\n\n commack ny feb computer terminal systems inc said\n completed sale shares common\nstock warrants acquire additional one mln shares \nsedio nv lugano switzerland dlrs\n company said warrants exercisable five\nyears purchase price dlrs per share\n co-mputer terminal said sedio also right buy\n	"comput termin system cpml complet sale commack ny feb comput termin system inc said complet sale share common stock warrant acquir addit one mln share sedio nv lugano switzerland dlrs compani said warrant exercis five year purchas price dlrs per share comput termin said sedio also right buy

Figure2. 5 Exemple d'application de la racinisation

2.2.5 La lemmatisation :

La lemmatisation est une phase trop compliqué par rapport a la désuffixation, elle consiste a utiliser l'analyse grammatical afin de convertir les verbes par leur formes infinitives et les nom par leur singuliers. En plus, dans un texte un mot donné peut avoir différentes formes, mais leur sens reste le même. Par exemple, l'ensemble des mots suivants : {players, walked, moved, workers} seront remplacé par leur lemmes : {player, walk, move, worker}.Malgré la simplicité de cette représentation, mais on utilise les règles de dérivation ainsi que des dictionnaires afin d'extraire des lemmes ou des racines.

TreeTagger (Schmid, 1994) est un algorithme développé dans ce contexte pour les langues anglaise, française, allemande et italienne. A trouver que la lemmatisation est très concurrentielle en termes de performances de catégorisation, mais plus coûteuse en termes de temps et ressources par rapport à la désuffixation.

2.3 Représentation des textes :

L'application des algorithmes d'apprentissage sur des données de type images, vidéos et notamment sur des données textuelles exige que ces données brutes soient transformées en une forme bien particulière c'est la forme numérique**.** [1]

Dans le processus de la catégorisation du texte, la représentation des textes est une étape très essentielqu'elle nécessite d'utilisé la représentation la plus courante et la plus éfficase est celles du modèle vectorielle. Le codage binaire : Si le terme est apparait dans le document on

met 1, si non 0. Le codage le plus appliqué dans la représentation est le codae TF IDF qui considéré comme une combinaison de deux importantes critères :

- la pondération TF (termfréquency) qui désigne l'importance locale du terme.
- La pondération IDF (inverse document fréquency) qui désigne l'importance globale du terme.

2.3.1 Représentation en (sac de mots) (Bag of words) :

La méthode de représentation la plus connue et la plus évidente pour la classification textuelle, est la représentation en sac de mot (bag of words). Ce dernier dépend à transformer chaque document en vecteur de mots, ou termes. Dont chaque composante, (Salton& McGill, 1986) et bien d'autres ont préférée l'utilisation du nombre d'occurrences des mots pour le codage des documents, comme ils ont ignoré l'ordre des mots, ainsi que les analyses grammaticaux dans cette représentation. Cette représentation montre ses avantages dans plusieurs applications, particulièrement la classification, malgré la détérioration de la structure des textes (perte de l'ordre des mots), comme inconvénient on trouve la difficulté de délimiter du mot qu'il soit traité d'une façon automatique. Pour clarifier la notion de mot, Y. Gilly dans son ouvrage « Texte et fréquence » (Gilly, 1988) l'a considéré comme étant une séquence de caractères appartenant à un dictionnaire, ou formellement, comme étant une suite de caractères séparés par des espaces ou des caractères de ponctuations (Cette définition n'est pas valable pour toutes les langues) [3].En tant qu'il faut prendre en compte la gestion des mots composés, il faut aussi étudier le cas des mots de la même famille (qui ont des différentes formes morphologiques et comportent le même sens), et les assembler en un seul terme à savoir : étude ; étudiant ; étudiants ; étudier… les regrouper en un seul mot c'est (étude), et cela pour amplifier la fréquence du terme et réduire la taille du vocabulaire.

D1= 'time is money' **D2= 'life is your time'** **D3= 'life is money'**
vocabulary={time, is, money, your, life}
D1=[1 1 1 0 0] *D2=[1 1 0 1 1]* *D3=[0 1 1 0 1]*

Figure2. 6 Exemple de trois documents représentés par sac de mots.

2.3.2 La représentation par phrase :

La représentation par phrase est une unité de représentation recommandée par de nombreux chercheurs afin d'éviter la déstructuration syntaxique causée par la représentation en "sacs de mots".Cela est dû au fait que les phrases sont plus informatives que les mots seuls et cette méthode de représentation permet aussi de conserver la sémantique, Ce qui serait difficile à faire avec un type de représentation qui repose sur des mots épars composant des textes très éloignés de ceux qu'ils sont censés représenter.. Logiquement, la représentation par phrase obtiendrait de meilleurs résultats d'usa la richesse sémantique. Cependant, les résultats empiriques obtenus en pratique montrent que les performances de l'algorithme de classification sont nettement inférieures à celles de la méthode du sac de mots. En effet, leurs propriétés statistiques ne permettent pas de formuler des hypothèses statistiques fiables, car le grand nombre de combinaisons de mots possibles génèrent des fréquences basses et trop aléatoires, ce qui empêche l'approximation correcte du risque réel en raison du risque empirique. Pour remédier à cette situation, on n'accorde pas d'importance à toutes les séquences possibles et on effectue une sélection des phrases en privilégient celles qui sont sémantiquement riches. Une autre méthode de (Caropreso, Fernandacnandad, Matwin, &Sebastiani, 2001) est d'utiliser une représentation basée sur le remplacement des phrases grammaticales par des phrases statistiques en tant que descripteur. Une phrase statistique est une collection de mots adjacents (qui n'est pas forcément ordonnés), qui apparaissent ensemble, mais qui ne respectent pas forcément les règles grammaticales. Selon une étude expérimentale menée par (Scott &Matwin, 1999), la représentation par phrases permet une amélioration des résultats comparés aux méthodes de type« sac de mots » lorsque les documents étudiés sont en nombre et en taille limités.

2.3.3 Représentation par les N-gramme :

La méthode de représentation par les N-gramme consiste à représenter un mot par une suite de caractères qui le composent. Cette technique présente de multiples avantages. L'un de ses plus grands atouts est qu'elle ne dépend pas de la langue utilisée contrairement au modèle sac de mots. De plus, elle n'a pas besoin de prétraitement ou d'opération de nettoyage comme : l'élimination des mots vides, la racinisation, la lemmatisation, etc, contrairement au modèle sac de mots. Puisque la notion de segmentation n'est plus, on n'a plus besoin de déterminer les débuts et les fins de mots. Par la suite, la représentation par les N-gramme a une tolérance aux déformations et supporte les erreurs d'orthographes : si on prend un corpus qui contient des erreurs d'horographes, des mots mal écrits ou scannés avec erreurs, donnant ainsi toujours

de bonnes performances ce qui est un autre grand avantage. À l'inverse, la méthode de représentation par sac de mots est très sensible aux déformations.

2.3.4 Etiquetage syntaxique (Part of speech tagger)

Les étiqueteurs syntaxiques ou POS taggers sont des outils d'analyse morphosyntaxique qui peuvent être employés pour aider un processus de lemmatisation. Ces étiqueteurs consistent à donner une étiquette grammaticale (catégorie grammaticale) à un mot dans un texte brut, telles que : nom, verbe, adjectif, etc, Certaines applications utilisent des étiquettes plus fine comme «nom-pluriel».

2.4 La pondération :

La façon la plus évidente pour coder les termes d'un vecteur pour chaque représentation textuelle, notamment la représentation en sac de mots est la pondération qui permet de mesuré l'importance des termes dans un document représenté.

Evidemment, on choisit la formule la plus simple (la pondération binaire), qui s'intéresse que sur la présence et l'absence des termes dans un corpus : 1 si le mot est présent, 0 si non.

Une autres approche simple consiste à le représenté à l'aide de leur fréquence d'un terme par rapport à leur nombre d'occurrence pour un tel document.

En général, un vecteur dans le modèle de sac de mots, comporte généralement les poids $w_{i,k}$de chaque mot i pour le document k. Soit la notation suivante :

- $\boldsymbol{f_{i,k}}$: la fréquence du mot i dans le document K.
- $\boldsymbol{N}$ **:** le nombre de documents dans la collection.
- $\boldsymbol{n_i}$**:** le nombre total de fois que le mot i se produit dans la collection.

2.4.1 Pondération booléenne :

Si le mot apparaît dans le document le poids est égal à 1, sinon égal à 0.

$$w_{i,k} = \begin{cases} 1 & si : f_{i,k} > 0 \\ 0 & sinon \end{cases} \quad \textbf{(2. 1)}$$

2.4.2 Pondération par fréquence de mot :

La formule est ci-dessous montre la pondération par fréquence qui désigne le nombre d'occurrences de terme pour un document. Elle est définie comme suit :

$$w_{i,k} = f_{i,k} \quad \textbf{(2. 2)}$$

2.4.3 Pondération TF-IDF :

- **TF*IDF** acronyme de **(Term Frequency Inverse Document Frequency):**est la formule la plus courante dans le monde du codage textuelle. Il prend en compte deux importants critères pour un terme :
- **TF (Term Frequency) :** La fréquence d'un terme est simplement le nombre d'occurrences de ce terme dans le document considéré**.**
- **IDF (Inverse Document Frequency) :** La fréquence inverse de document est une mesure de l'importance du terme dans l'ensemble du corpus.

Le poids $\boldsymbol{W_{i,k}}$ d'un terme i dans un document k est calculé comme suit :

$$\boldsymbol{w_{i,k} = TF * IDF = f_{i,k} * log(\frac{N}{n_i})} \qquad \textbf{(2. 3)}$$

(Salton& Buckley, 1988) et (Joachims, 1999) confirment que la représentation la plus utilisée dans la recherche d'information documentaire, qu'en classification est la représentation TF-IDF avec toutes ses variantes.

2.5 Réduction de la dimensionnalité :

L'inconvénient majeur de la représentation par sac de mots se présente dans la longueur de son vecteur de caractéristiques. De plus, la plupart de ses fonctionnalités sont redondantes ou non pertinentes. La suppression des mots vides, en particulier, a quelque peu atténué la gravité du problème, mais reste encore insuffisante. Par ailleurs, l'effet de la suppression des mots rare du corpus reste faible. Pour y remédier, une supplémentaire phase de réduction de dimensionnalité est nécessaire. Elle permet de réduire la taille du vocabulaire au-delà de l'étape de prétraitement, en éliminant certains attributs avant d'appliquer les techniques de catégorisation du texte. Cela permet, à la fois, d'accélérer le temps de calcul de la fonction de classification, réduire l'espace mémoire et d'éviter les problèmes de sur-apprentissage résultant de l'abondance du vocabulaire, ce qui améliore l'efficacité de la phase de classification. Il existe deux grandes familles qui utilisent des techniques issues de la théorie de l'information ou de l'algèbre linéaire (Sebastiani, 2002) : i) selon qu'elles agissent localement ou globalement. Dans le cas de la réduction de la dimension locale, pour une catégorie c donnée, un sous-ensemble de termes qui serviront à représenter les documents. Cependant, chaque document aura un sous-ensemble différent de termes selon la catégorie.

Quant à la réduction de la dimension globale, un seul sous-ensemble de termes apparaît. Ce dernier sera important et unique pour toutes les catégories afin que le document soit représenté par un seul sous-ensemble d'attributs dans toutes les catégories. ii) Selon la nature des résultats de la sélection. Il y a ici deux approches

- une approche de sélection de caractéristiques (featureselection) qui prend les attributs d'origine ne conserve que ceux jugés utiles pour la classification par critère et rejette les autres attributs. Dans ce contexte, il existe plusieurs mesures statistiques, à savoir : la fréquence, le gain informationnel, l'information mutuelle, la mesure du chi carré x^2 et la force du terme.
- La seconde approche, l'extraction d'attributs (feature extraction) génère de nouveaux attributs à partir des attributs initiaux par regroupement ou transformation. L'analyse en composantes principales (ACP) et l'analyse latente sémantique (LSA) font partie des méthodes couvertes par cette approche.

2.5.1 Extraction de caractéristiques

Afin d'isoler les caractéristiques (Gomez, Boiy& Moens, 2012), l'espace d'origine des caractéristiques est transformé en un nouvel espace plus compact. Tous les éléments d'origine sont transformés dans le nouvel espace réduit sans les supprimer, mais en les remplaçants par un ensemble de représentations plus petit. En utilisent deux méthodes d'analyse :

- Analyse des composants principaux (PCA)
- Analyse sémantique latente (LSA)

2.5.2 Sélection des caractéristiques :

La sélection des caractéristique, en anglais featureselection (FS), est généralement indiqué comme un processus de recherche qui nous permet de trouvé des sous-ensembles pertinents a partir de l'ensemble d'origine, c'est contrairement a l'extraction. Cette sélection a plusieurs applications dans des différentes domaines tels que le traitement d'image, la vision par ordinateur, la bio-informatique, les applications industrielles (Bogunović, 2015), le regroupement de documents (Kogan, 2003) et la catégorisation de textes (Lee & Lee, 2006).
(Dash et Liu [1997]) propose une procédure générale pour une méthode de la sélection de caractéristiques, représenté par la figure :

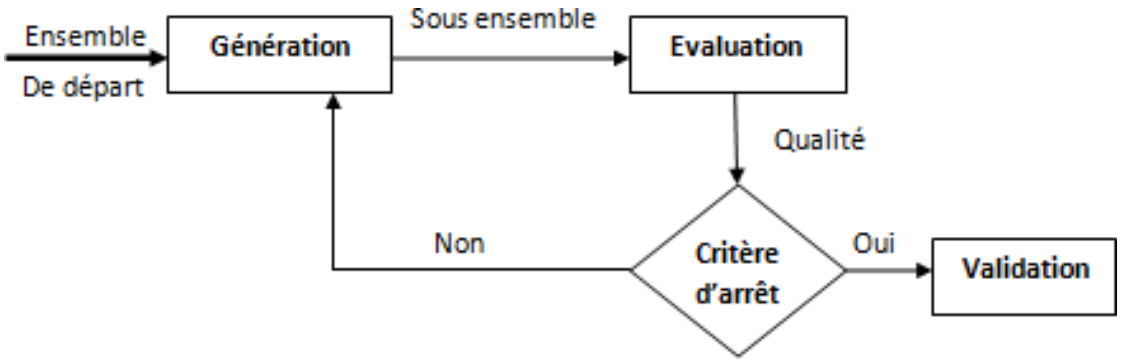

Figure2. 7 processus de sélection d'attributs

2.5.2.1 Le choix des caractéristiques :

Le processus de classification est effectué surtout selon les termes sélectionnés qui provient soit de l'étape de retraitement soit celle de sélection. Ce dernier, on peut l'amélioré et le rendre en bonne performance a partir de choisir les termes qui sont pertinent et d'éliminé tous ce qu'est intitule ou bien redondant. Par contre, si on choisir des termes non pertinent, la performance sera déradé.

Les éditeurs dans (John et al. [1994], John [1997]), déclare qu'il existe trois types de terme :

- **Très pertinent :** un terme qui on ne peut pas l'éliminer car son absence cause une dégradation significative de la performance de la classification utilisé.
- **Peu pertinent :** Une caractéristique fi est dite peu pertinente si elle n'est pas « très pertinente » et s'il existe un sous-ensemble V tel que la performance de V ∪ {fi} soit significativement meilleure que la performance de V. [2]
- **Non pertinent :** les caractéristique non pertinent sont les caractéristique ni peu pertinent ni très pertinent. En générale ces caractéristiques seront supprimées de l'ensemble des caractéristiques originales.

Pour l'évaluation d'un sous ensemble de caractéristique, les moyens utilisés dans les algorithmes de sélection peuvent être classés en trois catégories principales : « Filter », « Wrapper » et « Embedded ».

2.5.3 Filter :

La première approche appliquée pour la sélection des caractéristiques est l'approche « Filter ». Dans celui-ci le critère d'évaluation utilisé est la quantification de la pertinence d'un caractère a partir des mesure qui repose sur des propriétés d'apprentissage. Cette méthode est considérée davantage comme une partie réservée au processus de prétraitement (filtrage avant la phase d'apprentissage). En supplément, l'évaluation se fais généralement indépendamment de l'algorithme de catégorisation (Yiming Yang & Pedersen, 1997).pour

l'évaluation des caractéristique, les méthodes qui se basent sur se modèle utilise habituellement une approche heuristique comme stratégie de recherche. La procédure de modèle Filter est illustrée par la figure suivante :

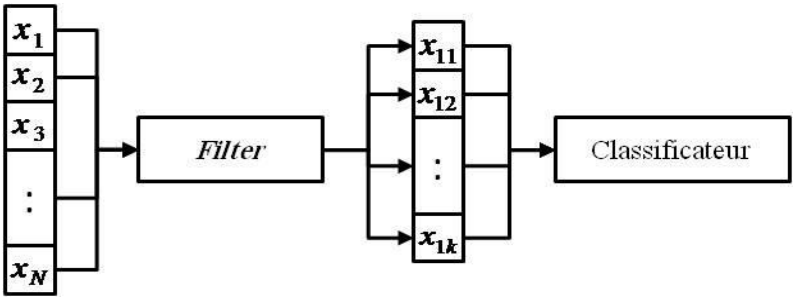

Figure2. 8 schéma de l'approche 'Filter'

Selon la procédure, il est nettement claire que le but de cette méthode d'évaluation est de sélectionné un sous-ensemble réduit de caractéristique et de calculer un score pour évaluer le degré de pertinence de chaque une de caractéristique, qui seront considéré comme pertinent et vont être utilisé dans l'étape d'apprentissage.

On citant quelques méthodes utilisé comme mesuré d'évaluation dans le filtrage :

- **La fréquence**
- **La force du terme**
- **Le coefficient de Gini**
- **Le gain d'information (information Gain)**
- **La statistique du $X2$**

2.5.3.1 Limite des méthodes 'Filter'

Malgré leur succès, leur efficacité, et même si elles se retrouvent dans plusieurs domaines d'application, les performances de ces dernières sont encore très limitées par rapport aux autres approches Wrapper et Embded. La principale raison de cette limitation est que les méthodes basées sur des filtres ne prennent pas en compte l'effet des caractéristiques sélectionnées sur les performances du classifieur utilisé par la suite. Ainsi, Il suffit d'évaluer les attributs individuels avec la classe vectorielle mais Il ne prend pas en compte la redondance entre les caractéristiques. Plus précisément, la progression des autres approches est due à la prise en compte de ces deux points.

2.5.4 Wrapper :

Malgré toutes les avantages de l'approche « filter », on trouve aussi des inconvénient, la principale de ces derniers est le fait qu'elles i l'effet des caractéristiques sélectionnées sur la performance du classificateur pour une utilisation ultérieur. C'est pour cela, (Kohavi et John [1997]) introduisent le concept « Wraper » pour contourner ce problème. « Wrapper » en anglais, ou bien Les méthodes enveloppantes, qui sont utilisées notamment pour la sélection des caractéristiques. Elles reposent généralement sur deux algorithmes d'apprentissage : l'un est de la recherche, pour explorer l'espace de solutions, et l'autre est l'algorithme de classification, pour évaluer des sous-ensembles de caractéristiques.

A la fin comme une solution finale, le meilleur sous ensemble sera approuvé. On voit que cette approche demande beaucoup de ressources et elle est aussi très couteuse en temps. Comme elle a un autre inconvénient est que l'algorithme de classification utilisé dans l'étape de classification finale est très dépendant de celui utilisé dans étape de classification. Cette approche « Wrapper » est illustré dans la figure suivante :

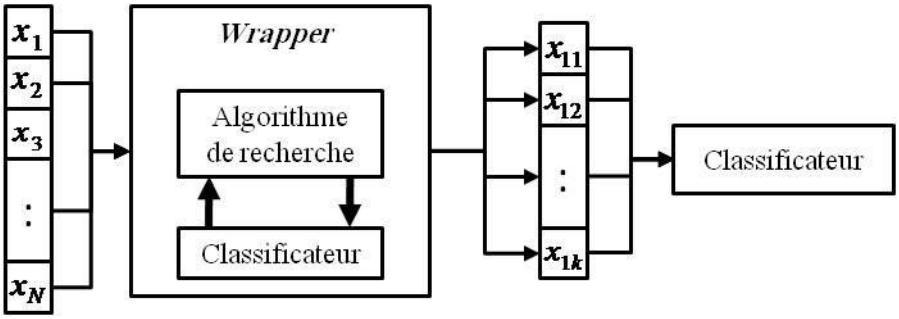

Figure2. 9 schéma général d'un « Wrapper »

2.5.5 Embedded :

Contrairement aux méthodes "Wrapper" et "Filter", la méthode "Embedded" (également appelée méthode intégrée) inclut la sélection de variables pendant le processus d'apprentissage. Par exemple, ce mécanisme d'intégration de sélection de fonctionnalités peut être trouvé dans des algorithmes de type SVM, AdaBoost ou bien l'arbre de décision. Dans la méthode de sélection de type « Wrapper », la bibliothèque d'apprentissage est divisée en deux parties : la bibliothèque d'apprentissage et la bibliothèque de vérification, qui sont utilisées pour vérifier le sous-ensemble de caractéristiques sélectionné. En revanche, l'approche intégrée peut utiliser tous les exemples d'apprentissage pour construire un système. C'est un avantage qui peut améliorer les résultats. Un autre avantage de ces méthodes est qu'elles sont

plus rapides que les méthodes "Wrapper" car elles empêchent le classificateur de redémarrer pour chaque sous-ensemble de fonctionnalités.

2.6 Conclusion

Dans ce chapitre, nous avons étudié la numérisation de documents Mot. Cette étape est non seulement essentielle, mais aussi décisive Informatisation des données. Tout d'abord, les étapes de nettoyage sont introduites. Prochain, Nous avons montré la dernière technologie de différentes manières de représenter le texte, Méthodes de pondération des termes et différents codages des termes. Cette La deuxième partie de ce chapitre est consacrée au problème des grandes dimensions. Exister Dans cette section, nous décrivons la réduction Dimension. Les méthodes existantes ont des limites : dans La complexité de la méthode "Wrapper" et de la méthode "Filter", L'inconvénient est qu'il est possible de choisir des attributs redondants et corrélés. Maintenant que nous avons décrit les méthodes de réduction de dimensionnalité, nous continuerons à introduire les principaux algorithmes de réduction de dimensionnalité dans le chapitre suivant La recherche est transférée dans ce domaine.

Chapitre III

Les algorithmes de recherche et sélection de caractéristique

3.1 Introduction :

Dans n'importe quel processus d'apprentissage, et particulièrement la classification textuelle Le choix des caractéristiques joue un rôle clé. Il permet d'extraire non seulement les meilleurs sous-ensembles mais aussi les sous-ensembles pertinents. Cela affecte positivement sur la qualité des algorithmes de classification de précision et de rappel. On a établi les concepts et les différentes méthodes utilisées pour la sélection des caractéristiques dans le chapitre précédent. Gardez cela à l'esprit, dans ce chapitre on va traiter principalement des algorithmes Recherche qui représente le cœur du processus FS. En outre, nous allons présenter les différentes stratégies de recherche. Comme nous allons introduire et décrire les algorithmes de sélection de caractéristique et notamment les algorithmes de recherche stochastiques. De plus, nous présenterons les concepts, les propriétés ainsi que les caractéristiques de chacun de ces algorithmes de recherches.

3.2 Les méthodes de sélection :

La méthode de sélection comprend généralement quatre étapes dont le but est de Extrayez le meilleur sous-ensemble de caractéristiques. Ces étapes sont exprimées en Comme suit : l'initialisation, le processus de recherche ou de génération d'une solution, une méthode d'évaluation et le dernier est le critère d'arrêt (Liu &Yu, 2005). Une méthode de sélection Commencez par la solution initiale afin que le processus de recherche puisse A partir de laquelle de nouvelles solutions sont générées, où la méthode d'évaluation est dans son aspect Sera utilisé pour attribuer un score à chaque solution trouvée. Ces deux étapes sont Répétez jusqu'à ce que la condition d'arrêt soit remplie.

Dans la section suivante, nous nous concentrons sur les procédures de recherche en décrivant leurs caractéristiques et ainsi que leurs caractéristiques Catégorie de recherche.

3.3 Les algorithmes de recherches :

Les trois méthodes de sélection principales sur la procédure d'évaluation sont : Wrapper, Filter et Embdded. Dans la première approche Wrapper l'algorithme d'apprentissage est utilisé pour apprécier un sous ensemble de variables. Alors que dans la deuxième méthode Filter, le critère d'évaluation n'a aucune relation avec l'algorithme d'apprentissage. Mais dans la troisième méthode Embdded on trouve que les approches Filter sont introduites dans l'approche Wrapper.

La qualité d'un sous ensemble sélectionnée dépend au choix de l'algorithme de recherche qui joue un rôle déterminant dans la méthode de sélection. Et delà on distingue trois types d'algorithmes de recherche : les algorithmes séquentiels, les algorithmes exponentiels et les algorithmes aléatoires. Qu'on les explique ci-dessous. Les algorithmes séquentiels sont heuristiques, ajoutant ou supprimant des variables séquentiellement, ce qui les confronte au problème de convergence prématurée vers l'optimum local. D'autre part, Les algorithmes exponentiels et déterministes évaluent plusieurs sous-ensembles, qui croissent de façon exponentielle par rapport à la taille de l'espace de recherche. Et delà nous obtenons un sous ensemble parfait. Enfin, l'algorithme aléatoire traite le problème FS comme une boîte noire.
Les derniers algorithmes atteignent itérativement la valeur de la variable de sortie en modifiant la valeur de la variable d'entrée lors de leur recherche. Cela leur permettra d'éviter les optima locaux et les rend également bien adaptés à tout type de problème d'optimisation.

3.3.1 Les stratégies de recherche :

Le processus de recherche met à jour les solutions trouvées pour améliorer les résultats de la fonction objectif, ou on doit prédéfini d'abord la recherche du meilleur sous en semble avant de démarrer le processus de sélection des caractéristiques. Comme on doit aussi prendre en compte la qualité du programme qui joue un rôle très essentiel dans cette recherche.
De manière générale, les stratégies de recherche sont devisées en trois importantes catégories : exhaustive, heuristique et stochastique.

- **Exhaustive**
- **Heuristique**
- **Stochastique**

Ci-dessous nous allons présenter quelques méthodes en détail pour une vue globale Algorithmes de recherche existants. Parmi elles, nous avons trouvé une stratégie de recherche, à savoir : stochastique, déterministe, heuristique. Et/ou ayant des comportements différents (évolutionnaire, essaim, local…etc,).

3.3.2 SFS et SBS

Pour la sélection de caractéristiques, on a proposé une heuristique de SFS (Sequential Forward Selection) ou (sélection séquentielle croissante). A chaque itération, la meilleure caractéristique parmi celles qui restent sera sélectionnée, supprimée de sous-ensemble de départ et ajoutée au sous-ensemble des caractéristiques sélectionnées. Le processus de sélection continue jusqu`à un critère d'arrêt. Répétez cette opération jusqu'à ce que le critère

d'arrêt soit atteint. Contrairement à l'algorithme SFS, la sélection séquentielle arrière SBS (Whitney [1971]) commence à partir de l'ensemble des fonctionnalités, et à chaque itération, la pire fonctionnalité sera supprimée. Ahan et Bankert ont comparé ces deux méthodes (Aha et Bankert [1995]). Les résultats de la recherche montrent les progrès de la méthode SBS, car elle calcule le score d'interaction entre une fonctionnalité et un ensemble de fonctionnalités plus large. Par rapport à SFS, SFS ne considère que l'interaction entre la fonctionnalité et le sous-ensemble actuel. De plus, la méthode SBS pose un problème au niveau de temps de calcul du fait de l'évaluation des sous-ensembles de grande taille.

3.3.3 Les algorithmes génétiques :

Les algorithmes génétiques sont un type d'algorithmes évolutifs basés sur l'inspiration naturelle. Ils imitent les Opérations de la mutation, traversée et sélection de gènes dans la nature. En faisons cette reproduction, les GAs tentent d'améliorer le score de la fonction objective. D'autre part, la mutation permet de désactiver ou de modifier une partie d'une solution existante pour éviter de tomber dans une solution optimale prématurée. Quant à la sélection, le troisième opérateur de GAs, la sélection permet de choisir les parents qui vont participer à la production des générations suivantes. GAs avaient plusieurs domaines d'applications et notamment dans celui de la sélection des caractéristiques. Par exemple, le GAs a été utilisé comme outil pour résoudre le problème de la sélection de la fonctionnalité dans le contexte de l'exploration de données.

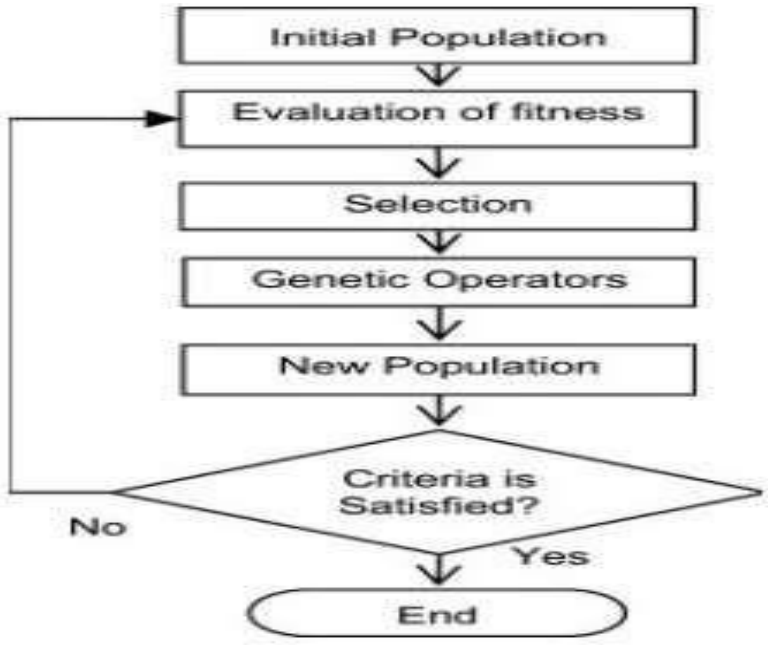

Figure 3. 1 Schéma général d'algorithmes génétiques.

3.3.4 Les colonies de fourmis (ANT) :

Les colonies de fourmis (en anglais, ant colony optimization, ou ACO), est une approche méta heuristique, qui appartient à la famille des méthodes intelligente en essaims. Elle est considérée comme une méthode de recherche aléatoire qui dépend du comportement de recherche de nourriture des essaims. Lorsque les fourmis suivre un certaine chemin, elles déposent chacune la même quantité des phéromones. Cette dernière attire les autres fourmis de la colonie à suivre le même chemin. La probabilité de choix du chemin optimale dépend de la quantité de phéromones, car ces deux derniers sont directement proportionnel : le chemin dont la quantité de phéromones est grande est le mieux. Par contre plus la quantité de phéromone n'est faible, la probabilité de choisir ce chemin est diminué. Au fil du temps, les courts trajets préservent davantage la phéromone. Contrairement, la quantité de ce dernier diminue dans les autres trajets selon le taux d'évaporation préréglé. Pour la classification du texte, les chercheurs ont appliqué une méthode de FS basé sur l'algorithme ACO. Comme ils ont proposé une autre méthode est FS avec PCA pour la catégorisation des paroles.

La figure montre le comportement des Fourmies sur une expérience durant une période de temps.

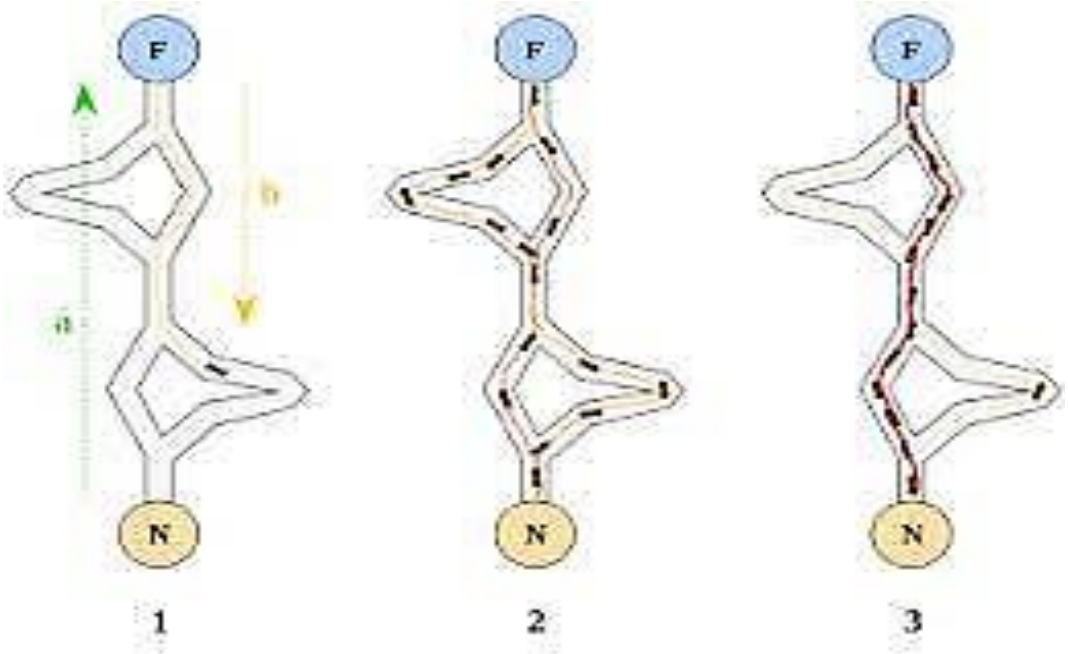

Figure 3. 2 optimisations par colonie des fourmis durant une période de temps.

3.3.5 L'optimisation par essaims particulaires (PSO) :

L'optimisation par essaims de particules, en anglais Particule swarm optimization (PSO) a été inspirée par le comportement animal et le comportement social humain. Initialement, cette méthode a été développée pour les problèmes persistants. PSO est une étude basée sur la

population qui vise à trouver des solutions sous-optimales dans l'espace de recherche. Chaque individu (particule) dans l'essaim et pendant le processus de recherche, change de manière itérative sa nouvelle position en fonction de sa vitesse, de la meilleure position locale trouvée jusqu'à présent et de la meilleure position globale. La PSO a ensuite été appliquée à l'optimisation discrète (Kennedy & Eberhart, 1997), qui prend en charge la sélection d'attributs et utilise des états binaires tels que 0 et 1.

Figure 3. 3 schémas de principe du déplacement d'une particule.

3.4 Conclusion :

Dans ce chapitre, nous avons traité les algorithmes de recherche qui sont la partie la plus importante dans le processus de classification de texte, particulièrement les méthodes de sélections. Où nous nous sommes notamment basés sur les algorithmes stochastiques qui sont utilisés pour la sélection des caractéristiques. Nous avons également étudié les algorithmes de recherche les plus connue comme les GAs, PSO et Aco. On a discuté aussi de certains des nouveaux algorithmes inspirés de l'algorithme SCA.

Chapitre IV

Implémentation et résultats

4.1 Introduction

Ce chapitre est essentiellement consacré à la partie implémentation et réalisation de notre projet afin d'atteindre l'objectif de notre thème d'étude. Dans cette optique, nous allons introduire : les outils exploités pour le développement du projet, tels que le choix du langage de programmation, l'environnement de programmation et le matériel utilisé. On va décrire l'approche utilisée dans les représentations de textes ainsi que le processus de classification appliqué. Les résultats des expérimentations sont communiqués et expliqués. Nous enchainons ensuite avec une discussion des résultats obtenus. A la fin de ce chapitre, nous terminons par une conclusion.

4.2 Outils Matériels et Logiciels

4.2.1 Configuration matérielle

Ce travail a été implémenté sur un PC, caractérisé comme suit :

- Un Processeur : Intel(R) Core(TM) I3-3217U CPU @ 1.80GHz
- Une RAM : 4 GO
- Sous un système d'exploitation 32 bits

4.2.2 Environnement logiciel

Pour implémenter notre application, on a choisi comme outils de développement, WEKA. Le tableau 4.1, donne une fiche technique de ce dernier. WEKA (acronyme pour *Waikato* environment for knowledge analysis, en français : « environnement Waikato pour l'analyse de connaissances »), est une suite de logiciels d'apprentissage automatique qui supporte plusieurs outils d'exploration de données standards, et en particulier, des préprocesseurs de données, des agrégateurs de données (data clustering), des classificateurs statistiques, des analyseurs de régression, des outils de visualisation, et des outils d'analyse discriminante.

Tableau4. 1 Caractéristique de l'outil utilisé, WEKA.

• Développeur	Université de WAIKATO (1992)
• Dernière version	3.8.1 (23/01/2017)
• Version avancée	3.9.1 (19/12/2016)
• Ecrite en	Java
• Environnement	plateforme Java
• Type	Structure logicielle d'apprentissage automatique
• Licence	libre disponible sous la licence publique générale (GPL)
• Site web	weka.wikispaces.com

4.3 La stratégie expérimentale suivie

Le processus de la catégorisation des textes passe généralement par cinq étapes : prétraitement, représentation, réduction du vocabulaire, puis la classification et après l'évaluation des résultats obtenus. Nous avons opté à appliquer la stratégie suivante, exprimée dans la Figure 4.2. :

1. télécharger
2. les données (Benchmark) .
3. Nettoyage de données
4. Appliquer une représentation sac de mots
5. Sélectionner le meilleur sous ensemble de termes en utilisant les algorithmes GA et PSO
6. Catégoriser les documents avec le classificateur NB
7. Evaluer les résultats obtenus.

Figure4. 1 la 1ere stratégie suivie dans cette expérimentation

8. télécharger les données (Benchmark) .
9. Nettoyage de données
10. Appliquer une analyse lexicale durant le processus de prétraitement
11. Sélectionner le meilleur sous ensemble de termes en utilisant les algorithmes GA et PSO
12. Catégoriser les documents avec le classificateur NB
13. Evaluer les résultats obtenus.

Figure4. 2 la deuxième stratégie appliquée dans cette expérimentation

```
14. télécharger les données (Benchmark ) .
15. Nettoyage de données
16. Appliquer une représentation hybride lexique et sac de mots
17. Sélectionner le meilleur sous ensemble de termes en utilisant
    les algorithmes GA et PSO
18. Catégoriser les documents avec le classificateur NB
19. Evaluer les résultats obtenus.
```

Figure4. 3 l'hybridation des stratégies suivie dans cette expérimentation.

4.4 Base de données utilisée

Le corpus sur lequel nous avons réalisé notre travail est composé de 1000 articles sportifs en anglais. Ces articles ont été rassemblé à partir de plus de 50 sites Web différents comme le site de : NBA.com, Fox Sports et Eurosport UK. La plupart de ces articles ont été rédigés par des professionnels journalistes, mais quelques-uns ont été écrits par des supporteurs sportifs sur des blogs. Les articles variaient en longueur de 47 à 4283 mots, avec une longueur moyenne de 697 mots. Ce corpus a été étiqueté par les auteurs dans en utilisant, Mechanical Turk d'Amazon, un outil de Crowdsourcing. Cet étiquetage a conduit à un résultat de : 658 articles ont été étiquetés objectifs et, 342 étiquetés subjectifs. Le jeu de données est téléchargeable à partir du site : https://archive.ics.uci.edu/ml/machine-learning-databases/00450/. Le tableau 4.2 suivant affiche les étiquètes utilisées dans l'étiquetage de ce corpus.

Nom	Modifié le	Type	Taille
Text0014.txt	12/03/2013 00:23	Fichier TXT	5 Ko
Text0015.txt	04/03/2013 09:57	Fichier TXT	6 Ko
Text0016.txt	12/03/2013 00:25	Fichier TXT	2 Ko
Text0017.txt	12/03/2013 00:26	Fichier TXT	4 Ko
Text0018.txt	12/03/2013 00:27	Fichier TXT	1 Ko
Text0019.txt	12/03/2013 00:27	Fichier TXT	2 Ko
Text0020.txt	04/03/2013 09:59	Fichier TXT	2 Ko
Text0021.txt	04/03/2013 09:59	Fichier TXT	7 Ko
Text0022.txt	04/03/2013 09:59	Fichier TXT	5 Ko
Text0023.txt	12/03/2013 00:30	Fichier TXT	1 Ko
Text0024.txt	12/03/2013 00:31	Fichier TXT	2 Ko
Text0025.txt	10/03/2013 16:30	Fichier TXT	1 Ko
Text0026.txt	12/03/2013 00:32	Fichier TXT	2 Ko
Text0027.txt	10/03/2013 11:12	Fichier TXT	6 Ko
Text0028.txt	12/03/2013 00:36	Fichier TXT	4 Ko
Text0029.txt	10/03/2013 16:32	Fichier TXT	6 Ko
Text0030.txt	04/03/2013 09:59	Fichier TXT	6 Ko
Text0031.txt	04/03/2013 09:59	Fichier TXT	3 Ko
Text0032.txt	04/03/2013 09:59	Fichier TXT	5 Ko
Text0033.txt	12/03/2013 00:39	Fichier TXT	4 Ko

Figure4. 4 Aperçue du dossier de la base de données.

Figure4. 5 Extrait du texte id 15 du DataSet.

4.5 Mesures utilisées dans l'évaluation des performances

Pour évaluer les algorithmes de classification, nous avons utilisé les mesures de *précision*, de *rappel*et de *F-mesure* qui sont communément utilisées. Pour définir ces mesures, nous avons besoin de définir les valeurs suivantes par rapport à une catégorie C:

- ***TP : le nombre de vrais positifs (True positives).*** C'est le nombre d'instances correctement classés dans la catégorie *C*.
- ***FP : le nombre de faux positifs (False positives).*** C'est le nombre d'instances incorrectement classés dans la catégorie *C*.

- ***FN : le nombre de faux négatifs (False negatives).*** C'est le nombre d'instances incorrectement classés en dehors de la catégorie *C*.

Ainsi les mesures de précision et de rappel sont définies comme suit :

Précision : la *précision(P)* est le rapport du nombre de documents correctement attribués à la catégorie *C* au nombre total de documents classés comme appartenant à la catégorie *C*.

$$\boldsymbol{P = \frac{TP}{TP+FP}} \quad \textbf{(4. 1)}$$

Rappel : le *Rappel (R)* présente le rapport du nombre de documents correctement attribués à la catégorie *C* au nombre total de documents appartenant réellement à la catégorie *C*.

$$\boldsymbol{R = \frac{TP}{TP + FN}} \quad \textbf{(4. 2)}$$

De plus, une troisième mesure commune est appelée *F-mesure*(FM) est définie comme suit :

F-mesure : la *F-mesure (F)* désigne la moyenne harmonique entre la précision et le rappel.

$$\boldsymbol{F = 2.\frac{P.R}{P + R}} \quad \textbf{(4. 3)}$$

4.6 Expérimentation

Nous avons réalisé une série d'expérimentations sur un jeu d'essai de données sportive ("Sports Articles For Objectivity Analysis Dataset") téléchargeable à partir la base UCI sur le lien :archives.ics.uci.edu/ml/datasets/Sports+articles+for+objectivity+analysis. En effet, Nous avons testé et comparé les résultats du classificateur Naïve bayes (NB) sur les trois différents jeux de données qui ont étés extrait à partir du corpus de 1000 articles sportif : bag-of-Words (BoW), Syntactique et Hybride. Les tableaux en-dessous montrent les résultats de performances de sur ces trois jeux de données et les métriques d'évaluation sont exprimées en termes de taux de TP (true positives), de FP (false positives), de Précision, de Rappel et de F-Measure. La longueur du vecteur des caractéristiques résultant de la phase de prétraitements est de : 200, 60 et 100 variables pour chacun des jeux de données mentionnés antérieurement dans leur ordre. On mentionne ici et dans la méthode de test, nous avons utilisé la validation croisée de 3-flods pour l'algorithme de classification NB.

4.6.1 Résultats

Dans cette partie on montre les résultats des classifications sur la base de données des articles sportifs à partir de laquelle on a extrait les trois jeux de données mentionnés auparavant. Dans ce sens, on va présenter chacune des expériences séquentiellement et séparément. A la fin nous terminerons cette partie par une comparaison et conclusion.

4.7 1ere expérimentation : la stratégie BoW

Dans ce cas, on a opté pour la représentation sac de mots (bag-of-words) BoW. Dans cette approche, il faut passer par une étape préliminaire qui est le nettoyage. Pour ce faire, nous avons réalisé et appliqué le pseudo code présenté en dessous dans la figure 4.6.

1. Input : Charger les données (1000 articles sportifs) % format texte
2. Supprimer les caractères de ponctuation
3. Supprimer les membres
4. Convertir les termes documents en minuscule
5. supprimer les mots vides (“dictionnaire anglais”)
6. racinisation des documents
7. codage en TF et/ou nominale
8. garder 97% des termes les plus fréquents
9. Output: matrices incidence des termes documents

Figure4. 6 processus de prétraitement dans l'approche BOW.

D’après les résultats exposés dans le tableau 4.2, on constate que dans le cas où aucune méthode de sélection d’attributs n’est utilisée, l’algorithme NB donne des résultats de performances médiques en termes de Taux de TP , Taux de FP, Précision , Rappel et F-1. 35% et 41% en termes de F1 sur les attributs numériques et nominaux consécutivement. On note ici, qu’on refait l’expérience sur des attributs (termes) catégoriques tandis qu’en premier c’était sur des attributs numériques.

Tableau4. 2 Les résultats de performances de Naïve Bayes sur des Attributs numériques.

	TP Rate	FP Rate	Precision	Recall	F1
Class 10,291	0,545	0,482	0,291	0,363	
Class 0	0,455	0,709	0,269	0,455	0,338

Avg.	0,351	0,605	0,404	0,351	0,354

Tableau4. 3 Les résultats de performances de Naïve Bayes sur des Attributs nominales.

TP Rate	FP Rate	Precision	Recall	F-Measure
Class1	0,638	0,907	0,550	0,638	0,591
Class0	0,093	0,362	0,129	0,093	0,108
Avg.	0,439	0,708	0,396	0,439	0,415

La deuxième partie de l'expérimentation, toujours sur les données Bow, a comme objet d'améliorer les performances de la classification en utilisant les techniques de réduction de la dimensionnalité. Dans cette optique, nous allons tester l'impact des algorithmes de recherche sur les résultats de performances le classificateurs NB. Nous allons appliquer les deux algorithmes de recherche suivants : Optimisation par essaim de particules (PSO) et les Algorithmes génériques (GA).

Le tableau 4.4 montre les résultats de classification de NB en appliquant le premier algorithme de recherche qui est PSO, tandis que le deuxième tableau 4.5 est consacré pour l'algorithme de recherche les algorithmes génétiques GAs avec le classificateur NB.

Tableau4. 4 . Les résultats de performances de Naïve Bayes et l'algorithme PSO

TP Rate	FP Rate	Precision	Recall	F-Measure
0,997	1,000	0,634	0,997	0,775
0,000	0,003	0,000	0,000	0,000
Weighted Avg.	0,633	0,636	0,403	0,633	0,492

Tableau4. 5 Les résultats de performances de Naïve Bayes et l'algorithme GAs.

TP Rate	FP Rate	Precision	Recall	F-Measure
1,000	1,000	0,635	1,000	0,777

	0,000	0,000	0,000	0,000	0,000
Weighted Avg.	0,635	0,635	0,403	0,635	0,493

Les résultats de performance montrent l'avancée de l'algorithme NB avec les AG par rapport aux autres méthodes. L'algorithme PSO en revanche donne de bons résultats et est très compétitif par rapport aux GAs, enivrent 49%. Il convient de noter que jusqu'à 8% de gain de performance (score F1) est obtenu en appliquant ces algorithmes de recherche PSO et GAs, par rapport aux données d'origine.

4.8 Approche syntactique

Dans cette expérimentation, nous suivons approche linguistique qui est l'étiquetage morphosyntaxique, voir la section dans chapitre 2. Cette phase de prétraitement sera suivie d'une phase de réduction de dimensionnalité par les GAs et l'algorithme PSO. A la fin du processus de classification l'algorithme de classification NB est appliqué sur les données finales. Le tableau 4.6 montre les étiquettes utilisées dans cette étude.

Tableau4. 6 Etiquettes utilisées dans ce travail.

N°	Etiquète	Sens
1	TextID	text file name
2	URL	link to article
3	Label	objective vs. Subjective
4	totalWordsCount	total number of words in the article
5	Semanticobjscore	Frequency of words with an objective SENTIWORDNET score
6	Semanticsubjscore	Frequency of words with a subjective SENTIWORDNET score
7	CC	Frequency of coordinating conjunctions
8	CD	Frequency of numerals and cardinals
9	DT	Frequency of determiners
10	EX	Frequency of existential there
11	FW	Frequency of foreign words
12	INs	Frequency of subordinating preposition or conjunction
13	JJ	Frequency of ordinal adjectives or numerals
14	JJR	Frequency of comparative adjectives
15	JJS	Frequency of superlative adjectives
16	LS	Frequency of list item markers

17	MD	Frequency of modal auxiliaries
18	NN	Frequency of singular common nouns
19	NNP	Frequency of singular proper nouns
20	NNPS	Frequency of plural proper nouns
21	NNS	Frequency of plural common nouns
22	PDT	Frequency of pre-determiners
23	POS	Frequency of genitive markers
24	PRP	Frequency of personal pronouns
25	PRP$	Frequency of possessive pronouns
26	RB	Frequency of adverbs
27	RBR	Frequency of comparative adverbs
28	RBS	Frequency of superlative adverbs
29	RP	Frequency of particles
30	SYM	Frequency of symbols
31	Tos	Frequency of "to" as preposition or infinitive marker
32	UH	Frequency of interjections
33	VB	Frequency of base form verbs
34	VBD	Frequency of past tense verbs
35	VBG	Frequency of present participle or gerund verbs
36	VBN	Frequency of past participle verbs
37	VBP	Frequency of present tense verbs with plural 3rd person subjects
38	VBZ	Frequency of present tense verbs with singular 3rd person subjects
39	WDT	Frequency of WH-determiners
40	WP	Frequency of WH-pronouns
41	WP$	Frequency of possessive WH-pronouns
42	WRB	Frequency of WH-adverbs
43	Baseform	Frequency of infinitive verbs (base form verbs preceded by “to”)
44	Quotes	Frequency of quotation pairs in the entire article
45	Questionmarks	Frequency of questions marks in the entire article
46	Exclamationmarks	Frequency of exclamation marks in the entire article
47	Fullstops	Frequency of full stops
48	Commas	Frequency of commas
49	Semicolon	Frequency of semicolons
50	Colon	Frequency of colons

51	Ellipsis	Frequency of ellipsis
52	pronouns1st	Frequency of first person pronouns (personal and possessive)
53	pronouns2nd	Frequency of second person pronouns (personal and possessive)
54	pronouns3rd	Frequency of third person pronouns (personal and possessive)
55	Compsupadjadv	Frequency of comparative and superlative adjectives and adverbs
56	Past	Frequency of past tense verbs with 1st and 2nd person pronouns
57	Imperative	Frequency of imperative verbs
58	present3rd	Frequency of present tense verbs with 3rd person pronouns
59	present1st2nd	Frequency of present tense verbs with 1st and 2nd person pronouns
60	sentence1st	First sentence class
61	Sentencelast	Last sentence class
62	Txtcomplexity	Text complexity score

4.8.1 Les résultats de deuxième expérimentation : approche syntactique

Les résultats de performance montrent l'avantage de cette approche dont on a plus de 81 % en termes de F score en appliquant Nb sur les 60 attributs d'origine. Rappelons que la première approche BoW on a eu 41% de performance.

Tableau4. 7 résultats de performances de Naïve Bayes 60 attr

60 attr.	TP Rate	FP Rate	Precision	Recall	F-measure
NB	0,674	0,124	0,757	0,674	0,713
	0,876	0,326	0,824	0,876	0,849
	0,802	**0,252**	**0,799**	**0,802**	**0,799**

En deuxième lieu nous avons appliqué une réduction de la dimensionnalité en appliquant l'algorithme de recherche PSO et la méthode de recherche qui est les algorithmes génétiques GAs. Le tableau 4.8 montre les résultats de classification de PSO avec NB et de l'autre cote le tableau 4.9 NB avec GAs.

Tableau4. 8 les résultats de classification de PSO avec NB

GA	TP Rate	FP Rate	Precision	Recall	F-measure

NB	0.690	0.109	0.785	0.690	0.735
	0.891	0.310	0.834	0.891	0.861
	0.818	**0.236**	**0.816**	**0.818**	**0.815**

Tableau4. 9 résultats de classification de GAs avec NB

PSO	TP Rate	FP Rate	Precision	Recall	F-Measure
NB	0.685	0.104	0.791	0.685	0.734
	0.896	0.315	0.832	0.896	0.863
	0.819	**0.238**	**0.817**	**0.819**	**0.816**

Les résultats de performance montrent toujours l'avancée de l'algorithme NB avec les AG et PSO par rapport aux données initiales, sans réduction de dimensionnalité. En effet, 1% d'amélioration est constaté. Le meilleur score est enregistré par GAs est de 81.6 % en terme de F1 score.

4.8.2 Approche 3 : Hybride

La stratégie hybride présente le but final de notre étude actuelle. Elle vise à profiter des avantages des deux précédentes stratégies BoW et Morpho-syntactique à la fois. Dans ce but, on a fusionné les deux jeux de données en un seul jeu de données et qui pressente les données finales de prétraitement. Pour le classificateur NB est toujours appliqué en premier lieu sur les attributs d'origine. Ensuite, les algorithmes de recherche GAS et PSO interviennent dans la phase de réduction de dimensionnalité afin de montrer leurs impacts sur les résultats de performances. Les tableaux 4.10 et 4.11 montrent deux expérimentations : une avec 200 attributs et la deuxième en appliquant 50 attributs résultant de la mesure de Gain d'information.

Tableau4. 10 . Les résultats de performances de Naïve Bayes 200 attr.

TP Rate	FP Rate	Precision	Recall	F-Measure
0,871	0,337	0,818	0,871	0,844

	0,663	0,129	0,747	0,663	0,702
Weighted Avg.	0,795	0,261	0,792	0,795	0,792

Tableau4. 11 Les résultats de performances de Naïve Bayes et 50 top attr.

	TP Rate	FP Rate	Precision	Recall	F-Measure
	0,874	0,332	0,821	0,874	0,847
	0,668	0,126	0,753	0,668	0,708
Weighted Avg.	0,799	0,256	0,796	0,799	0,796

Les résultats de performance montrent plus de 79 % en termes de F1 score en appliquant Nb sur les deux jeux de données utilisés. On constate, qu'elle est compétitive par rapport la deuxième approche et meilleur de l'approche BoW.

En réduisant la dimensionnalité avec l'algorithme de recherche PSO et la méthode de recherche qui est les algorithmes génétiques GAs. Les tableaux 4.12 et 4.13 montrent les résultats de classification de PSO et GAs avec le classificateur NB.

Tableau4. 12 Les résultats de performances de Naïve Bayes et PSO.

	TP Rate	FP Rate	Precision	Recall	F-Measure
	0,909	0,321	0,831	0,909	0,868
	0,679	0,091	0,810	0,679	0,739
Weighted Avg.	0,825	0,237	0,824	0,825	0,821

Tableau4. 13 Les résultats de performances de Naïve Bayes et Ga.

	TP Rate	FP Rate	Precision	Recall	F-Measure
	0,902	0,321	0,830	0,902	0,865
	0,679	0,098	0,800	0,679	0,735

Weighted Avg.	0,821	0,239	0,819	0,821	0,817

Enivrent 1% d'avancement en terme de F1 est enregistré dans les résultats de performance en appliquant l'algorithme NB avec les AG et PSO, et cela par rapport aux données initiales qui sont présentées avec 50 et 200 attributs. Un résultat de plus de 82 % en termes de F1 score est enregistré par l'algorithme PSO. Dans l'autre côté, GA a obtenu envient de 81% de F1 score mais inferieur de celui obtenu par PSO.

4.9 Conclusion

Ce chapitre nous a permis de conduire la partie expérimentale de notre projet. En effet, on a présenté la démarche suivie pour le processus de classification, les outils de développement utilisé ainsi que la base documentaire traité dans ce travail. Dans cette étude nous avons choisi une approche syntaxique lors du prétraitement de données. Nous avons testé trois jeux de données BoW, Morpho-syntactique. Et Hybride. De l'autre côté, nous avons choisi 2 algorithmes de recherche pour la partie sélection des attributs et nous avons comparés leur comportements. Rappelons, que ces algorithmes de recherche ont été combinés le classificateur N.Bayes. les résultats des performances ont montré clairement l'avance de l'algorithme PSO avec Naive Bayes. L'approche la plus efficace parmi l'ensemble traité est la deuxième, Morpho-syntactique.

Conclusion générale

Conclusion Générale

La fouille de texte en général et la classification automatique de textes en particulier constitue un domaine de recherche très actif et très bénéfique dans de nombreuses applications pratiques, surtout de nos jours où de gros volumes de textes sont disponibles sur le web.

La quantité considérable et la forme brute des textes disponibles rend la tâche de leur fouille en vue par exemple de les catégoriser plutôt complexe et faisant appel à plusieurs techniques et algorithmes situés à plusieurs niveaux. On parle justement d'étapes de prétraitements linguistiques, de réduction de dimensionnalité, de classification et de visualisation et d'interprétation des résultats. Il est à noter que chacune des étapes citées ci-dessus utilisent différentes techniques et fait appel à de multiples méthodes et algorithmes.

Dans ce travail, nous avons travaillé sur l'analyse de subjectivité d'articles de sport à l'aide d'une approche de classification textuelle. Outre le fait que nous avons suivi le pipeline général de ce type d'application avec une mise en œuvre particulière pour l'application d'analyse de subjectivité, nous avons aussi expérimenté une panoplie de méthodes aussi bien au niveau de sélection d'attributs pour réduire la dimensionnalité (cinq méthodes) qu'au niveau de la classification automatique proprement dite (trois classificateurs). Cette étude détaillée nous a permis de mener une comparaison entre ces différentes méthodes et de souligner les conclusions suivantes entre autres :

- La combinaison entre l'algorithme GA et la représentation BOW a été pratiquement toujours meilleures aux autres combinaisons.
- Les méthodes de sélection d'attributs est très utiles pour l'amélioration des performances des classificateurs. Ceci était notamment très clair dans notre étude pour le cas du classificateur bag of words.

Pour les travaux futurs, plusieurs pistes peuvent être explorées. On peut citer :

- Appliquer le système proposé à d'autres corpus de textes issus d'autres domaines.
- Etendre l'étude comparative à d'autres algorithmes de recherche et d'autres classificateurs.
- Généraliser le travail à d'autres langues, en particulier la langue arabe.
- Etc.

Liste Bibliographiques

Liste bibliographiques

[1]

BELAZZOUG Mouhoub «Apprentissage statistique pour l'extraction des relations à partir d'une base de données textuelle. », Mémoire doctorat, Université Ferhat Abbas - Sétif -1, juillet 2021

[2]

Hassan CHOUAIB «Sélection de caractéristiques: méthodes et applications », Mémoire doctorat, université Paris Descartes, juillet 2011

[3]

Mataalah Hocine « classification automatique de textes Orienté Agent » faculté des sciences – algerie2010-2011

[4]

https://fr.wikipedia.org/wiki/Analyse_s%C3%A9mantique_latente#:~:text=L'analyse%20s%C3%A9mantique%20latente%20(LSA,1988%20et%20publi%C3%A9e%20en%201990.

http://helios.mi.parisdescartes.fr/~vincent/siten/en/Publications/Conf_inter/pdf/chouaib.pdf

M.F. Caropreso, M. Fernandacnandad, S. Matwin&F. Sebastiani. *A Learner-Independent Evaluation of the Usefulness of Statistical Phrases for Automated Text Categorization*. Text Databases and Document Management: Theory and Practice, pp. 78–102 (2001).

V.R. Carvalho&W.W. Cohen. (2005). *On the collective classification of email "speech acts*.". In Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.345–352 (2005).

S. Chakrabarti, B. Dom, R. Agrawal &P. Raghavan. *Using Taxonomy, Discriminants, and Signatures for Navigating in Text Databases*. In In Proceedings of the 23rd VLDB Conference, pp. 446–455(1997).

C. Cortes & V. Vapnik. Support-Vector Networks. *Machine Learning*, *20*(3):273–297(1995).

M. Dorigo&T.Stützle.*Ant colony Optimization*.MIT Press(2004).

J.C. Gomez, E. Boiy &M.F. Moens. *Highly discriminative statistical features for email classification*. Knowledge and Information Systems, 31(1):23–53 (2012).

M. Hearst. *What Is Text Mining?*(2003).Retrieved from : *https://people.ischool.berkeley.edu/~hearst/text-mining.html*

D. Hull. *Improving Text Retrieval for the Routing Problem Using Latent Semantic Indexing*. In Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 282–291. New York, NY, USA: Springer-Verlag New York, Inc (1994).

R. Jalam. *Apprentissage automatique et catégorisation de textes multilingues*. Thèse de doctorat. University de Lumière Lyon 2 (2003).

Liste bibliographiques

J. Kennedy & R.C. Eberhart. *Particle Swarm Optimization*.IEEE international Conference on Neural Networks, pp. 1942–1948.Piscataway (1995).

J. Kennedy & R.C. Eberhart.*A Discrete Binary Version of Particle Swarm Algorithm.*IEEE Conference on systems, man and cybernetics, pp. 4104–4109.Piscataway(1995).

J. Kogan. *Feature Selection and Document Clustering*. Survey of Text Mining, pp. 73–100. (2003).

K. Lang. *NewsWeeder: Learning to Filter Netnews*. InMachine Learning Proceedings, pp. 331–339, Elsevier(1995).

G. L'Huillier, A. Hevia, R. Weber&S. Ríos. *Latent semantic analysis and keyword extraction for phishing classification.* In International Conference on Intelligence and Security Informatics: Public Safety and Security, pp. 129–131(2010).

D.D. Lewis. *Representation and Learning in Information Retrieval*. PhD thesis, University of Massachusetts , Amherst, MA, USA (1992).

E. Miller, D. Shen, J. Liu&C. Nicholas. *Performance and Scalability of a Large-Scale N-gram Based Information Retrieval System*. Journal of Digital Information *1(21), (2000)*.

M.F. Porter. *An algorithm for suffix stripping*. Program 14(3):130–137(1980).

P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom &J. Riedl. GroupLens: *An open architecture for collaborative filtering of netnews*. In Proceedings of ACM Conference on Computer Supported Cooperative Work (CSCW-1994*)*, pp. 175–186 (1994).

Y.Rizk, M.Awad. *Syntactic Genetic Algorithm for a Subjectivity Analysis of Sports Articles*.International conference on Cybernitic Intelligent systems. Limerick, Ireland (2012).

G. Salton &M.J. McGill. *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, Inc(1986).

G. Salton &C. Buckley. *Term-weighting approaches in automatic text retrieval.* Information Processing & Management,*24*(5):513–523(1988).

H. Schmid. *Probabilistic Part-of-Speech Tagging Using Decision Trees*. Proceedings of International Conference on New Methods in Language Processing (1994)

J. Verbeek. *Supervised feature extraction for text categorization*.10th Belgian-Dutch Conference on Machine Learning (Benelearn '00) (2000).

X.S. Yang.*Nature-Inspired Metaheuristic Algorithms.*Luniver Press, UK(2008).

Y. Yang&J.O. Pedersen.*A comparative study on feature selection in text categorization.* In Machine Learning-International Workshop Then Conference, pp. 412–420 (1997).

L. Zhang&B. Liu. Sentiment Analysis and Opinion Mining. In *Encyclopedia of Machine Learning and Data Mining*, pp. 1152–1161, Springer US(2017).