République Algérienne Démocratique et Populaire
Ministère de L'enseignement Supérieur et de la Recherche Scientifique
Université Amar Télidji Laghouat

Faculté Des Sciences
Département de Mathématiques et Informatique
Mémoire présenté pour l'obtention du diplôme de Magister en Informatique
Option : Informatique Répartie et Mobile
École doctorale STIC
Thème :

Étude comparative de classifieurs pour les textes arabes

Présenté par : M^{me} Charef Fatiha

Soutenu devant le jury composé de :

M. LAGRAA Nasereddine	MC	Université de Laghouat	Président
M. YAGOUBI Mohamed Bachir	Professeur	Université de Laghouat	Examinateur
M. BENSAAD Mohamed Lahcen	MC	Université de Laghouat	Examinateur
M. OUINTEN Youcef	MCA	Université de Laghouat	Rapporteur

Année universitaire 2015/2016

Remerciements

J'adresse tout d'abord, ma profonde reconnaissance et mes vifs remerciements à M. OUINTEN Youcef pour avoir accepté de diriger cette thèse. Je le remercie pour son soutien et la patience dont il a fait preuve à mon égard.

Mes remerciements et mes sincères salutations s'adressent aux membres du jury M. LAGRAA Nasereddine, M. YAGOUBI Mohamed Bachir, et M. BENSAAD Mohamed Lahcen, qui m'ont honoré en prenant soin d'évaluer ce travail .

Je remercie également tous mes enseignants qui ont participé à ma formation Mme BOUZOUAD Hadda, Mme KARROUCHE Baya, M. DJOUDI Mohamed.

Enfin, mes derniers remerciements, mais non les moindres, vont à ma famille, à mes amis et mes stagiaires pour la confiance qu'ils m'accordent, leur amour, et leurs encouragements .

Sommaire

Table des figures

Liste des tableaux

Résumé

La généralisation de l'utilisation de l'Internet et des Technologies de l'Information et de la Communication (TIC) a engendré une quantité d'informations textuelles phénoménales. Pour permettre une bonne exploitation de cette masse d'information, la classification automatique de textes constitue un des traitements les plus importants car elle permet un accès plus rapide à l'information classée. Dans le cas de textes arabes la particularité de cette langue qui réside dans sa richesse morphologique entraine des difficultés supplémentaires dans son traitement. Ainsi, nous avons mené une étude comparative de différentes approches de représentation de textes Bag of Words, Light Stemming et le Stemming, elle vise à tester les performances et l'efficacité du classifieur Support Vector Machines(SVM) sur les textes arabes.

Mots clés : La classification automatique, Light Stemming, Stemming, SVM.

Abstract

The rapid growth of the Internet and Information Technology and Communication (ITC) has generated a phenomenal amount of textual information. This has led to the development of automated text classification systems that are capable of automatically organizing and classifying documents. Automatic text classification aims to automatically assign text to a predefined category based on their content. In case of Arabic texts, arabic language is a very rich language with complex morphology, so it has a very different and difficult structure. So a comparative study of different texts representation approaches Bag of Words, Light Stemming Stemming, it aims to test the performance and efficiency of the classifier Support Vector Machines (SVM) on the Arabic texts.

Keywords : Arabic text classification, Light Stemming, Stemming, SVM.

Introduction

Avec l'avènement de l'Internet et les avancées technologiques dans les systèmes de communication électronique, la quantité d'informations disponibles croît de façon extrêmement rapide, particulièrement les données textuelles. Pour cette raison, il devient de plus en plus important de disposer d'un accès intelligent aux données et leurs manipulations, d'une part. Et d'autre part convaincre les limites du travail manuel qui est coûteux en temps, et relativement peu efficace. Dans ce contexte, la classification de textes, définie comme le processus permettant d'assigner une classe(ou catégorie) à un texte, en fonction des informations qu'il contient. Plusieurs applications sont concernées par la classification automatique de textes, prenons à titre d'exemples : l'indexation des documents, organisation ou classement des articles d'un journal, la détection de courrier indésirable, et la catégorisation hiérarchique des pages Web.

La classification automatique de textes a débuté dans les années 1960 et a connu des avancées considérables à partir des années 90 avec l'apparition des algorithmes beaucoup plus performants dans le domaine de l'apprentissage automatique ML(Machine Learnnig) et recherche d'information IR(Information Retrieval).

Concernant la catégorisation automatique de textes arabes, elle est apparue en retard et peu de travaux se rapportent à ce domaine, malgré la richesse morphologique de cette langue, et ses variations orthographiques liées au phénomène d'agglutination des lettres.

Problématique

La problématique qui nous occupe dans ce travail consiste à utiliser l'apprentissage automatique pour affecter des catégories à des documents [1] écrits en caractères arabes en fonction de leur contenu. La catégorisation [2] de textes consiste à chercher une liaison fonctionnelle entre un ensemble de textes et un ensemble de catégories (étiquettes, classes).

La première étape à suivre pour identifier la catégorie à laquelle un texte est associé est la disposition d'un ensemble de textes étiquetés, ou corpus d'apprentissage. La majorité des travaux a été réalisée sur un seul corpus et généralement de taille réduite. D'où la nécessité de tester la fiabilité et l'efficacité des solutions existantes sur des corpus différents.

1. les mots document et texte représentent le même concept
2. les mots catégorisation et classification ont le même sens dans le contexte de notre travail

La deuxième étape, le choix d'un descripteur [3] pour la représentation de l'espace d'apprentissage, c'est une étape critique pour la classification automatique de textes. Afin de faire face aux difficultés morphologiques reconnues dans la langue arabe(non-vocalisation, agglutination, flexions et dérivation), on veut tester une approche qui se base sur une analyse morphologique pour extraire les descripteurs de textes .

Le choix de l'algorithme d'apprentissage se présente comme un autre élément essentiel pour une classification automatique efficace, plusieurs algorithmes d'apprentissages ont été mis au point, la méthode Rachio, K-Nearest Neighbor(KNN), Naïve Bayes(NB), Support Vector Machines(SVM), ...etc. La plupart des travaux menés sur la classification automatique de textes arabes confirment la performance de l'algorithme d'apprentissage Support Vector Machines(SVM).
Dans notre étude, nous avons choisi de tester Support Vector Machines(SVM) sur la classification automatique des textes arabes.

Organisation du mémoire

Ce mémoire est composé de trois chapitres, après une introduction générale, le premier chapitre présente d'abord la définition de la classification et le déroulement du processus d'apprentissage. Il expose ensuite les difficultés qui caractérisent ce processus, et quelques applications de la catégorisation de textes. Puis nous décrivons les principales approches de représentation de textes utilisés et les techniques appliquées pour la réduction de la dimension de l'espace d'apprentissage et le codage des termes. Enfin nous décrivons les algorithmes d'apprentissages ayant fait leurs preuves dans le domaine de la classification et la manière d'évaluation des classifieurs.

Le deuxième chapitre est consacré à la classification automatique des textes arabes, nous décrivons en premier lieu les caractéristiques morphologiques et difficultés de la langue arabe. Nous citons quelques outils pour le traitement automatique de la langue arabe. Par la suite, nous détaillons les travaux antérieurs de la classification automatique des textes arabes.

Dans le dernier chapitre nous exposons nos expérimentations. Nous décrivons d'abord les corpus utilisés. Puis nous discutons les différents résultats obtenus en appliquant les différentes approches pour la représentation de l'espace d'apprentissage pour choisir le descripteur avec le classifieur SVM sur les différents corpus .

Enfin, nous terminons le mémoire par donner une synthèse générale avec les perspectives de notre travail.

3. les mots descripteur, terme et attribut désignent la même chose dans la classification automatique de textes

Chapitre 1

Classification automatique de textes

A l'intersection de la recherche d'information IR(Information Retrieval) et l'apprentissage automatique ML(Machine Learnnig), la classification automatique de textes prend sa place. Elle partage également des caractéristiques avec la Fouille de Textes(Texte Mining). Voir la figure 1.1.

L'apprentissage automatique permet de construire automatiquement un classifieur d'une maniére inductive à partir d'un ensemble de textes pré-étiquetés et permet aussi l'évaluation des performances de ce dernier. La Fouille de Textes est un ensemble de traitements automatiques consistant à extraire des connaissances à partir de textes [25].

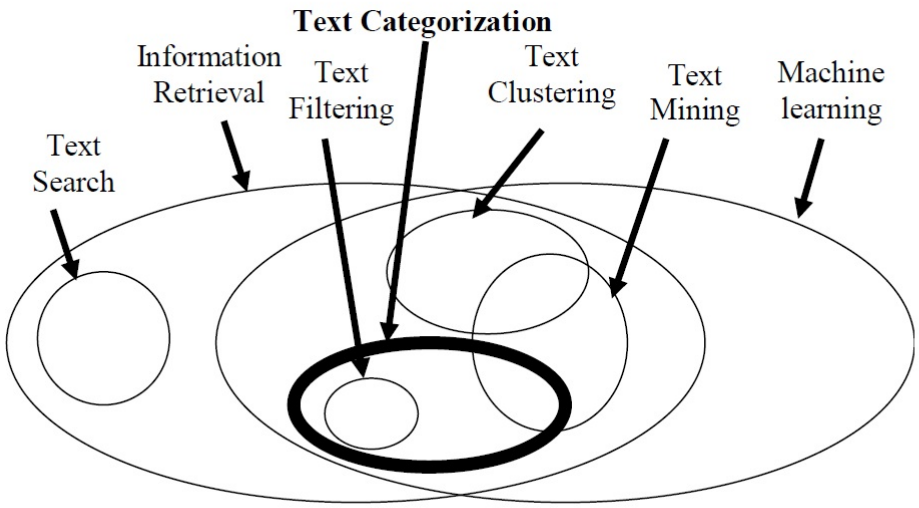

FIGURE 1.1: La classification de textes à l'intersection de la recherche d'information et l'apprentissage automatique [15]

La classification automatique de textes qui remonte au début des années 60 a connu des progrès considérables à partir des années 90 avec l'apparition des algorithmes beaucoup plus performants dans le domaine ML. Ces évolutions technologiques et leurs algorithmes avancés font aujourd'hui de la catégorisation un outil fiable.
Au début des années 90, les travaux proviennent essentiellement de la communauté de recherche d'information(IR). En effet, les méthodes de numérisation, les algorithmes de classification et les méthodologies de test ont été adaptés à la catégorisation des textes [25].

On distingue dans le domaine de la classification automatique deux types d'approches, qui diffèrent sur la façon dont les classes sont générées[34] :

- La classification(catégorisation) automatique des documents d'une manière *supervisée* correspond à la procédure d'assigner d'une façon autonome et automatique des documents à une ou plusieurs catégories prédéfinies(exemple Politique, Économie, Sport,...etc).
- La manière dite *non-supervisée(clustering)*, quant à elle, ignore les catégories de sortie et c'est à l'algorithme d'apprentissage d'analyser les documents pour les concevoir.

C'est dans le contexte de la première approche que s'inscrit notre travail de recherche.

1.1 Définition de la classification automatique de textes

La classification de Textes(CT) consiste à l'affectation d'une ou plusieurs catégories parmi une liste prédéfinie à un texte. L'objectif du processus est d'être capable d'assigner automatiquement les classes d'un ensemble de nouveaux textes.

Selon Jalam dans [18]. *La catégorisation de textes consiste à chercher une liaison fonctionnelle entre un ensemble de textes et un ensemble de catégories (étiquettes, classes). Cette liaison fonctionnelle, que l'on appelle également modèle de prédiction, est estimée par un apprentissage automatique(traduction de machine learning method).*

Formellement, la catégorisation de textes consiste à associer une valeur booléenne à chaque paire $(\mathrm{d}_j, c_i) \in D * C$ où D est l'ensemble des textes et C est l'ensemble des catégories. La valeur V(Vrai) est alors associée au couple (d_j, c_i) si le texte d_j appartient à la classe c_i tandis que la valeur F(Faux) lui sera associée dans le cas contraire. Le but de la catégorisation de textes est de construire une procédure (modèle,classifieur)
$\Phi : D \to C \in \{V, F\}$ qui associe une ou plusieurs étiquettes (catégories) à un texte d_j telle que la décision donnée par cette procédure coïncide le plus possible avec la fonction $\Phi : D \to C \in \{V, F\}$, la vraie fonction qui retourne pour chaque vecteur d_j une valeur c_i. Pour ce faire, l'algorithme doit se disposer d'un ensemble de textes préalablement étiquetés, que l'on appelle ensemble d'apprentissage, à partir duquel le modèle de prédiction le plus fidèle est estimé c'est-à-dire le modèle qui génère le moins d'erreurs en prédiction [38].

1.1.1 Les différents types de classification

- ***La classification bi-classe :***
  C'est une problématique pour laquelle la classification bi-classe correspond au filtrage. Le système de classification répond à la question « Le texte appartient-il à la classe C ou non ? » (Par exemple, un document est un sparm ou non).
- ***La classification multi-classes :***
  La classification multi-classes permet de transmettre le texte vers le ou les catégories(s) le(s) plus approprié(s), on parle alors de routage. Cette classification multi-classes, peut être disjointe ou non.

  La classification multi-classes disjointes : consiste à attribuer un texte à une et une seule catégorie. Un système de classification multi-classes disjointes répond à la question « A quelle classe (au singulier) appartient le texte ? ».

  La classification multi-classes : consiste à associer un texte à une ou plusieurs catégories voire à aucune catégorie. Le système répond donc à la question : « A quelles classes (au pluriel) appartient le texte ? ».

- ***La classification déterministe :***Le but de la classification déterministe est d'avoir une réponse définitive pour chaque texte(le texte T appartient à la classe C, oui ou non).

- ***La classification floue ou le ranking :***
  Contrairement aux cas précédents au lieu d'associer un texte à une classe, le système ordonne les classes par ordre de pertinence pour un texte donné. On peut également souhaiter dans certains cas d'avoir simplement une évaluation des classes les plus adéquates dans l'ordre pour y classer le texte. Ce type de classification va permettre à l'utilisateur de savoir si le texte est "proche" du thème que si le texte n'a absolument rien à voir avec celui-ci dans le cas ou ce dernier est incorrectement attribué à la classe. Les techniques qui évaluent une distance d'un texte à une catégorie permettent facilement ce type de classement de même pour les approches qui estiment des probabilités d'appartenance d'un texte à une classe.

1.1.2 les démarches à suivre pour la classification de textes

Le processus de catégorisation comporte deux phases :

1. **L'apprentissage** : le modèle de prédiction est construit à partir d'un ensemble de textes préalablement étiquetés, qui en entrée reçoit un texte et, en sortie lui associe une ou plusieurs catégories. Les étapes à suivre pour aboutir au modèle de prédiction sont :
   (a) la disposition d'un ensemble des textes étiquetés corpus d'apprentissage(en anglais, « dataset »), est un élément essentiel à la construction d'un système de classification automatique. Plusieurs sites web proposent gratuitement des corpus pour réaliser les recherches portant sur la catégorisation automatique des textes, exemple le corpus Reuters 21578 [1].

1. Le corpus Reuters-21578 est un ensemble de dépêches financières émises au cours de l'année 1987 par l'agence Reuters, en langue anglaise, et disponible gratuitement sur le web,il est très utilisés par les

(b) la représentation de l'espace d'apprentissage consiste a extraire les k descripteurs $(t_1; ..; t_k)$ les plus pertinents(mot, racine, lemme, concept,... etc) à partir du corpus. Nous disposons alors d'un tableau « individus descripteurs » où les lignes correspondent aux textes à catégoriser et les colonnes correspondent à leurs termes, et pour chaque texte nous connaissons la valeur de ses descripteurs et son étiquette(classe).

(c) Un des problèmes majeurs qu'on peut rencontrer lors de l'apprentissage est la grande dimensionnalité de l'espace d'apprentissage. Pour cela, il est indispensable de réduire la taille de ce vocabulaire, avant de pouvoir l'utiliser, en appliquant plusieurs techniques qui seront présentées en détail dans le chapitre suivant.

(d) plusieurs algorithmes d'apprentissages ont été mis au point, afin de construire le modèle de prédiction, on peut citer la méthode de Rocchio, les plus proches voisins(PPV), bayésien naïf(BN), les machines à vecteurs de support(SVM),...etc.

2. **Le classement** : d'un nouveau texte comporte la représentation puis l'application du modèle de prédiction afin de prédire l'étiquette de ce texte.

figure 1.2 résume les démarches à suivre pour la catégorisation des textes.

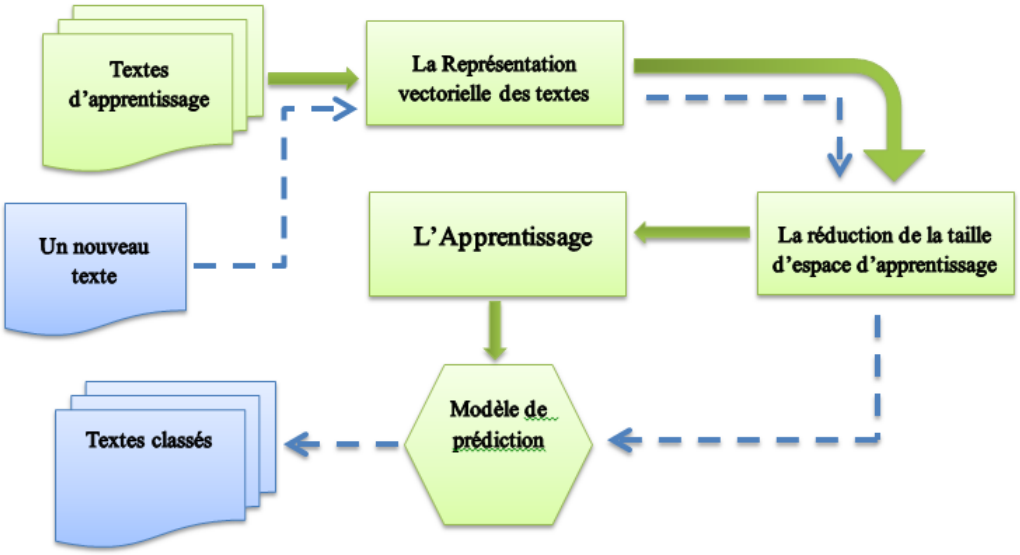

FIGURE 1.2: Le processus de classification de textes(les flèches en vert représentent le processus d'apprentissage, et les flèches en bleu représentent le processus de classement d'un nouveau texte)

chercheurs

1.1.3 Applications de la classification de textes

Classer les textes revient à les organiser par différentes thématiques. Cependant, l'évolution des besoins des utilisateurs dans lesquelles les catégories ne sont pas interprétables, comme le classement de textes par auteur, par genre, par style, par langue, ou encore selon que le texte exprime un jugement positif ou négatif rend la problématique de classification plus large.

L'indexation des documents : Les premières applications concernées étaient l'indexation automatique pour les systèmes de Recherche d'Information (IR). A chaque document est attribué un ou plusieurs mots ou expressions clés parmi un ensemble prédéfini. L'objectif est de décrire le contenu de ces textes par des mots ou des phrases clés qui font partie d'un ensemble de vocabulaire contrôlé [2]. Dans un tel contexte, si nous regardons ce vocabulaire contrôlé comme des catégories, l'indexation de textes peut être alors vue comme une forme de catégorisation de textes [18], [21],[40]. Dans les bibliothèques numériques, nous sommes souvent plus intéressés par le marquage des documents par des méta-données qui les décrivent sous différents aspects (par exemple, date de création, type de document ou le format, disponibilité,...etc). Le rôle de certaines de ces méta-données est de décrire la sémantique du document de la signification des codes bibliographiques, des mots-clés ou des phrases-clés[38].

Organisation ou classement des documents : Plusieurs problèmes relatifs à l'organisation du document peuvent être réglés par les techniques de TC. Dans les bureaux d'un journal, par exemple, les annonces doivent être classées dans les catégories telles que les rencontres, voitures à vendre, immobilier,... etc. Avant les publications, les journaux avec un grand nombre d'annonces bénéficieraient d'un système automatique qui pourrait choisir pour une annonce la catégorie donnée la plus appropriée. D'autres applications possibles sont les applications d'organisation des brevets en catégories pour rendre leur recherche plus facile. Le classement automatique des articles de journaux sous les sections appropriées(par exemple, la politique, événements, styles de vie,...etc.)[38].

Le Filtrage et le routage des documents : Le routage ne consiste pas à classer les documents pour les retrouver plus tard, mais de les envoyer aux personnes concernées, dans cette problématique de classification les classes représentent les rôles des personnes qui vont recevoir ces documents. Le système de routage doit bloquer l'envoi de tous les documents qui n'intéressent pas le récepteur [34].

Le filtrage est la classification des documents en deux catégories disjointes, la catégorie "pertinents" et la catégorie "non pertinents", exemple système de filtrage des mails peut filtrer les spam [6], [41], [27].

Désambiguïsation des mots : La WSD(Word Sense Disambiguation - WSD) est l'activité de recherche dans un texte des sens des mots ambigus. Un seul mot peut avoir plusieurs significations. La WSD peut décider de quel sens il s'agit. La WSD est importante

2. L'appellation vocabulaire contrôlé constitue la désignation générale de tout ensemble de termes, définis et sélectionnés par un ensemble d'experts, il est généralement mis au point de manière à couvrir et à décrire un ou plusieurs domaines particuliers. Son utilité est de permettre l'organisation des connaissances à des fins de recherche d'informations, exemple(taxonomie, thésaurus ou ontologie)[21]

dans le traitement du langage naturel et l'indexation des documents par le sens des mots. La WSD peut être considérée comme une tâche de TC si nous considérons le contexte d'occurrence des mots comme un document et le sens du mot comme une catégorie[38].

Catégorisation hiérarchique des pages Web : Le domaine de la catégorisation des pages ou sites Web s'intéresse aux techniques de la classification automatique de documents, qui lui permet de placer les pages ou sites Web dans une ou plusieurs catégories.

1.1.4 Problèmes de la classification de textes

Le domaine de la catégorisation de textes rencontre plusieurs difficultés. Des problèmes liés à l'apprentissage automatique supervisé comme la subjectivité de la décision prise par les experts, le sur-apprentissage,...etc. Mais aussi des problèmes liés à la nature des données textuelles traitées comme la polysémie, la redondance, les variations morphologiques,...etc.

Redondance(Synonymie) : La redondance et la synonymie permettent d'exprimer le même concept par des mots différents, par exemple « document » et « texte » ou « classification » et « catégorisation ». Lors d'une représentation vectorielle d'un document, ces termes sont représentés séparément, et les occurrences du concept sont dispersées. Il est alors important de rassembler ces termes en un groupe sémantique commun.

Ambiguïté (Polysémie) : à la différence des données numériques, les données textuelles sont sémantiquement riches, du fait qu'elles sont conçues et raisonnées par la pensée humaine. Pour le même mot plusieurs définitions lui sont associées, par exemple le mot Avocat peut désigner le fruit ou le juriste.

Les variations morphologiques : Les différentes variations morphologiques (conjugaisons, pluriels,...etc.) influent négativement sur la qualité des résultats puisque elles vont être considérées séparément par exemple « maître », « maîtresse » et « maîtriser ». Pour y remédier on applique les techniques de lemmatisation, Stemming, ou les N-grammes.

Subjectivité de la décision : Parmi les problèmes classiques usuels dans le domaine de l'apprentissage supervisé c'est la subjectivité de la décision prise par les experts qui décident de la classe à laquelle le texte va être attribué.
Ainsi un même document peut être classé différemment par deux experts, ou encore un même document peut être classé différemment par le même expert, soumis à deux instants différents [38].

Déséquilibrage : Dans la pratique, les effectifs des classes sont souvent déséquilibrés et, pour certaines classes, le nombre d'exemples positifs est faible comparé à celui des exemples négatifs. Ceci crée une difficulté supplémentaire car les classes peu nombreuses sont mal représentées. Une solution proposée pour pallier à ce problème est d'utiliser les techniques de redressement[18].

Complexité de l'algorithme d'apprentissage : La représentation des textes se fait généralement sous forme d'un vecteur contenant les nombres d'apparitions des termes dans ce texte. Or, le nombre de textes qu'on va traiter est très important sans oublier

le nombre de termes composant le même texte donc on peut bien imaginer la dimension du tableau (textes * termes) à traiter qui va compliquer considérablement la tâche de classification en diminuant la performance du système. De ce fait, une réduction de la taille du tableau est primordiale avant d'entamer l'apprentissage.

Sur-apprentissage : Le nombre de termes très important et très varié qui ne se répètent dans tous les textes va causer énormément de creux dans le tableau de grande dimension (textes*termes) qui peut provoquer du sur-apprentissage. Qui s'explique par le fait que le modèle n'arrive pas à bien classer les nouveaux textes. Ce phénomène se produit lorsqu'un classifieur devient trop spécifique aux données d'apprentissage. Il devient très performant lorsqu'il s'agit de classer les documents de son ensemble d'apprentissage, mais quand vient le temps de traiter d'autres textes, son efficacité diminue. En fait, on est en présence d'un manque de généralisation. Plutôt que de retenir seulement les caractéristiques générales d'une catégorie, le classifieur tient aussi compte des particularités des données d'apprentissage. C'est évidemment une situation à éviter, car cela fait obstacle à une amélioration de la performance des classifieurs. Pour limiter le sur-apprentissage, on doit sélectionner des termes pour réduire la dimensionnalité. Expérimentalement, pour éviter le sur-apprentissage, on doit limiter le nombre de termes en fonction du nombre d'exemples dans l'échantillon d'apprentissage. Dans la pratique, comme on dispose d'un nombre limité d'exemples d'apprentissage, on tend à réduire le nombre des termes utilisé pour éviter ce sur-apprentissage. La réduction de dimension doit cependant être utilisée avec précaution pour ne pas supprimer des termes pertinents[38].

La classification de textes est l'activité de l'étiquetage des textes avec des catégories de thématiques prédéfinies a essentiellement progressé ces dix dernières années grâce à l'introduction des techniques héritées de l'apprentissage automatique qui ont amélioré très significativement les taux de bonne classification . La catégorisation est un domaine entre l'apprentissage automatique et la recherche d'information . Elle partage un certain nombre de caractéristiques avec d'autres tâches telles que l'extraction de connaissances à partir de textes et la Fouille de Textes.
Nous avons montré que la classification de textes est une tache qui a été adoptée par différentes applications l'indexation, le classement, le routage et le filtrage,.. etc. Est un processus qui possède plusieurs difficultés liés à l'apprentissage automatique supervisé comme le sur-apprentissage, la grande dimensionalité, la subjectivité de l'attribution d'un texte à une telle ou telle catégorie. Et même à la nature des données traitées (la synonymie, la polysémie,...etc).

1.2 La représentation des textes

Il n'existe pas à l'heure actuelle une méthode d'apprentissage artificiel capable d'exploiter directement les documents de l'espace d'apprentissage dans leur état d'origine. Ces derniers doivent être représentés sous une autre forme qui permet à l'algorithme d'apprentissage de les utiliser. Cette étape consiste généralement à représenter chaque document par un vecteur, dont les composantes sont les termes contenus dans le texte, à ces termes on associe des poids pour rendre chaque vecteur exploitable par les algorithmes d'apprentissage, et enfin réduire la dimensionnalité.

Une collection de textes peut être ainsi représentée par une matrice dont les lignes sont les termes qui apparaissent au moins une fois et les colonnes sont les documents de cette collection. L'entrée w_{kj} est le poids du terme t_k dans le document d_j.

1.2.1 Le prétraitement

Avant la construction des vecteurs pour représenter les textes, il est préférable d'effectuer quelques prétraitements :

La segmentation : La première opération que doit effectuer un système de classification est la reconnaissance des termes utilisés. La segmentation consiste à découper la séquence des caractères afin de regrouper les caractères formant un même mot. Habituellement, cette étape permet d'isoler les ponctuations(reconnaissance des fins de phrase ou de paragraphe), ensuite découper les séquences de caractères en fonction de la présence ou l'absence de caractères de séparation(de type « espace », « tabulation » ou « retour à la ligne »), puis regrouper les chiffres pour former des nombres(reconnaissance éventuelle des dates), de reconnaitre les mots composés. Éventuellement, nous pouvons unifier les écritures en lettres majuscules ou en lettres minuscules avant ou après les opérations déjà indiquées. C'est un traitement de surface assez simple dans le principe, mais particulièrement difficile à réaliser de manière exacte sur les documents ayant beaucoup de bruit et des représentations assez variées.

Suppression des Mots Outils : Une autre opération consiste à supprimer les mots faisant partie d'une liste prédéfinie : les stops words. Ce sont des mots génériques non porteurs de sens tels que les articles, les prépositions, les mots de liaisons, les déterminants, les adverbes, les pronoms et les verbes auxiliaires...etc. A titre d'exemple on peut citer dans la langue Française, le cas des articles « le », « la », « les » ou de certains mots de liaison « ainsi », « toutefois » ...etc. Ou en Anglais : Les prépositions(about, after, through), les déterminants(the, no, one), les conjonctions(though, and, or), certains verbes(are, can, have, may,will). Et en Arabe(بين, هذه, من, كان, ليس)

Ces termes très fréquents peuvent être supprimés du corpus pour en réduire la dimension. Cette possibilité de réduire la taille du vocabulaire s'explique par deux raisons :

- D'un point de vue linguistique, ces mots ne comportent que très peu d'informations. La présence ou l'absence de ces mots n'aident pas à deviner le sens d'un texte. Pour cette raison, ils sont communément appelés « mots vides ».
- d'un point de vue statistique, ces mots se retrouvent sur l'ensemble des textes sans aucune discrimination et ne sont d'aucune aide pour la classification.

Suppression des mots rares : Sont des mots qui n'apparaissent qu'une ou deux fois sur un corpus, la suppression de ces mots réduit de façon appréciable la dimension des vecteurs utilisés pour représenter les textes. D'un point de vue linguistique, la suppression de ces mots n'est pas nécessairement justifiée, certains mots peuvent être très rares, mais très informatifs. Néanmoins, ces mots ne peuvent pas être utilisés par des méthodes à base d'apprentissage du fait de leur très faible fréquence, il n'est pas possible de construire de statistiques fiables à partir d'une ou deux occurrences. Une des méthodes communément retenue pour supprimer ces mots consiste à ne considérer que les mots dont la fréquence totale est supérieure à un seuil fixé préalablement.

1.2.2 Choix de descripteurs

Le choix de descripteurs est une étape très importante car ces derniers constituent la structure de l'espace dans lequel seront représentés les textes. Pour cette étape on transforme chaque document d_j en un vecteur $\mathrm{d}_j = (w_{1j}, w_{2j}, ..., w_{|T|j})$ où T est l'ensemble des termes (ou descripteurs) qui apparaissent au moins une fois dans le corpus d'apprentissage. Le poids w_{kj} correspond à la contribution des termes t_k à la sémantique du texte d_j. Notons que la représentation par un vecteur entraîne une perte d'information notamment celle relative à la position de mots dans la phrase[18]. Les descripteurs peuvent être les mots simples, les lemmes, les racines, les concepts ou les N-grammes.

- ***Représentation en « sac de mots » « Bag of Words » :*** Le choix des mots comme descripteurs d'un texte c'est le choix le plus intuitif. Un texte est représenté sous forme d'un vecteur dont chaque composante correspond au nombre d'apparition d'un mot dans le texte, cette représentation est connue par « sac de mots », « Bag-of-Words ». Cette représentation des textes exclut toute analyse grammaticale et toute notion de distance entre les mots : c'est pourquoi cette représentation est appelée "sac de mots" avec cette approche, les documents sont représentés par des vecteurs de dimension égale à la taille du vocabulaire, qui est en général assez grand. En effet, même des collections de documents de taille moyenne peuvent contenir de nombreux mots différents, et des vocabulaires de plusieurs dizaines de milliers de mots sont désormais communs. Or la grande dimension de ces données rend la plupart des algorithmes de classification difficiles à utiliser. À cette difficulté algorithmique vient s'ajouter le fait que les représentations des données textuelles sont typiquement creuses[18].

- ***Représentation avec des phrases :*** Au lieu d'utiliser les mots comme descripteurs on utilise les phrases, puisque elles sont plus informatives que les mots seuls, et elles ont l'avantage de conserver l'information relative à la position du mot dans la phrase. Logiquement, une telle représentation doit obtenir de meilleurs résultats que ceux obtenus via les mots. Mais les expériences présentées ne sont pas concluantes car, si les qualités sémantiques sont conservées, les qualités statistiques sont largement dégradées : le grand nombre de combinaisons possibles entraîne des fréquences faibles et trop aléatoires[18].

- ***Représentation des textes avec des racines lexicales (Stemming) :*** Le stemming ou la désuffixation regroupe sous un même terme(stem) les mots qui ont la même racine. L'extraction des stems se fait par la technique de racinisation (ou

Stemming) qui utilise à la place des dictionnaires, des algorithmes simples basés sur des règles de remplacement de chaînes de caractères pour supprimer les suffixes les plus utilisés. Le Stemming est un traitement linguistique moins approfondi que la lemmatisation, ayant deux avantages : plus rapide que la lemmatisation (algorithmes simples ne faisant pas référence aux dictionnaires et règles de dérivation) et la possibilité de traiter les mots inconnus sans traitement spécifique.

- ***Représentation des textes avec des lemmes (lemmatisation) :*** La lemmatisation conserve, non pas les mots eux-mêmes, mais leur racine ou lemme. Ce principe permet de prendre en compte les variations flexionnelles(singulier/pluriel, conjugaisons,...etc) ou dérivationnelles(substantifs, verbes, adjectifs,...etc) en regroupant sous le même terme tous les mots de la même famille et donc d'améliorer la classification. La lemmatisation est donc une tâche plus compliquée à mettre en œuvre que la recherche de racines, puisqu'elle s'appuie sur des outils de Traitement automatique du langage naturel(TALN), ce qui nécessite beaucoup de ressources linguistiques(dictionnaires, règles de dérivation,...etc). De plus les résultats contiennent encore des erreurs à cause des problèmes de polysémie(ambiguïté) et d'incomplétude des dictionnaires :
  - La perte de l'information donnée par le contexte syntagmatique, nécessaire à la distinction des lemmes polysémiques(« prix » n'a pas le même sens dans « prix Goncourt », « grand prix » ou « prix d'une marchandise »).
  - La présence de synonymes, considérés comme des lemmes différents même s'ils font référence au même concept(« mission » et « délégation » peuvent dénommer la même entité dans un article de journal) .

- ***Représentation des textes avec la méthode des N-grammes :*** Un N-gramme est une séquence de N caractères. Pour un document quelconque, l'ensemble des N-grammes est le résultat obtenu en déplaçant une fenêtre de N cases sur le texte, ce déplacement se fait par étapes de un caractère et à chaque étape on prend une photo, l'ensemble de ces photos donne l'ensemble de tous les N-grammes du document. Il y a plusieurs avantages à l'utilisation des N-grammes selon[18] :
  - les N-grammes capturent automatiquement les racines des mots les plus fréquents sans passer par l'étape de recherche des racines lexicales.
  - elles opèrent indépendamment des langues contrairement aux systèmes basés sur les mots dans lesquels il faut utiliser des dictionnaires spécifiques(féminin masculin, singulier-pluriel, conjugaisons,...etc.) pour chaque langue. De plus, avec les N-grammes, on n'a pas besoin de segmentation préalable du texte en mots, ceci est intéressant pour le traitement de langues dans lesquelles les frontières entre mots ne sont pas fortement marquées, comme le chinois, ou encore pour les séquences ADN en génétique.
  - elles sont tolérantes aux fautes d'orthographes et aux déformations causées lors de l'utilisation des lecteurs optiques. Lorsqu'un document est scanné, la reconnaissance optique est souvent imparfaite. Par exemple, il est possible que le mot "chapitre" soit lu comme "clapitre".
  - Enfin, ces techniques n'ont pas besoin d'éliminer les mots-outils(Stop Words) ni de procéder à la Stemming. Ces traitements augmentent la performance des systèmes basés sur les mots. Par contre, pour les systèmes N-grammes, de nombreuses études ont montré que la performance ne s'améliore pas après l'élimination des

"Stop Words" et de "Stemming".

- ***Représentation des textes basée sur les concepts :*** Les approches précédentes n'extraient pas la sémantique d'un document mais simplement une comparaison morphologique. Si on peut supposer que chaque terme a un sens, il est plus difficile de prouver que deux documents étant composés des mêmes termes aient forcément le même sens. Les auteurs proposent donc, une nouvelle approche de représentation textuelle « plus sémantique » basée non pas sur les termes présents sur le texte à traiter mais sur les concepts correspondants. Ainsi, au lieu de définir un espace vectoriel dont chaque composante représente un terme(mot, stem, lemme, ou N-gramme), on projette l'ensemble de termes du texte sur un ensemble fini de concepts (une liste de concepts). Cette liste peut être décrite dans un thésaurus, une ontologie, une hiérarchie de concepts,...etc.

1.2.3 Réduction de la dimensionnalité

Si on utilise directement le vocabulaire contenu dans les textes d'apprentissage et qu'on crée un attribut pour chaque mot qu'il contient, on se retrouve avec un espace vectoriel ayant une dimension très élevée. Chacun des textes sera représenté par un vecteur ayant autant de termes qu'il y a de mots dans le vocabulaire. Le traitement d'un tel espace vectoriel demanderait beaucoup de mémoire et de temps de calcul et pourrait nous empêcher d'utiliser des algorithmes de classification plus complexes. Utiliser tous ces mots influencerait aussi négativement la précision de la classification. Même après les prétraitements appliqués dans la première phase, qui ont procédé à l'élimination des mots les plus fréquents et les plus rares, soit parce qu'ils n'étaient pas discriminants(Mots vides très faiblement informatifs), soit parce qu'ils n'étaient pas exploitables statistiquement(très faible fréquence), le nombre de termes s'avère encore très élevé. Il faut utiliser d'autres méthodes pour choisir les mots utiles pour discriminer entre documents pertinents et documents non pertinents ou, plus généralement, entre les classes de documents.

Les techniques utilisées pour la réduction de dimension sont issues de la théorie de l'information et de l'algèbre linéaire. [38] classe ces techniques de deux façons :

- selon qu'elles agissent localement ou globalement,
- selon la nature des résultats de la sélection(s'agit-il d'une sélection de termes ou d'une extraction de termes).

a) **Réduction locale ou globale**
dépend de la « localité » de la réduction i.e si la réduction est réalisée localement ou globalement. Il faut noter que toutes les techniques de réduction du vocabulaire peuvent être appliquées localement ou globalement.

- *Réduction locale :*Si pour chaque catégorie C_i on propose un nouvel ensemble d'attributs T' avec $|T'_i| << |T_i|$, on parle d'une réduction locale du vocabulaire. Dans ce cas là, chaque document D_j sera présenté par un ensemble de vecteurs V_j différents selon la catégorie.
- *Réduction globale :*Dans ce cas, le nouvel ensemble de termes T' est choisi en fonction de toutes les catégories. Ainsi, chaque document D_j sera représenté par un seul vecteur V_j quelque soit la catégorie.

b) **Sélection d'attributs** La sélection d'attributs ou (feature selection) prend les termes d'origine et conserve seulement ceux jugés utiles à la classification, selon une certaine fonction d'évaluation. Les autres sont rejetés.

- ***La Fréquence-document (Document Frequency) :***
  Une première méthode de sélection qui peut être considérée comme une méthode de prétraitement approfondi : elle est très simple puisqu'elle correspond simplement au pourcentage de documents dans lesquels le terme apparaît, cette méthode conduit à supprimer les termes très fréquents et très rares afin de conserver les mots les plus importants avec le risque de supprimer des termes très riches et informatifs pour le système. Pour écarter les mots les plus fréquents, nous fixons un seuil maximal de fréquence n'autorisant pas de sélectionner les termes présents dans une très forte proportion de textes, de même un seuil minimal est fixé pour éliminer les termes très rares. Cette méthode ne prend pas en compte les catégories.

- ***Le Gain d'Information(Information Gain) :***
  C'est une mesure nécessaire pour prédire la catégorie d'un document selon la présence ou l'absence d'un mot dans un texte, on peut interpréter cette statistique par la quantité d'informations apportée par la présence ou l'absence d'un terme dans un document. Un gain d'information important indique que le terme contient plus d'informations pour le texte, en revanche, une perte d'informations indique que le terme contient moins d'informations nécessaire pour classer les textes avec ce terme. Le GI se traduit par la formule suivante selon [38] :

$$GI(t_k, c_i) = \sum_{c \in \{c_i, \overline{c_i}\}} \sum_{t \in \{t_k, \overline{t_k}\}} P(t, c) \cdot \log \frac{P(t, c)}{P(t) \cdot P(c)}$$

  t_k : Un terme appartenant à un document. c_i représente une catégorie. $P(t, c)$: Fréquence de document pour le terme t dans la catégorie c. $P(t)$: Probabilité qu'un document contient le terme t. $P(t, c)$: Probabilité qu'un document appartient à la catégorie c.

- ***L'Information Mutuelle(Mutual Information) :***
  Représente la corrélation entre deux variables aléatoires, pour notre cas, les deux variables sont le terme et la classe, l'information mutuelle s'appuie fortement sur le nombre de fois qu'un attribut apparait dans une certaine catégorie. Plus un attribut va apparaitre dans une catégorie, plus l'information mutuelle de cet attribut par rapport a cette catégorie va être élevée. Par contre, plus un attribut va apparaitre en dehors de la catégorie et, de suite, plus une catégorie va apparaitre sans l'attribut, moins l'information mutuelle va être élevée. On calcule l'information mutuelle d'un attribut t par rapport a une catégorie c_j comme suit [34] :

$$IM(t, c_j) = \log \frac{A \times N}{(A + C) \times (A + B)}$$

  Où,
  A est la fréquence de l'occurrence de t et c_j ensemble ;
  B est la fréquence de l'occurrence de t sans c_j ;
  C est la fréquence de l'occurrence de c_j sans t, et
  N est le nombre total de documents dans l'espace d'apprentissage. Le reproche que l'on peut faire à cette méthode est qu'elle se base fortement dans son calcul sur

la fréquence des attributs i.e pour une même probabilité conditionnelle sachant la catégorie, un attribut rare va être avantagé, car il risque moins d'apparaître en dehors de la catégorie.

- ***Le Rapport de Gain(Gain Ratio) :***
  La méthode de Gain d'Information a le biais naturel de favoriser les attributs ayant plusieurs valeurs. La méthode Rapport de Gain vise, par contre, à les pénaliser en incorporant un terme, que l'on appelle le partageur d'information (le dénominateur de l'équation ci-dessous), qui est sensible à l'uniformité du partitionnement des données selon chaque valeur v d'un certain attribut t. Par définition, le RG d'un attribut t est calculé selon [34] comme suit :

$$RG(N,t) = \frac{GI(N,t)}{-\sum_{i=1}^{v} \frac{|D_i|}{N} log_2 \frac{|D_i|}{N}}$$

  où, D_v est le nombre de documents contenants l'attribut t dont la valeur est v et N est le nombre total de documents dans l'espace d'apprentissage.

- ***Le Chi-deux ($\chi 2$) :***
  Mesure le manque d'indépendance entre un attribut t et une catégorie c_j selon [34], sa formule est désignée par :

$$\chi_2(t,c_j) = \frac{N \times (AD - CB)^2}{(A+C) \times (B+D) \times (A+B) \times (C+D))}$$

  Où, A est le nombre de documents appartenant à la catégorie c_j contenant l'attribut t ; B est le nombre de documents contenant l'attribut t mais n'appartenant pas à la catégorie c_j ; C est le nombre de documents appartenant à la catégorie c_j mais ne contenant pas l'attribut t ; D est le nombre de documents n'appartenant pas à la catégorie c_j et ne contenant pas l'attribut t ; et N est le nombre total de documents dans l'espace d'apprentissage. Pour calculer la valeur finale de $\chi 2(t)$, on choisit entre la somme, le moyen ou la valeur maximale individuelle. Elle s'adapte bien à la sélection d'attributs car elle évalue le manque d'indépendance entre un attribut et une catégorie. Elle utilise les mêmes notions de concurrence attribut/catégorie de la méthode l'Information Mutuelle. Elle est soumise à une normalisation qui rend les attributs appartenant à la même catégorie plus comparables entre eux. Il faut noter que le moindre changement des valeurs de A, B, C, ou D endommage sa fiabilité surtout pour les attributs peu fréquents.

- ***Le Rapport des chances(Odds Ratio) :***
  Mesure la possibilité qu'un attribut t existe dans une catégorie c_j. Par définition, on calcule le rapport des chances d'un attribut comme suit[34] :

$$RC(t,c_j) = \frac{AD}{BC}$$

  Où, A est le nombre de documents appartenant à la catégorie c_j et contenant l'attribut t ;
  B est le nombre de documents contenant l'attribut t mais n'appartenant pas à la catégorie c_j ;

C est le nombre de documents appartenant à la catégorie c_j mais ne contenant pas l'attribut t ;
D est le nombre de documents n'appartenant pas à la catégorie c_j et ne contenant pas l'attribut t.
Ensuite la somme des RC individuels est calculée pour obtenir le rapport des chances finales de l'attribut t.

- ***La Force du Terme (Term Strength) :***
  Il s'agit d'une méthode plutôt différente des autres. Elle se propose d'estimer l'importance d'un terme en fonction de sa propension à apparaître dans des documents semblables. Une première étape consiste à former des paires de documents dont la similarité cosinusoïdale est supérieure à un certain seuil. La force d'un terme est ensuite calculée à l'aide de la probabilité conditionnelle qu'il apparait dans le deuxième document d'une paire, sachant l'a été dans le premier.

c) **Extraction d'attributs** D'autres méthodes effectuent plutôt une extraction d'attributs (feature extraction). À partir des attributs de départ, elles créent de nouveaux attributs, en faisant soit des regroupements ou des transformations. Ainsi le texte sera représenté de telle manière que le descripteur ne sera plus un terme simple mais une combinaison de termes ou il va correspondre à un concept sémantique. Les méthodes de représentation des textes à base de concepts ou les combinaisons de termes, sont naturellement des techniques pour diminuer le nombre de termes, qui peuvent solutionner les problèmes de synonymie et polysémie. Les principales méthodes sont :

- ***Term Clustering*** **:**
  Cette méthode consiste à regrouper plusieurs termes pour former un nouvel attribut. Chaque attribut est donc censé représenter un concept sémantique. Cette association de plusieurs termes avec un concept permet de gérer la synonymie. La polysémie des mots est également prise en compte en permettant à un terme d'appartenir à plusieurs groupes.
- ***Latent Semantic Indexing (LSI)*** **:**
  Elle a été conçue au début par Deerwester et Landauer et utilisée dans le domaine de la recherche d' information pour résoudre les problèmes provenant des mots synonymiques, homonymiques, et polysémiques. L'analyse par sémantique latente est une méthode statistique permettant de découvrir les liens entre les mots. Elle fait l'hypothèse que la cooccurrence des termes cache une relation sémantique latente.
  L'analyse de la sémantique latente se fait en deux étapes. Dans un premier temps, la matrice d'occurrences est construite. Il s'agit d'une matrice dont les lignes représentent les termes, les colonnes sont les documents. L'élément (i, j) de la matrice correspond ainsi au nombre d'occurrences du terme j dans le texte i. L'étape suivante consiste à réduire les dimensions. Cette réduction est réalisée par le biais d'une décomposition aux valeurs singulières. La réduction à n dimensions va consister à ne conserver que les n premières de ces valeurs pour reconstituer une matrice approchée, de dimension n. Ainsi, si deux termes apparaissent fréquemment ensembles (nombre de cooccurrences important) alors ils ont plus de chance d'exprimer le même concept (synonymes). De même, soit trois termes t_1, t_2, t_3 si les couples (t_1, t_2) et (t_1, t_3) ont un nombre de cooccurrences important mais que

le couple (t_2, t_3) a un nombre de cooccurrences faible alors le terme t_1 exprime plusieurs concepts selon le contexte (polysémie). Une faiblesse de cette technique est que si jamais un attribut est particulièrement discriminant indépendamment des autres, le fait de le remplacer par un autre peut le rendre inefficace[7].

1.2.4 Le calcul du poids des descripteurs

Une fois on a réduit la taille du vocabulaire, le résultat est un tableau croisé de taille réduite dont les colonnes sont les attributs constituants le corpus d'apprentissage et les lignes sont les textes à catégoriser. Au départ, le contenu du tableau est la fréquence de chaque attribut t_i dans le document d_j où il existe. Lors de calcul du poids de chaque attribut, cette fréquence est remplacée par un poids qui indique sa pertinence. En effet, il existe plusieurs méthodes pour calculer le poids d'un attribut et la plupart d'entre elles respectent ces deux observations empiriques [18] :

- Plus le terme t_k est fréquent dans un document d_j, plus il est en rapport avec le sujet de ce document.
- Plus le terme t_k est fréquent dans l'ensemble des documents existants dans l'espace d'apprentissage, c'est qu'il a un pouvoir discriminant très faible.

Il existe plusieurs façons d'associer un poids à un terme :

- ***Codage booléen :*** Commençant par le choix le plus simple, qui ne s'intéresse que sur la présence ou la non présence d'un terme dans le texte, il consiste à utiliser une pondération binaire : 1 si le terme est présent une ou plusieurs fois dans le document, 0 dans le cas contraire. Cette représentation binaire est historiquement la plus ancienne et la plus simple. Néanmoins, cette fonction est moins utilisée pour les méthodes statistiques, car ce codage supprime de l'information qui peut être utile : l'apparition du même mot plusieurs fois dans un texte peut constituer un élément de décision important.

- ***Codage TF :*** Le codage TF(Term term Frequency) consiste à utiliser seulement le nombre d'occurrences des termes. Cette pondération ne peut être valable que dans les documents de même taille, sinon elle avantage les termes qui se répètent souvent dans les documents les plus longs.

- ***Codage*** $TF \times IDF$ ***:*** Le codage TF $\times IDF$ a été introduit dans le cadre du modèle vectoriel et utilise une fonction de l'occurrence multipliée par une fonction de l'inverse du nombre de documents différents dans lequel un terme apparaît. Ce sigle provient de l'anglais et signifie « Term FrequencyInverse Document Frequency ». Les termes caractérisant une classe apparaissent plusieurs fois dans les documents de cette classe, et moins, ou pas du tout, dans les autres. Le codage TF IDF est défini comme suit :

$$TF \times IDF(t_k, d_j) = \sharp(t_k, d_j) * \log \frac{|T_r|}{\sharp(T_r)(t_k)}$$

$\sharp(t_k, d_j)$ le nombre d'occurrences du terme t_k dans le texte d_j, $|T_r|$ le nombre de documents du corpus d'apprentissage et $\sharp(T_r)(t_k)$ le nombre de documents de cet ensemble dans lesquels apparaît au moins une fois le terme t_k.
Cette pondération issue du domaine de la Recherche d'Informations (RI) tire son

inspiration de la loi de Zipf introduisant le fait que les termes les plus informatifs d'un corpus ne sont pas ceux apparaissant le plus dans ce corpus. Ces mots sont la plupart du temps des mots outils. Par ailleurs, les mots les moins fréquents du corpus ne sont également pas les plus porteurs d'informations[38].

- ***Codage TFC :*** Le codage TF $\times IDF$ ne corrige pas la longueur des documents. Pour ce faire, le codage TFC est similaire à celui de TF $\times IDF$ mais il corrige la longueur des textes par la normalisation en cosinus, afin de ne pas favoriser les documents les plus longs.

$$TFC(t_k, d_j) = \frac{TF \times IDF(t_k, d_j)}{\sqrt{\sum_{s=1}^{|r|}(TF \times IDF(t_s d_j))^2}}$$

- ***L'entropie :*** Pour mesurer l'importance d'un terme dans un document, on peut également regarder la quantité d'informations qu'il apporte. Cette information peut être obtenu en se basant sur le modèle de la théorie de l'information, qui nous permet de calculer la valeur informationnelle de terme dans un document. Le facteur d'entropie qui a été élaboré par Shannon en 1948 pour modéliser la transmission des signaux électriques est utilisé dans le domaine de la RI pour mesurer la transmission de quantité d'informations par les termes de la collection. L'information produite par la densité d'un terme dans un document donné, peut être agrégée pour mesurer l'entropie d'un terme dans tous les documents de la collection. Une dernière approche de pondération significative s'appuie sur l'utilisation de l'entropie. Cette dernière mesure la dispersion d'un descripteur dans un corpus et peut s'avérer une information importante dans le cadre de la sélection de descripteur et/ou de pondération de la représentation fréquentielle d'un corpus. L'entropie E pour un descripteur i est décrite par la formule ci-dessous :

$$E(i) = \sum_j \frac{p_{ij} \log_2(p_{ij})}{log_2 N}$$

$$p_{ij} = \frac{tf_{ij}}{gf_i}$$

Où gf_j représente le nombre total de fois où le descripteur i apparaît dans le corpus de N documents. Une représentation avec l'approche fréquentielle peut alors être la suivante avec un terme i et un document j :

$$w_{ij} = (1 + E(i)) \log(tf_{ij} + 1)$$

Dans cette section on a vu que les textes en langage naturel ne peuvent pas être directement interprétés par un classifieur ou par les algorithmes de classification. Le sens d'un document peut être porté par un ensemble d'unités linguistiques particulières ou descripteurs. Ainsi avant la représentation des textes, un ensemble d'opérations préliminaires doivent être faites pour épurer le texte de tous les mots inutiles et conserver seulement ceux qui sont porteurs d'informations et utiles pour le processus de classification.
Dans le premier cas un descripteur est le mot tel qu'il apparait dans le document. Chaque mot est extrait du texte en considérant des séparateurs comme l'espace, la tabulation, la

ponctuation. Ces mots peuvent être remplacés par leurs racines, lemmes, N-grammes ou un concept.
Le nombre de mots caractérisant un corpus de textes peut être très grand, il est donc nécessaire de conserver un sous ensemble de ces mots pour éviter les problèmes provenant de la grande dimension de l'espace d'apprentissage. Ce filtrage repose à la base de différentes techniques de la réduction du vocabulaire. Plusieurs techniques de sélection des descripteurs ou réduction de dimensionnalité sont proposées dans la littérature, une bonne partie de ces approches sont étalées dans ce chapitre.
Une fois choisies la liste des descripteurs, un degré d'importance ou poids est attribué à tous les termes présents dans la représentation vectorielle puisque chaque terme possède un certain nombre d'occurrences dans le document ou dans le corpus qui est différent des autres.

1.3 Les techniques de classification

1.3.1 La méthode Rocchio

La méthode Rocchio proposée par (Rocchio,1971), est un classifieur linéaire basée sur le calcul des mesures de distance, il fait partie des premières techniques de classification supervisée. Dans la phase d'apprentissage de la méthode, les représentations vectorielles, vont permettre le codage de chaque catégorie par un vecteur dont lequel figure tous les termes générés avec leur nombre d'occurrence. Le processus représente donc les classes par des profils prototypiques correspondants à des vecteurs dans un espace vectoriel similaire aux documents. Ces profils sont donc des listes de termes pondérés générées pendant l'apprentissage de même pour les vecteurs correspondants aux textes qui sont aussi générés durant cette phase. Le profil d'une catégorie doit contenir tous les termes qui caractérisent cette catégorie par rapport aux autres. Selon la méthode de Rocchio[18], pour chaque catégorie c_i, les coordonnées t_{ki} du profil prototypique $c_i = (t_{1i}, t_{2i}, ..., t_{|\tau|i})$ sont calculés ainsi :

$$t_{ki} = \beta \sum_{d_j \in Pos_i} \frac{t_{kj}}{|Pos_i|} - \gamma \sum_{d_j \in Neg_i} \frac{t_{kj}}{|Neg_i|}$$

Où t_{ki} est le poids du terme t_k dans la document d_j ; $|Pos_i|$ est la cardinalité de l'ensemble des exemples positifs de la catégorie c_i ; et $|Neg_i|$ est la cardinalité de l'ensemble des exemples négatifs. β et γ (avec $\beta + \gamma = 1$; $\beta \geq 0$ et $\gamma \leq 0$) sont deux paramètres choisis selon l'importance que l'on accorde aux deux ensembles Pos_i et Neg_i. Dans la phase de classification, il s'agit de comparer le profil (vecteur) du nouveau document à classer à tous les profils des classes déjà calculés dans l'étape d'apprentissage. Cette comparaison équivaut au calcul d'une fonction de similarité ou de distance entre les vecteurs représentant les classes et le vecteur correspondant au nouveau document. Elle permet d'ordonner les classes en fonction de leur distance du document. Par conséquent, le principe de catégorisation Rocchio se résume à assigner le document à la classe dont la distance euclidienne entre le vecteur du document et le vecteur de la classe est la plus courte. Rocchio représente une caractéristique intéressante : il est robuste au bruit : même avec 50% des exemples bruités, ses performances sont presque inchangées[18]. L'inconvénient principal de la méthode Rocchio comme tous les classifieurs linéaires est que l'espace est divisé seulement en deux portions, ce qui peut être restrictif, car tous les problèmes ne sont pas nécessairement linéairement séparables[38].

1.3.2 k- Plus Proches Voisins

Le modèle des k plus proches voisins, en anglais(nearest neighbor (kNN)) est l'une des méthodes de classification les plus naturelles, cet algorithme fait parti des approches discriminatives car il évalue directement la classe d'un document à partir de ses caractéristiques. Cet algorithme est uniquement basé sur la mémorisation des exemples rencontrés pendant l'apprentissage, contrairement aux approches probabilistes, il ne nécessite aucune connaissance préalable, et a très peu de paramètres : le nombre K d'exemples jouant un rôle dans la classification d'un nouveau document, et la fonction de similarité pour comparer deux documents. Pendant la phase de classification, un nouveau document est confronté à l'ensemble des exemples mémorisés, selon une fonction de distance. Le classifieur calcule la similarité du nouveau texte à catégoriser avec l'ensemble des autres exemples du corpus d'apprentissage, dont les catégories sont déjà connues, puis il sélectionne les k documents les plus proches du document à classer. Ensuite, pour affecter la catégorie, les relations entre ces k documents et les catégories sont évaluées et un score est calculé par catégorie afin d'évaluer la pertinence de la catégorie au document. La catégorie (ou les catégories) ayant le score le plus élevé (celle qui contient le plus de textes voisins) est affectée au document. L'algorithme1 montre comment classer un nouveau document par la méthode k- PPV selon[18] :

Algorithm 1 Algorithme de classification par k-PPV

Paramètre : le nombre k de voisins

Contexte : un échantillon de textes classés en $C = c_1, c_2, ..., c_n$ classes

1 :**Pour** chaque texte t **Faire** ;
2 : transformer le texte t en vecteur $t = (x_1, x_2, ..., x_m)$;
3 :déterminer les k plus proches textes du texte t selon une métrique de distance ;
4 : combiner les classes de ces k exemples en une classe c ;
5 :**Fin Pour**

Sortie : le texte t associé à la classe c.

La mesure de dissimilarité généralement utilisée est la distance Euclidienne. Cependant, selon la représentation des caractéristiques, remarquons que des fonctions de similarité plus appropriées peuvent être utilisées.
Le problème de cette approche selon[18] :

- est qu'elle consomme beaucoup de ressources elle nécessite beaucoup de mémoire pour stocker les exemples et est très couteuse en temps de classification étant donné qu'il faut pour chaque document calculer sa distance à tous les exemples.
- elle est sensible au bruit, la précision de l'algorithme se dégrade en présence de données bruitées. Il devient alors difficile de généraliser.
- elle est sensible au choix de la fonction de similarité de l'algorithme.

1.3.3 Arbres de décision

Les arbres de décision sont des règles de classification qui basent leur décision sur une suite de tests associés aux attributs, les tests étant organisés de manière arborescente. Les premiers algorithmes de classification par arbres de décision sont anciens. Les deux travaux les plus marquants sont la création de CART, par Breiman en 1984 et la création de C4.5 par Quinlan en 1993.

Un classifieur de texte basé sur la méthode d'arbre de décision est un arbre de nœuds internes qui sont marqués par des termes, les branches qui sortent des nœuds sont des tests sur les termes, et les feuilles sont marquées par catégories. Ce classifieur classe un document du test d_j en testant récursivement les poids des nœuds internes de vecteur $\overrightarrow{d_j}$, jusqu'à ce qu'une feuille soit atteinte. L'étiquette de ce nœud est alors attribuée à d_j.
Une méthode pour effectuer l'apprentissage d'un arbre de décision pour la catégorie c_i consiste à vérifier si tous les exemples d'apprentissage ont la même étiquette (c_i ou $\overline{c_i}$), dans le cas contraire nous sélectionnons un terme t_k, et nous partitionnons l'ensemble d'apprentissage en classes de documents qui ont la même valeur pour t_k, et à la fin l'on crée les sous-arbres pour chacune de ces classes. Ce processus est répété récursivement sur les sous-arbres jusqu'à ce que chaque feuille de l'arbre généré de cette façon contienne des exemples d'apprentissage attribués à la même catégorie c_i, qui est alors choisie comme l'étiquette de la feuille. L'étape la plus importante est le choix du terme de t_k pour effectuer la partition. Toutefois, une telle méthode de construction d'arbre peut faire l'objet de sur-apprentissage, comme certaines branches peuvent être trop spécifiques pour les données d'apprentissage. La plupart des méthodes d'apprentissage des arbres incluent une méthode pour la construction d'arbre et pour élaguer les branches trop spécifiques, selon l'algorithme 2 donné par [18] :

Algorithm 2 Algorithme général d'apprentissage par arbres de décision

Contexte : un échantillon Ω de S textes classés (d_j, c_i)
Vérifie : arbre vide ; nœud courant : racine ; échantillon courant Ω :
1 : **répéter**
2 : **Si** le nœud courant est terminal **Alors**
3 : étiqueter le nœud courant par une feuille portant le nom de cette classe ;
4 : **Sinon**
5 : Choisir le meilleur attribut (terme) pour créer le sous-arbre(Sélectionner un test et créer autant de nouveaux nœuds qu'il y a de réponses possibles au tests ;
6 : **Fin Si**
7 : Passer au nœud suivant non-exploré (s'il existe) ;
8 : **jusqu'à** production d'un arbre de décision(Plus de nœud sans classe) ;
9 :élaguer l'arbre de décision obtenu ;
Sortie : arbre de décision élagué ;

- *Nœud terminal ?*, on décide qu'un nœud est terminal lorsque tous les exemples associés à ce nœud, ou du moins la plupart d'entre eux sont dans la même classe, ou encore, s'il n'y a plus d'attributs non utilisés dans la branche correspondante.
- *Quelle classe à un nœud terminal ?*,on attribue au nœud la classe majoritaire (éventuellement calculée à l'aide d'une fonction de cout lorsque les erreurs de prédiction ne sont pas équivalentes). Lorsque plusieurs classes sont en concurrence, on peut choisir la classe la plus représentée dans l'ensemble de l'échantillon, ou en choisir une au hasard.
- *Sélection d'un test ?*La sélection d'un test à associer à un nœud est plus délicate. Puisqu'on cherche à construire un arbre de décision le plus petit possible rendant compte au mieux des données, une idée naturelle consiste à chercher un test qui fait le plus progresser dans la tache de classification des données d'apprentissage. Comment mesurer cette progression ? CART utilise l'indice de Gini et C4.5 utilise la notion d'entropie et gain d' information.

– *Élagage ?* Faible pouvoir prédictif (notion de sur-apprentissage – ou apprentissage par cœur) nécessité d'obtenir un arbre plus petit .

Les arbres de décision présentent les caractéristiques suivantes selon [18]

– lisibilité du résultat : un arbre de décision est facile à interpréter car il est la représentation graphique d'un ensemble de règles.
– tout type de données : l'algorithme peut prendre en compte tous les types d'attributs et les valeurs manquantes. Il est robuste au bruit.
– sélection des variables : l'algorithme intègre une procédure de sélection de variables, ainsi les variables contenues dans l'arbre sont utiles pour la classification.
– classification efficace : l'attribution d'une classe à un exemple à l'aide d'un arbre de décision est un processus très efficace et rapide (parcours d'un chemin dans un arbre).
– sensible au nombre de classes : les performances tendent à se dégrader lorsque le nombre de classes devient trop important.
– èvolutivité dans le temps : l'algorithme n'est pas incrémental, c'est-à-dire, que si les données évoluent avec le temps, il est nécessaire de relancer une phase d'apprentissage sur l'échantillon complet (anciens exemples et nouveaux exemples).

1.3.4 Naïve Bayes

L'algorithme Naïve Bayes (NB), est une autre méthode bien utilisée en apprentissage, elle est également employée dans la classification automatique de textes. La classification bayésienne naïve de textes est une approche probabiliste de classification simple. Cette approche est basée sur un modèle probabiliste dérivant du théorème de Bayes qui fait l'hypothèse que les mots qui apparaissent dans un document sont indépendants les un des autres. Ce qui n'est pas tout à fait le cas dans la pratique(d'où vient la partie naïve de ce modèle). [33].

Supposons que nous disposons de n catégories de documents, déterminer à quelle catégorie c_i sera associée un document D revient à calculer la probabilité d'appartenance du document D à la catégorie c_i. En se basant sur le théorème bayes, on peut calculer cette probabilité de la façon suivante :

$$P(c_i|D) = \frac{P(D|c_i) * P(c_i)}{P(D)}$$

Dans cette formule, $P(c_i|D)$ représente la probabilité d'appartenance du document D à la catégorie c_i qui peut être également déterminée, en évaluant la fréquence d'apparition des mots du document D qui sont associés à la catégorie c_i ; $P(D|c_i)$ est la probabilité selon laquelle, pour une catégorie donnée, les mots du document D sont associés à la catégorie c_i ; $P(c_i)$ est la probabilité qui associe le document D à la catégorie c_i indépendamment du contenu du document ; $P(D)$ est la probabilité propre du document D.

Pour réellement déterminer à quelle catégorie un document appartient, il faut calculer $P(c_i|D)$ pour chacune des catégories. Étant donné que $P(D)$ reste constant pour toutes les catégories,déterminer $P(c_i|D)$ se résume juste au calcul de $P(D|c_i) * P(c_i)$.

En considérant que le document D est composé d'un ensemble de mots que nous noterons $w_1, ..., w_m$, calculer$P(D|c_i)$ reviendrait à calculer le produit des probabilités d'apparition de chaque mot w_i dans la catégorie c_i. Ce calcul se justifie par l'hypothèse selon laquelle tous les mots apparaissent indépendamment les un des autres dans un document. Ce qui

permet finalement d'écrire

$$P(D|c_i) = P(w_1|c_i) * P(w_2|c_i)*, ..., *P(w_m|c_i)$$

Pour chacune des catégories, $P(w_m|c_i)$ est le rapport entre le nombre de fois que le mot w_i apparaît dans la catégorie c_i et le nombre total de mots que comprend la catégorie c_i. $P(c_i)$ est calculé en divisant le nombre total de mots pour la catégorie c_i par la somme du nombre total de mots dans toute les catégories. D'où la formulation suivante :

$$P(c_i|D) = P(w_1|c_i) * P(w_2|c_i)*, ..., *P(w_m|c_i) * P(c_i)$$

Ce calcul est effectué pour chaque catégorie et on considère la probabilité la plus élevée pour choisir à quelle catégorie sera associée le document qu'on souhaite classer.
Le calcul ainsi présenté se justifie par l'hypothèse selon laquelle tous les mots apparaissent indépendamment les uns des autres dans le document. D'où le caractère naïf de la classification : En réalité, la probabilité d'apparition d'un mot est lié aux mots précédents. Mais malgré cette hypothèse simpliste, ce classifieur peut donner de bons résultats[33].
L'algorithme NB est connu par son efficacité et sa simplicité qui revient à l'effet admis, d'indépendance entre les différents descripteurs et à cause de cette hypothèse d'indépendance des mots dans ce modèle, on le qualifie souvent de "Naïve", "Idiot", "Simple". Ce classifieur est très favorable pour les documents courts qui donne des résultats très intéressants, néanmoins ces performances sont réduites quand il s'agit d'un vocabulaire important à traiter, ainsi le manque d'une meilleure prise en compte de la taille des documents, fait que ses performances en qualité de classement se dégradent avec l'augmentation du nombre de caractéristiques. En effet, si le nombre de termes augmente, alors le nombre des dépendances entre l'ensemble des termes augmente aussi, et donc, la vérification de l'hypothèse de Naïve Bayes diminue. Le fonctionnement de naïve bayes est relativement similaire à celui de Rocchio. Chaque classe est décrite par un profil qui gère un coefficient par terme (P_{jk} pour Rocchio, $P(t_k|c_j)$ pour Naïve Bayes). Tous ces coefficients sont ensuite regroupés pour former une valeur de pertinence (un degré de similarité pour Rocchio, une probabilité pour Naïve Bayes).

1.3.5 Machines à Vecteurs de Support

Les machines à vecteurs de support (Support Vector Machines) ou Séparateurs à Vastes Marges (SVM) découlent directement des travaux de Vapnik en théorie de l'apprentissage statistique. C'est une méthode de classification supervisée binaire qui a été introduite en 1992. Par la suite, elle a été étendue à des problèmes de régression, d'estimation de densité et de classification non supervisée. Le succès de cette méthode est justifié par les solides bases théoriques qui la soutiennent. Les SVM reposent sur deux idées fortes : le principe de la maximisation de la marge et le principe de la fonction de noyaux

Principe de fonctionnement général

a) **Les concepts de bases des SVM :**
   - **Hyperplan :** Plaçons-nous dans le cas d'une classification binaire (i.e. les exemples à classifier réparties en 2 classes). On appelle hyperplan séparateur un hyperplan qui sépare les deux classes (figure 1.3), en particulier il sépare leurs points

d'apprentissage. Comme il n'est en générale pas possible d'en trouver un, on se contentera donc de chercher un hyperplan discriminant qui est une approximation au sens d'un critère a fixer (maximiser la distance entre ces deux classes) [16].

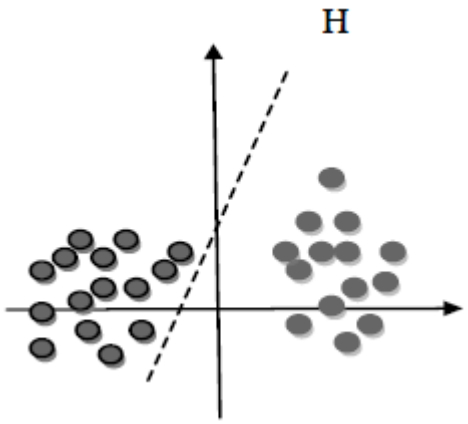

FIGURE 1.3: L'hyperplan H qui sépare les deux classes.

– **Vecteur de support :**
Pour une tache de détermination de l'hyperplan séparable des SVM est d'utiliser seulement les points les plus proches (i.e. les points de la frontière entre les deux classes des données) parmi l'ensemble total d'apprentissage, ces points sont appelés vecteurs de support(figure 1.4)[16] .

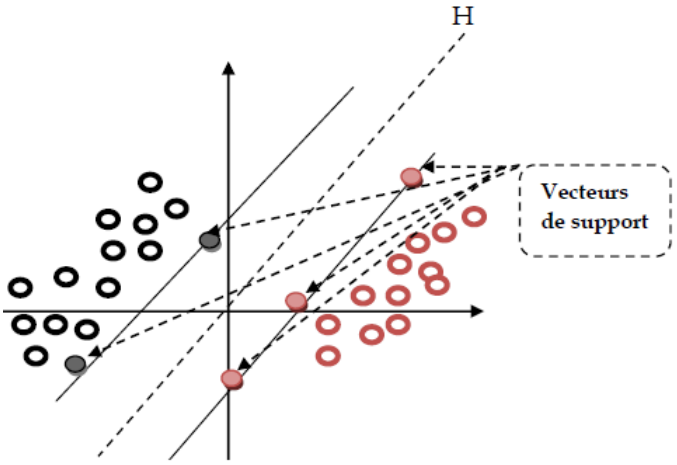

FIGURE 1.4: Les vecteurs de support

– **Marge :**
Il est évident qu'il existe une multitude d'hyperplans valides mais la propriété remarquable des SVM est que cet hyperplan doit être optimal. Formellement, cela revient à chercher un hyperplan dont la distance minimale aux exemples d'apprentissage est maximale. On appelle cette distance « marge » entre l'hyperplan et les exemples. L'hyperplan séparateur optimal est celui qui maximise la marge (figure 1.5)[16].

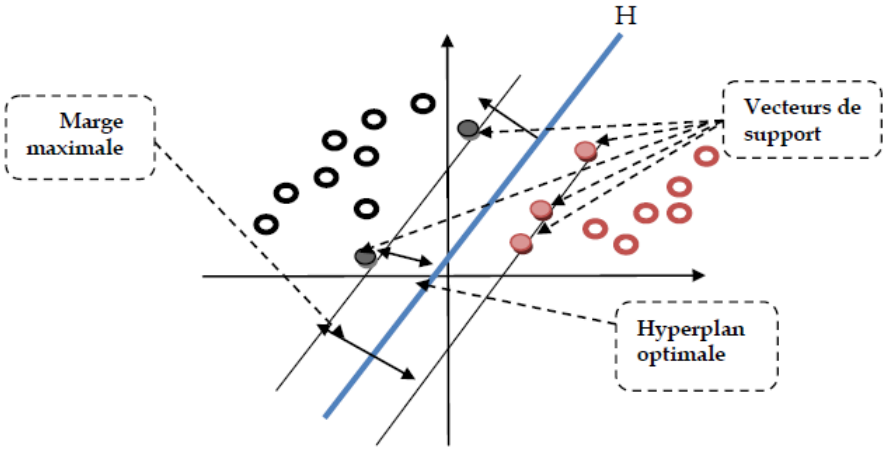

FIGURE 1.5: Hyperplan optimal, Vecteurs de support et Marge maximale

b) **Fonctionnement :** Les machines à vecteurs de support sont basés sur la recherche de l'hyperplan de marge optimale qui, lorsque c'est possible, classe ou sépare correctement les données tout en étant le plus éloigné possible de toutes les observations. En garantissant que la marge entre le plus proche des positifs et des négatifs soit maximale. Le principe est donc de trouver un classifieur, ou une fonction de discrimination, dont la capacité de généralisation (qualité de prévision) est la plus grande possible. SVM est considéré comme un des algorithmes les plus performants en classification textuelle [19].

SVM binaires

Le cas le plus simple est celui où les données d'apprentissage viennent uniquement de deux classes différentes (+1 ou -1), on parle alors de classification binaire. Parmi les modèles des SVM, on constate les cas linéairement séparables et les cas non linéairement séparables. Les premiers sont les plus simples des SVM car ils permettent de trouver facilement le classifieur linéaire. Dans la plupart des problèmes réels il n'y a pas de séparation linéaire possible entre les données, le classifieur de marge maximale ne peut pas être utilisé car il fonctionne seulement si les classes de données d'apprentissage sont linéairement séparables .

i) **Données linéairement séparables :** Soit S un ensemble d'exemples d'apprentissage de données linéairement séparables(figure 1.6), tel que $S = \{(x_i, y_i)/i = 1...n\}$ avec $y_i = \{+1, -1\}$ L'appartenance d'un vecteur à une classe ou à l'autre est matérialisée ici par la valeur 1 ou -1 de l'étiquette y.
Un séparateur linéaire noté $f_{w,b}$ est fourni par l'équation :$f_{w,b} =< w, x > +b$ Pour obtenir la classe on utilisera seulement le signe de $f_{w,b}(x)$
On note S^+ l'ensemble des exemples d'apprentissage dont la classe vaut 1 (cas $y = 1$), et S^- l'ensemble des exemples d'apprentissage dont la classe vaut -1 (cas$y = -1$)
S est linéairement séparable s'il existe W et b tels que $\forall x \in S^+ f_{w,b}(x) > 0$ et $\forall x \in S^- f_{w,b}(x) < 0$
Nous sommes maintenant face à un problème d'optimisation : trouver W et b maximisant le marge.

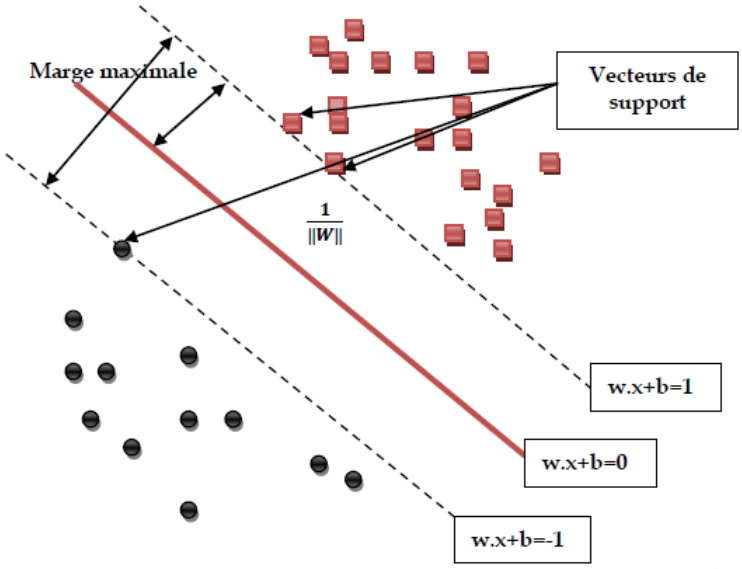

FIGURE 1.6: Exemple graphique des données linéairement séparables

a) **Forme Primale du problème d'optimisation SVM :** Si les données sont linéairement séparables, alors il existe un hyperplan d'équation $Wx + b = 0$ tel que :

$$\begin{cases} Wx_i + b \geq +1 & si \quad y_i = +1 \\ Wx_i + b \leq -1 & si \quad y_i = -1 \end{cases}$$

On peut combiner ces deux inéquations en une seule :

$$y_i(Wx_i + b) \geq +1$$

La distance perpendiculaire de l'origine à l'hyperplan :

$$H_1 : Wx_i + b = 1 \qquad est \qquad \frac{|1-b|}{\| W \|}$$

De même pour :

$$H_2 : Wx_i + b = -1 \qquad est \qquad \frac{|1-b|}{\| W \|}$$

De façon similaire, la distance entre un point situé sur H_1 et l'hyperplan H_2 est donnée par :

$$\frac{|Wx_i + b|}{\|W\|} = \frac{1}{\|W\|}$$

Donc la marge (la distance entre les deux hyperplans H_1 et H_2) est $\frac{2}{\|W\|}$. La maximisation de cette quantité revient à minimiser l'inverse $\frac{2}{\|W\|}$. La forme primale (qui dépend seulement de W et b) des SVM est donc un problème de minimisation sous contrainte qui s'écrit[28] :

$$\begin{cases} \min \frac{1}{2}\|W\|^2 \\ \text{ sous la contrainte} \\ y_i(Wx_i + b \geq 1), \forall (x_i, y_i) \in S \end{cases} \tag{1.1}$$

b) **Forme duale du problème d'optimisation SVM :** La formulation primale du problème 1.1 peut être transformée en formulation duale en utilisant les multiplicateurs de Lagrange $\alpha_i T$ à chaque contrainte ($\alpha_i \geq 0$) [28]Le lagrangien est donné par :

$$[L(W, b, \alpha) = \frac{1}{2}\|W\|^2 - \sum_{i=1}^{N} \alpha_i \, y_i(Wx_i + b) + \sum_{i=1}^{N} \alpha_i \tag{1.2}$$

En passant à la formulation duale, le problème devient : maximiser le Lagrangien, cela revient à dire, de trouver les α_i et W qui annulent ses dérivées partielles :

$$[\frac{\partial L(W, b, \alpha)}{\partial W} = 0, \frac{\partial L(W, b, \alpha)}{\partial b} = 0 \quad et \quad \alpha_i \geq 0 \tag{1.3}$$

on trouve :

$$[W = \sum_{i=1}^{N} \alpha_i \, y_i \, x_i \quad et \quad \sum_{i=1}^{N} \alpha_i \, y_i = 0 \tag{1.4}$$

Et en réinjectant les deux premières dérivées partielles 1.4 dans l'équation 1.2 nous obtenons la formulation duale (dépendant des α_i) suivante :

$$[L(W, b, \alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \, \alpha_j \, y_i \, y_j \, X^T \, X \tag{1.5}$$

Ainsi notre problème est de maximiser L (w, b,) sous la contrainte : $\sum_{i=1}^{N} \alpha_i y_i = 0$; et $\alpha_i \geq 0$.
La résolution des α_i donne la valeur du vecteur $W = \sum_{i=1}^{N} \alpha_i \, x_i \, y_i$
et peut classer une nouvelle cible suivant son vecteur de caractéristique x selon la fonction :

$$f(x) = signe(Wx + b) = signe(\sum_{i=1}^{N} \alpha_i \, x_i \, y_i . x + b)$$

Données linéairement non séparables : Malgré une base théorique solide, les SVM restent toutefois fortement limitées par la restriction aux séparateurs linéaires. Il est en effet rare que des données réelles soient providentiellement réparties de chaque côté d'un hyperplan.la figure 1.7 donne une illustration des données non-linéairement séparables.

a) **Classification à marge souple :** En réalité, un hyperplan séparateur n'existe pas toujours, et même s'il existe, il ne représente pas généralement la meilleure solution pour la classification. En plus une erreur d'étiquetage dans les données d'apprentissage (un exemple étiqueté +1 au lieu de -1 par exemple) affectera crucialement l'hyperplan.
Dans le cas où les données ne sont pas linéairement séparables, ou contiennent du bruit (données mal étiquetées) et il y a nécessité de les relaxer un peu. Ceci peut être fait en admettant une certaine erreur de classification des données ce qui est appelé "SVM à marge souple (Soft Margin)" [22].

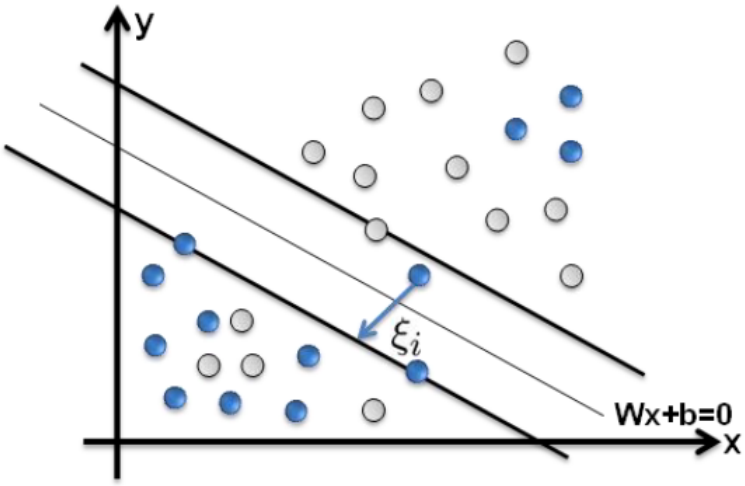

FIGURE 1.7: Hyperplans séparateurs dans le cas de données linéairement non séparables

L'hyperplan optimal est celui qui satisfait les conditions suivantes :

– La distance entre les vecteurs bien classés et l'hyperplan optimal doit être maximale.
– La distance entre les vecteurs mal classés et l'hyperplan optimal doit être minimale.
Pour formaliser tout cela, on introduit des variables de pénalité non-négatives appelées variables d'écart $\xi_i (i = 1, ..., N)$ avec $\xi > 0$,dans les contraintes, qui deviennent :

$$\begin{cases} Wx_i + b \geq 1 - \xi & Si \quad y_i = 1 \\ Wx_i + b \leq -1 + \xi & Si \quad y_i = -1 \end{cases}$$

On a ajoute des variables d'écart et un paramètre C qu'il faudra régler afin d'équilibrer d'un côté la maximisation de la marge et de l'autre le nombre d'observations que l'on accepte de mal classer. Un moyen naturel de donner un coût aux erreurs est de remplacer la fonction à minimiser précédente formule 1.1 par :

$$\begin{cases} \min \frac{1}{2}\|W\|^2 + C(\sum(\xi_i) \\ \text{ sous la contrainte} \\ y_i(Wx + b \geq 1 - \xi_i), \forall (x_i, y_i) \in S, \xi \geq 0 \end{cases}$$

Le problème dual a la même forme à l'exception d'une constante C

$$\begin{cases} \max \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \, \alpha_j \, y_i \, y_j \, x_i \, x_j \\ \text{ sous la contrainte} \\ \forall i \qquad 0 \leq \alpha_i \leq C \quad , \qquad \sum_{i=1}^{N} \alpha_i y_i = 0 \end{cases}$$

b) **Noyau :[28]** Le fait d'admettre la mal-classification de certains exemples, ne peut pas toujours donner une bonne généralisation pour un hyperplan même si ce dernier est optimisé. Pour surmonter les inconvénients des cas non linéairement séparables, l'idée des SVM est de changer l'espace des données. La transformation non linéaire des données peut permettre une séparation linéaire des exemples dans un nouvel espace. On va donc avoir un changement de dimension. Cet nouvel espace est appelé « espace de re-description ». En effet, intuitivement, plus la dimension de l'espace de re-description est grande, plus la probabilité de pouvoir trouver un hyperplan séparateur entre les exemples est élevée. Ceci est illustré par le schéma de la figure 1.8 :

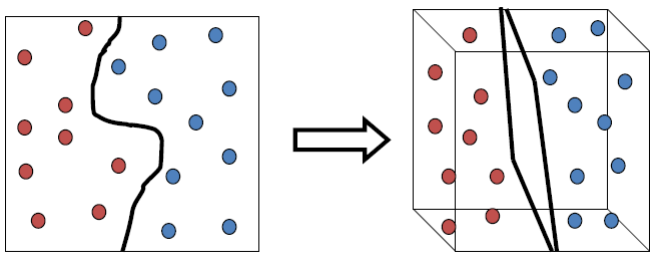

FIGURE 1.8: Données linéairement non séparables

On a donc une transformation d'un problème de séparation non linéaire dans l'espace de représentation en un problème de séparation linéaire dans un espace de re-description de plus grande dimension. Cette transformation non linéaire est réalisée via une fonction noyau. En pratique, quelques familles de fonctions noyau paramétrables sont connues et il revient à l'utilisateur de SVM d'effectuer des tests pour déterminer celle qui convient le mieux pour son application. On peut citer les exemples de noyaux suivants : polynomiale, gaussien, sigmoïde et laplacien.

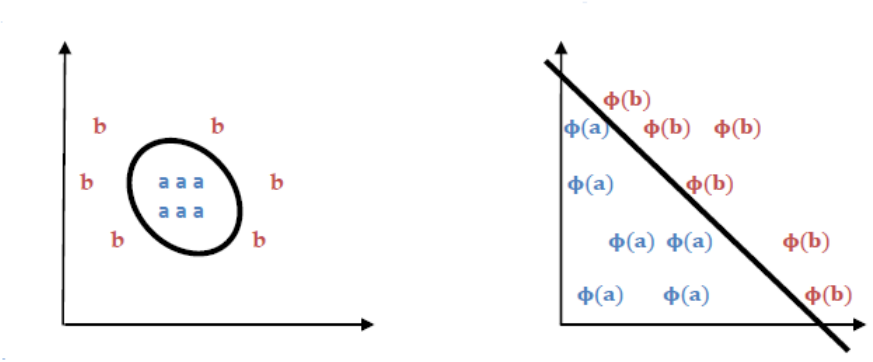

FIGURE 1.9: Espace de projection des données non linéairement séparables.

Dans le cas où les données sont non linéairement séparables, c'est-à-dire la surface séparatrice est non linéaire, on transpose le problème dans un autre espace F de dimension plus élevé pour rendre les nuages de points des deux classes linéairement séparable au moyen d'une transformation figure 1.9 tel que :

$$\Phi \ : x \rightarrow \Phi \in F$$

Le calcul de la surface de séparation revient alors à chercher l'hyperplan optimal dans ce nouvel espace F. La fonction de décision peut être représentée par le produit scalaire :

$$\Phi^T(x_i) * \Phi(x_j)$$

Cette dernière quantité peut être remplacée par une fonction de la forme $K(x_i, y_i)$ (Les fonctions scalaires symétriques et définies positives, que l'on désigne souvent simplement par noyaux, sont plus précisément des noyaux de Mercer), c'est ce qu'on appelle le noyau. Donc :

$$K(x_i, y_i) = \Phi^T(x_i) * \Phi(x_j)$$

Le lagrangien devient alors :

$$L(W, b, \alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \, \alpha_j \, y_i \, y_j \, x_i \, x_j K(x_i, y_i)$$

A ce stade, le problème se situe dans le choix de la transformation Φ ou plus généralement à la fonction noyau K. Il existe peu de noyaux régulièrement utilisés avec

les SVM.
Quelques noyaux utilisables :
- Linéaire : $K(x, y) = \langle x, y \rangle$.
- Polynômial : $K(x, y) = (c + \langle x, y \rangle)^d$.
- Gaussien (Radial Basis Function, RBF) : $K(x, y) = exp(-\frac{\|x-y\|^2}{2\sigma^2})$
- Laplacien : $K(x, y) = e^{\frac{-\|x-y\|}{\sigma^2}}$

SVM multi-classes

Le principe du SVM expliqué dans la partie précédente se résume dans la résolution des problèmes de classification binaire, or les problèmes rencontrés dans la réalité, sont de type multi-classes. D'où l'importance d'étendre le principe du SVM aux problèmes de plus de deux classes. Les méthodes des machines à vecteur support multiclasse, réduisent le problème multi-classe à une composition de plusieurs hyperplans bi-classes permettant de tracer les frontières de décision entre les différentes classes. Ces méthodes décomposent l'ensemble d'exemples en plusieurs sous ensembles représentant chacun un problème de classification binaire. Pour chaque problème un hyperplan de séparation est déterminé par la méthode SVM binaire. On trouve dans la littérature plusieurs méthodes de décomposition :

a) **Un contre un (One-Versus-One) :** Dans la méthode OVO, pour un problème de classification comportant k classes, on construit pour chaque combinaison possible de deux classes distinctes un classifieur SVM binaire, elle consiste à utiliser un classifieur par couple de catégories. Donc on aura un total de $k(k-1)/2$ classifieurs binaires et lors du test une classe peut être prédite au maximum $k-1$ fois. Plusieurs alternatives de vote sont envisageables et généralement une classe parmi les classes qui ont un nombre élevé de prédictions est choisie. Dans la stratégie intitulée "le max qui gagne", l'exemple x à prédire est testé par tous les classifieurs binaires possibles entre tout couple de classes C_i et C_j. Pour chaque classifieur, si l'exemple x est assigné à la classe C_i donc on incrémente le compteur de la classe C_i sinon on incrémente le compteur de la classe C_j. A la fin, la classe gagnante sera la classe ayant la valeur du compteur maximale. En cas de conflit, c'est-à-dire deux classes ou plus ont des valeurs maximales identiques, on choisira la classe du petit indice [14].

b) **Un contre tous (One-Versus-All) :** Dans un problème de classification comportant k classes, on conçoit k classifieurs binaires. Pour chaque classifieur i, les exemples de la classe i sont étiquetés comme positifs et les exemples de toutes les autres classes restantes sont marqués comme négatifs, donc on a opposé la classe i aux autres classes. Si l'exemple x à tester est assigné à plusieurs classes (cas de conflit), on peut par exemple choisir la classe la plus proche de x [14]. La marge de$x$ dans chaque classifieur i est donnée par l'expression :

$$marge \qquad e_i = (\sum_{j=1}^{s} \alpha_i \, y_i \, k(x_j, x) + b)$$

Parmi les classifications où la marge est positive on prend la maximale :

$$classe\ finale = argmax(marge\ e_i)$$

1.4 Évaluation d'un classifieur

L'évaluation consiste à mesurer la différence entre un résultat attendu et un résultat obtenu. Aucune métrique n'y est associée mais on utilise en général un indicateur compris entre 0 et 1 pour en faciliter l'interprétation selon [32].
L'évaluation du classifieur des documents est effectuée de façon expérimentale, plutôt que de façon analytique. Le traitement est expérimental car pour évaluer un système de façon analytique, donc pour prouver que le système est correct et complet, nous aurions besoin d'une spécification formelle du problème que le système tente de résoudre. La notion de systèmes de catégorisation de texte est, en raison de son caractère subjectif, par nature non-formalizable comme il s'agissait de déterminer l'appartenance d'un document à une catégorie [38].
Plusieurs mesures d'évaluation ont été proposées dans la littérature. Nous allons nous contenter de présenter, celles souvent utilisées par les chercheurs du domaine de la classification automatique des documents. Pour mesurer les performances des classifieurs, plusieurs mesures sont proposées, dans la présente section nous allons présenter les mesures de performance souvent utilisées dans la littérature, utilisées par [38],[18],[36],[34].

a) ***Matrice de contingence :***
Pour évaluer les résultats obtenus par un classifieur, les documents de l'espace d'apprentissage sont souvent divisés en deux ensembles : le premier est utilisé pour la construction du classifieur tandis que le deuxième est utilisé pour faire le test. Puisqu'on adopte l'approche de classification supervisée on connait à l'avance la catégorie de chaque document. Ainsi, on compare la catégorie prédite avec celle prédéfinie et on calcul un score de performance. Nous pouvons construire la matrice de contingence pour chaque classe (Voir tableau 1.1), qui fournit 4 informations essentielles :

- Vrai Positif (VP) : Le nombre de documents attribués à une catégorie convenablement. (Documents attribués à leurs vraies catégories).
- Faux Positif (FP) : Le nombre de documents attribués à une catégorie inconvenablement. (Documents attribués à des mauvaises catégories).
- Faux Négatif (FN) : Le nombre de documents inconvenablement non attribués. (Qui auraient dû être attribués à une catégorie mais qui ne l'ont pas été).
- Vrai Négatif(VN) : Le nombre de documents non attribués à une catégorie convenablement (Qui n'ont pas à être attribués à une catégorie, et ne l'ont pas été).

Catégorie C_i		Jugement Expert	
		Oui	Non
Jugement Classifieur	oui	VP_i	FP_i
	Non	FN_i	VN_i

TABLE 1.1: Tableau de contingence de catégorie C_i

A partir de ce tableau de contingence, la communauté du TALN (Traitement Automatique de Langues Naturelles) calcule divers indicateurs de base, eux-mêmes combinés pour donner d'autres mesures.

b) ***Précision et Rappel :***
Les performances en termes de classification sont généralement mesurées à partir de deux indicateurs traditionnellement utilisés c'est les mesures de rappel et précision.

Initialement elles ont été conçues pour les systèmes de recherche d'information, mais par la suite la communauté de classification de textes les a adoptées. Formellement, pour chaque classe C_i, on calcule deux probabilités qui peuvent être estimées à partir de la matrice de contingence correspondante, ainsi ces deux mesures peuvent être définies de la manière suivante :

- **Le rappel :**
  étant la proportion de documents correctement classés par le système par rapport à tous les documents de la classe C_i.

$$\text{Rappel}(C_i) = \frac{\text{Nombre de documents bien classés dans } C_i}{\text{Nombre de documents de la classe } C_i}$$

$$\text{Rappel}(C_i) = \frac{VP_i}{VP_i + FN_i}$$

  Le rappel mesure la capacité d'un système de classification à détecter les documents correctement classés. Cependant, un système de classification qui considérerait tous les documents comme pertinents obtiendrait un rappel de 100%. Un rappel fort ou faible n'est pas suffisant pour évaluer les performances d'un système. Pour cela, on définit la précision.

- **La Précision :**
  est la proportion de documents correctement classés parmi ceux classés par le système dans C_i.

$$\text{Précision}(C_i) = \frac{\text{Nombre de documents bien classés dans } C_i}{\text{Nombre de documents classés dans } C_i}$$

$$\text{Précision}(C_i) = \frac{VP_i}{VP_i + FP_i}$$

  La précision mesure la capacité d'un système de classification à ne pas classer un document dans une classe, un document qui ne l'est pas. Comme elle peut aussi être interprétée par la probabilité conditionnelle qu'un document choisi aléatoirement dans la classe soit bien classé par le classifieur. Ces deux indicateurs pris l'un indépendamment de l'autre ne permettent d'évaluer qu'une facette du système de classification : la qualité ou la quantité. Les courbes de précision vs rappel permettent de mieux comprendre le comportement du classifieur, et de visualiser l'évolution de la précision en fonction du rappel.

c) ***Bruit et Silence :***
On peut également définir les notions de Bruit et de Silence qui sont respectivement les notions complémentaires de la précision et du rappel. On utilise aussi la notion de bruit qui présente le problème selon le point de vue opposé de la précision. Le bruit est le pourcentage de textes incorrectement associés à une classe par le système :

$$\text{Bruit}(C_i) = 1 - \text{Précision}(C_i) = \frac{FP_i}{VP_i + FP_i}$$

La notion de silence est le point de vue opposé du rappel. Le silence est le pourcentage de textes à associer à une classe incorrectement non classés par le système :

$$\text{Silence}(C_i) = 1 - \text{Rappel}(C_i) = \frac{FN_i}{VP_i + FN_i}$$

d) ***Taux de succès et taux d'erreur :***
Le taux de succès ou l'exactitude (Accuracy rate) et le taux d'erreur (Error rate) sont deux mesures souvent utilisées par la communauté de l'apprentissage automatique. Le taux de succès désigne le pourcentage d'exemples bien classés par le classifieur, tandis que le taux d'erreur désigne le pourcentage d'exemples mal classés. Les deux taux sont estimés comme suit :

$$\text{Accuracy rate} = 1 - \text{Error rate} = \frac{VP + VN}{VP + VN + FP + FN}$$

$$\text{Error rate} = 1 - \text{Accuracy rate} = \frac{FP + FN}{VP + VN + FP + FN}$$

e) ***Taux de chute et la spécificité :***
Deux autres indicateurs peuvent être utilisés pour mesurer la performance d'un classifieur :

$$\text{Taux de chute} = \frac{FP}{FP + VN}$$

$$\text{Taux de spécificité} = \frac{VN}{FP + VN}$$

f) ***L'overlap et la généralité :***

$$\text{L'overlap} = \frac{VP}{VP + FP + FN}$$

$$\text{La généralité} = \frac{VP}{VP + VN + FP + FN}$$

g) ***F-mesure :***
Observés conjointement, les indicateurs les plus célèbres à savoir le rappel et la précision, sont une estimation courante de la performance d'un système de classification. Cependant plusieurs mesures ont été développées afin de synthétiser cette double information. Nous ne retiendrons ici que la mesure F_β décrite dans (Van Rijsbergen, 1979). La F-mesure est la mesure de synthèse communément adoptée depuis les années 80 pour évaluer les algorithmes de classification de données textuelles à partir de la précision et du rappel. Elle est employée indifféremment pour la classification (Non supervisé) ou la catégorisation (Supervisé), pour la problématique de recherche d'information ou de classification. Elle permet donc, de combiner, selon un paramètre β, rappel et précision.On définit la mesure F_β comme la moyenne harmonique entre le rappel et la précision :

$$F_\beta = \frac{(\beta^2 + 1) * \text{Précision} * \text{Rappel}}{\beta^2 * \text{Précision} + \text{Rappel}}$$

Pour utiliser cette mesure, il est donc nécessaire de fixer préalablement un seuil de décision pour le classement, et de calculer la valeur de F_β pour ce seuil. Le paramètre β permet de choisir l'importance relative que l'on souhaite donner à chaque quantité. On choisit en général de donner la même importance aux deux critères, donc habituellement, la valeur deβ est fixée à 1 et la mesure est ainsi notée $F1$(notéF) qui s'écrit :

$$F_1 = \frac{2 * \text{Précision} * \text{Rappel}}{\text{Précision} + \text{Rappel}}$$

h) ***Matrice de contingence globale :***
Pour la catégorisation à plusieurs classes de textes, une approche commune consiste à couper le processus de catégorisation en sous-problèmes. Chaque sous-problème concerne uniquement une classe et l'objectif est alors de juger si le nouveau texte appartient ou n'appartient pas à cette classe par opposition aux autres. Pour la catégorisation multi-classes de textes, nous avons un ensemble de classes $C = C_1, ..., C_N$ où N est le nombre de classes ($N > 2$). Nous notons S_i le nombre de documents de C_i. Pour chacune des classes, nous pouvons calculer comme précédemment le rappel, la précision et la mesure $F1$, notés respectivement R_i , P_i et F_{1i}. Nous pouvons donc obtenir des mesures globales pour le système à N classes en moyennant ces mesures par classe. La précision et le rappel globaux, c-à-d, sur toutes les classes peuvent être calculés à travers une moyenne des résultats obtenus pour chaque catégorie. Cependant, si les classes ne possèdent pas le même nombre de documents, ces moyennes risquent de ne pas refléter la performance du classifieur pour les grandes classes. Les résultats de chaque catégorie peuvent être combinés de deux manières :
- On peut calculer un score pour chaque catégorie à partir de sa matrice de contingence puis déterminer la moyenne des scores sur l'ensemble des catégories (macro-averaging). Dans ce cas, toutes les catégories interviennent de la même manière dans le calcul du score final quelque soit le nombre de documents qu'elles contiennent.
- Une autre possibilité est de créer une table de contingence globale pour toutes les catégories (micro-averaging) : le contenu d'une cellule de cette table correspond à la somme des valeurs de la même cellule dans la table de chaque catégorie.

i) ***Matrice de contingence globale***

L'ensemble des catégories $C_1, ..., C_N$		Jugement Expert	
		Oui	Non
Jugement Classifieur	oui	$VP = \sum_{i=1}^{N} VP_i$	$FP = \sum_{i=1}^{N} FP_i$
	Non	$FN = \sum_{i=1}^{N} FN_i$	$VN = \sum_{i=1}^{N} VN_i$

TABLE 1.2: Tableau de contingence globale

j) ***La micro-moyenne :***

Les mesures de type micro moyenne correspondent à une moyenne qui pondère chaque classe par son effectif. La micro-moyenne (traduction de micro-averaging) calcule les mesures rappel et précision de façon globale : si l'on considère les tables de contingences associées à chaque catégorie, cela revient à sommer les cases VP, FP, FN et VN de chaque catégorie pour obtenir la table de contingence globale (voir le tableau 1.2).
Les différentes mesures sont ensuite calculées à partir des valeurs cumulées. La micro-moyenne accorde donc des poids importants aux catégories ayant beaucoup d'exemples. La performance du classifieur dépend surtout de sa capacité à traiter les catégories les plus fréquentes. Ainsi, la précision micro-moyenne et le rappel micro-moyenne sont estimés comme suit :

$$P = \frac{VP}{VP + FP} = \frac{\sum_{i=1}^{N} VP_i}{\sum_{i=1}^{N} VP_i + \sum_{i=1}^{N} FP_i}$$

$$R = \frac{VP}{VP + FN} = \frac{\sum_{i=1}^{N} VP_i}{\sum_{i=1}^{N} VP_i + \sum_{i=1}^{N} FN_i}$$

k) ***La macro-moyenne :***

Les mesures de type macro moyenne correspondent à une moyenne qui ne prend pas en compte la taille des classes. La macro-moyenne (traduction de macro-averaging) évalue d'abord indépendamment chaque catégorie. Ensuite, la performance globale du classifieur est calculée en faisant la moyenne des mesures individuelles. Les différentes catégories ont alors la même importance. La précision et le rappel macro-moyenne sont calculés comme suit :

$$P = \frac{\sum_{i=1}^{N} P_i}{N}$$

$$R = \frac{\sum_{i=1}^{N} R_i}{N}$$

Ainsi, les mesures de type micro moyenne permettent d'obtenir une estimation du système en privilégiant les classes de grande taille tandis que les mesures de type macro moyenne donnent une information quant aux performances d'un système sur les petites classes.

Chapitre 2

Classification automatique de textes arabes et état de l'art

L'abondance de données, due au développement de l'Internet et aux supports de stockage, est devenue une réalité qui croît de manière exponentielle, et rend le traitement et l'analyse de cette masse de données l'une des taches les plus importantes.
Plusieurs travaux se sont intéressés à la classification des documents textuels et l'extraction de l'information, bien que la majorité de ces travaux a été réalisée sur des documents écrits en caractères latins (français, anglais, espagnol,...etc). Et très peu des travaux se rapportent à la classification automatique des documents écrits en caractères arabes malgré la richesse morphologique de cette langue. D'où avant de citer ces travaux, on expose les spécificités et difficultés morphologique de cette langue.

2.1 La langue arabe

L'arabe est une langue parlée par plus de 200 millions de personnes. Elle est langue officielle d'au moins 22 pays, la péninsule arabique comme « l'Arabie saoudite, Bahreïn, les Émirats Arabes Unis,etc », le Moyen-Orient « l'Irak, la Jordanie, le Koweït, le Liban,...etc » et l'Afrique « l'Algérie, la Libye, le Maroc, la Tunisie,...etc ». C'est aussi la langue de référence pour plus d'un milliard de musulmans [10]. L'arabe peut être considérée comme un terme générique rassemblant plusieurs variétés :

– L'arabe classique : la langue du Coran.

– L'arabe standard moderne (l'ASM) : une forme un peu différenciée de l'arabe classique, et qui constitue la langue écrite de tous les pays arabophones. L'ASM reste le langage de la presse, de la littérature et de la correspondance formelle, alors que l'arabe classique appartient au domaine religieux et est pratiqué par les membres du clergé.

– Les dialectes arabes : malgré l'existence d'une langue commune, chaque pays a développé son propre dialecte. Issus de l'arabe classique, leurs systèmes grammaticaux respectifs affichent de nettes divergences avec celui de l'ASM. On peut regrouper ces dialectes en quatre grands groupes :

a) les dialectes arabes, parlés dans la Péninsule Arabique : dialectes du Golfe, dialecte du najd, yéménite .

b) les dialectes maghrébins : algérien, marocain, tunisien, hassaniya de Mauritanie.

c) les dialectes proche-orientaux : égyptien, soudanais, syro-libano-palestinien, irakien (nord et sud).

d) la langue maltaise est également considérée comme un dialecte arabe.

2.1.1 Caractéristiques de la langue arabe

L'alphabet arabe la langue arabe est une langue sémitique à la différence des langues latines elle s'écrit et se lit de droite à gauche, les majuscules n'existent pas, son alphabet compte 28 consonnes (29 si l'on compte la hamza, qui est plus ambiguë). La plupart d'entre elles, changent de forme selon leur position dans le mot, qu'elles sont isolées ou écrits au début, au milieu ou à la fin d'un mot et suivant les règles d'attachement de la lettre qui la précède (la lettre Kaf, ك) qui s'écrit isolée au début du mot (كـ), au milieu (ـكـ), et à la fin (ـك) voir le tableau 2.1.

Toutes les lettres se lient entre elles sauf les lettres (ا ,و ,ر ,ز ,د ,ذ) qui ne se joignent pas à gauche.

Lettre	Début	Milieu	Fin	Isolée
Kafe	كـ	ـكـ	ـك	ك
exemple	كتاب	سمكة	ديك	ك

TABLE 2.1: des exemples de variations de la lettre Kafe

Un mot arabe s'écrit avec des consonnes et des voyelles. Les voyelles sont ajoutées au-dessus ou au-dessous des lettres. L'arabe compte 6 voyelles qui sont aussi divisées en deux groupes, voyelles courtes(ُ , َ , ِ) et voyelles longues(ى,و,ا). Elles sont nécessaires à la lecture et à la compréhension correcte d'un texte, elles permettent de différencier des mots ayant la même représentation. Cependant, les voyelles ne sont utilisées que pour des textes sacrés et didactiques. Les textes courants rencontrés dans les journaux et les livres n'en comportent habituellement pas.

Morphologie de l' arabe la langue arabe a une morphologie riche et différente, par rapport aux langues latines. L'analyse morphologique d'un mot arabe, consiste principalement à déterminer la structure générale de ce mot, et les autres éléments utilisés pour construire ce mot. Les éléments essentiels de la morphologie de la langue arabe sont :

a) **La racine**
Une caractéristique importante de la langue arabe est de langue à racines réelles, c'est-à-dire ces dernières sont à l'origine de la plupart de mots arabes. On peut déduire le reste du lexique arabe par application de différents schèmes (modelés) morphologique, qui consiste en l'adjonction de voyelles et en manipulations de la racine. En effet, en arabe, les verbes et les noms sont le plus souvent issus d'une dérivation d'une racine de trois ou quatre lettres. Par exemple, la racine (كتب il a écrit) a la signification de base «écrire». Plusieurs mots sont dérivés à partir de cette racine, (يكتب il écrit), (كتبنا nous avons écrit), (كاتب auteur), (كتاب livre).

b) **Les modèles**
Le modèle arabe permet essentiellement de déterminer la structure de la plupart des mots (les noms, les verbes conjugués,...etc). Les modèles sont des déclinaisons du

mot (فعل, faire) qui sont obtenus en utilisant des diacritiques ou en y ajoutant des affixes. Par exemple, le modèle(فُعل, a été fait) est obtenu en utilisant les diacritiques, le modèle (مستفعل) est obtenu en y ajoutant le préfixe. Le mot (كُتِبَ, a été écrit) a pour modèle(فُعِل, a été fait), le mot (مستكتب, une personne employée pour écrire) a pour modèle(مستفعل).

c) **La structure d'un mot**
En arabe un mot peut signifier toute une phrase grâce à sa structure composée qui est une agglutination d'éléments de la grammaire, la représentation suivante(dans le tableau 2.2) schématise une structure possible d'un mot. Notons que la lecture et l'écriture d'un mot se fait de droite vers la gauche.

Enclitique	Suffixe	Corps schématique	Préfixe	Proclitique
نا	ون	تَذَكرُ	تَ	أَ
pronom suffixe complément du nom	suffixe verbal exprimant le pluriel	dérivé de la racine ذكر	préfixe verbal du temps de l'inaccompli	conjonction d'interrogation

TABLE 2.2: La structure du mot أتتذكرونا

- *Les proclitiques* sont des prépositions ou des conjonctions.
- *Préfixes et Suffixes* expriment les traits grammaticaux et indiquent les fonctions : cas du nom, mode du verbe et les modalités (nombre, genre, personne, ...etc).
- *Les enclitiques* sont des pronoms personnels.
- *Le corps schématique* représente la base de mot.

d) **Les formes de base d'un mot arabe**
La grammaire arabe traditionnelle ne connait que trois sous ensembles :noms, verbes et les particules. En effet la classe des particules a été étendue pour inclure différentes sous catégories qui, en réalité, appartiennent à d'autres classes telle que les pronoms démonstratifs ou relatifs qui constituaient des entrées nominales particulières.cette extension a aboutit à un réorganisation en quatre sous ensembles du lexique arabe : les verbes, les noms, les pronoms et les mots outils [31].

- **Les verbes :** un verbe est une entité exprimant un sens indépendant du temps. La majorité des verbes arabes sont formés sur des radicaux de 3 consonnes tel que le cas du verbe (خرج) et éventuellement 4 consonnes tel que le du verbe (زلزل). Ces racines peuvent donner naissance à plusieurs schèmes à la suite d'une ou plusieurs transformations morphologiques. Il s'agit dans ce cas de racine à schème augmenté, exemple(اَفعَّل, فاعَلَ, أفْعَلَ, تفَعَّلَ, تَفَاعَل, افْتَعَلَ, اسْتَفْعَلَ).
- **Les noms :** le système morphologique arabe distingue trois sous catégories :
  - *Les noms primitifs :* ce sont des noms qui ne peuvent pas être rattachés à une racine verbale, exemple (طاولة une table), (رأس une tête),...etc.
  - *Les noms dérivés ou déverbaux :* ce sont les noms qui peuvent êtres dérivés à partir d'une racine verbale.
  - *Les nombres :* exemple (صفر zéro), (تسعة عشر dix-neuf),...etc.

- **Les pronoms :** nous distinguons
  - *Les pronoms démonstratifs :* (أسماء الاشارة) : exemple (هَؤُلاءِ, هذا),...etc.
  - *Les pronoms relatifs(أسماء موصولة) :* ils se rapportent au nom ou au pronom personnel qui les précédé et que nous désignons par un antécédent, exemple (الَّذي celui), (الذين ceux),...etc.
  - *Les pronoms personnels(ضمائر منفصلة) :* servent à désigner :
    - la première personne (أنا je), (نحن nous).
    - la deuxième personne (أنْتَ tu masculin), (أنْتِ tu féminin), (أنتم vous masculin pluriel), (أنتن vous féminin pluriel) ,(أنتما vous duel).
    - la troisième personne :la personne absente, (هو il), (هي elle), (هما ils duel), (هم ils), (هن elles).
- **Les mots outils :** sont des entités qui servent à situer des faits ou des objets par apport au temps ou au lieu, nous citons :
  - *Les prépositions :*par exemple (في dans) , (على sur).
  - *Les conjonctions de coordination :* par exemple (ثم puis).
  - *Les adverbes :* par exemple (أبدا jamais) .
  - *Les quantificateurs :* par exemple (كل tout), (بعض un peu).
  - *...etc*

2.1.2 Problèmes du traitement automatique de la langue arabe

Par ses propriétés morphologiques et syntaxiques, la langue arabe présente les difficultés suivantes :

L'absence de voyellation de la majorité des textes arabes écrits : ce phénomène entraîne un nombre important d'ambiguïtés morphologiques. Si elles sont présentes, les voyelles brèves sont représentées par des diacritiques qui apparaissent au-dessus ou en dessous des consonnes qu'elles suivent. Elles sont nécessaires à la lecture et à la compréhension correcte d'un texte et elles permettent de différencier des unités lexicales ayant la même représentation.
par exemple le mot سلم

- signifie paix est voyellé ainsi سِلْمٌ
- signifie échelle est voyellé ainsi سُلَّمٌ
- signifie a transmis est voyellé ainsi سَلَّمَ
- signifie est guérie est voyellé ainsi سَلِمٌ

Hamza et Alef : les textes arabes courants confondent les lettres (أ et إ) au début et au milieu des mots. Ils les notent indifféremment en tant que (ا alif). Ce qui en plus

d'être une erreur d'orthographe présente une grande source d'ambiguïté. Exemple (سال il a coulé) et (سأل il a posé une question).

Ya et Alif maqsûra : l'absence des deux points change totalement le mot, et pose un vrai problème de reconnaissance de la forme écrite. La plus part des mots se terminant avec alif maqsûra ont un homographe avec le Ya. Exemple : le mot (علي un nom propre) et le mot (على un mot outils).

Le caractère ـ (kashida) : kashida permit l'allongement du trait au milieu des mots, pour une meilleure lisibilité, et limiter les espaces blancs sur une ligne justifiée. Ce caractère, ne faisant pas partie de l'alphabet arabe, est souvent une source de confusion pendant le traitement des textes arabes à titre d exemple (كتـــــاب et كتاب)

shadda : la shadda dans la langue arabe représente un accent plus élevé sur le caractère qui peut générer deux mots un avec la shadda (l'équivalent de la répétition de lettre en français : tt, mm,...etc) et un sans « shadda », ces deux mots peuvent avoir des sens différents (شدّ et شدد). Se place sur la lettre et marque le redoublement de la consonne. Le 'shadda' est toujours associé à une voyelle brève ou un signe de 'tanwin'. Ce signe n'affecte jamais la première consonne d'un mot. Il correspond au dédoublement de lettres en français.

La nature agglutinante de la langue : l'ensemble des morphèmes collés à l'unité lexicale véhiculent plusieurs informations morphosyntaxiques. Contrairement aux langues latines, en arabe, les articles, les prépositions, les pronoms,...etc. collent aux adjectifs, noms, verbes et particules auxquels ils se rapportent. Ces mots arabes sont souvent traduisibles par l'équivalent d'une phrase en français.
Par exemple le mot en arabe (أتتذكروننا) représente en français la phrase suivante : « Est-ce que vous vous souvenez de nous ? ». Cette caractéristique engendre une ambiguïté morphologique au cours de l'analyse. En effet, il n'est pas toujours facile de distinguer un proclitique ou enclitique d'un caractère original du mot. Par exemple, le caractère (و) dans le mot (وصل il est arrivé) est un caractère original alors que dans le mot (وفتح et il a ouvert), il s'agit plutôt d'une proclitique.

La flexion est la variation de la forme des mots : Une langue flexionnelle est une langue dans laquelle les unités lexicales varient en nombre et en flexion (soit le nombre des noms, soit le temps verbal) suivant les rapports grammaticaux qu'ils entretiennent avec les autres unités lexicales. L'ensemble des formes différentes d'une même unité lexicale fléchie constitue son paradigme [10]. Exemple : le mot (يتأثرون ils s'influencent) est le résultat de la concaténation du préfixe (ي) indiquant le présent et du suffixe (ون) indiquant le masculin pluriel du verbe (تأثّر).

2.1.3 Outils de traitement automatique de la langue arabe

Les outils de traitement automatique de la langue arabe sont l'ensemble des recherches et développements visant à modéliser et reproduire, à l'aide de machines, la capacité humaine à produire et à comprendre des énoncés linguistiques dans des buts de communication. Notre objectif dans cette section est de présenter les principaux outils de TAL en langue arabe, soit les analyseurs morphologiques et les racineurs(Stemming) de l'arabe.

a) **Les analyseurs morphologiques**
L'analyseur morphologique consiste après segmentation du texte, à étudier la forme d'un mot pris isolement (sans contexte) et à déduire les informations dérivationnelles et flexionnelles. Ainsi, l'analyseur doit générer pour le mot traité une ou plusieurs solutions morphologiques décrites par les informations suivantes : les suffixes, les préfixes, le radical, la forme canonique (lemme) ainsi que d'autres informations comme le genre grammatical(féminin, masculin), le nombre (singulier, pluriel) ou le temps (verbe conjugué au présent, au passé parfait,...etc).

AraMorph développé par Tim Buckwalter, en langage Perl pour le compte du LDC (Linguistic Data Consortium) Université de Pennsylva nie, actuellement sous Java. Cet Analyseur baptisé Aramorph. Le texte analysé en entrée doit être transformé en code ASCII (translitération)avant traitement et le résultat d'analyse doit retranscrit en arabe (translitération inverse) afin d'être compris. Il est réalisé à partir de trois fichiers de lexique : préfixes (299 entrées), suffixes (618 entrées) et les lexèmes (82158 entrées représentants 38600 lemmes) [31].

Xerox l'analyseur morphologique de Xerox permit de segmenter une phrase en unités lexicales par un transducteur à états finis. Ce transducteur découpe la chaîne d'entrée en une séquence d'unités lexicales qui peuvent correspondre à une forme fléchie, une marque de ponctuation,...etc. La deuxième étape est l'analyse morphologique des unités lexicales produites par la segmentation de la phrase. Cette étape est aussi réalisée par un transducteur qui relie la forme fléchie à la forme lexicale (et vice-versa). La forme lexicale est une séquence comprenant la représentation canonique de l'unité lexicale (le lemme), un ensemble d'étiquettes représentant le comportement morphologique de l'unité lexicale, et sa catégorie syntaxique Boulaknadel2008.

b) **Les Racineurs**
Le racineur consiste à détecter la racine d'une unité lexicale, Les algorithmes de racinisation(Stemming) sont classés en deux types :
- *Algorithmes à base de Stem (Light Stemmer) :* qui suppriment les affixes (préfixe, suffixe), ils utilisent une liste étendue d'affixes les plus fréquents pour extraire des racines. Parmi les implémentations :

Racineur de larkey L'approche est une analyse morphologique assouplie. Elle consiste à essayer de déceler les préfixes et les suffixes ajoutés à l'unité lexicale : par exemple le duel (ان) dans (معلمان, deux professeurs), le pluriel des noms masculins (ين, ون) dans (معلمون des professeurs) et féminins (ات) dans (مسلمات musulmanes). La forme possessive (هم, كم,نا) dans (كتبهم ses livres) et les préfixes

dans les articles définis (فال,كال,بال,وال,ال). L'ensemble des préfixes et suffixes à supprimer sont présentés dans le tableau 2.3

Préfixes			**Suffixes**	
1- Caractère	2- Caractère	3- Caractère	1- Caractère	2- Caractère
ت	ال	وال	ة	اة
ل	بت	فال	ه	ان
ا	ت	بال	ي	تا
ي	يت		ا	تك
م	لت			تي
	مت			ته
	وت			تم
	ست			هم
	نت			هن
	بم			ها
	لم			كم
	وم			وا
	كم			ون
	فم			وه
	لّ			ية
	وي			نا
	لي			ين
	سي			يه
	في			
	وا			
	فا			
	لا			
	با			

TABLE 2.3: des préfixes et suffixes

– *Algorithmes à base de racine (Stemming)* qui récupèrent la racine du mot.

Racineur de Khoja Khoja est développé en langage Java et utilise des listes des modèles (patterns) et des racines (root) pour extraire la racine d'un mot donné.
Shereen khoja est développé en langage Java, au sein de l'université de Lancaster, a été utilisé dans le cadre d'un système de recherche d'information développé à l'Université du Massachusetts. L'approche de Khoja consiste à utiliser des listes des modèles (patterns) et des racines (root) pour extraire la racine d'un mot

donné.
Pour détecter la racine d'une unité lexicale, il faut connaître le schème par lequel elle a été dérivée et supprimer les éléments flexionnels (préfixes et suffixes) qui ont été ajoutés. Ensuite, comparer la racine extraite avec une liste des racines préalablement conçue.

2.2 État d'art

La classification automatique de textes est un domaine qui propose plusieurs axes de recherche, ces derniers sont justifiés par l'application de différentes techniques dans le processus de classification, on trouve :

- Le choix d'un descripteur de texte : mot simple, lemme, racine, N-grammes, et concept [1].

- La réduction de la dimension de l'espace d'apprentissage, plusieurs techniques ont été proposées : Document Frequency(DF), Information Gain(IG), Mutual Information(MI), Gain Ratio(GR), Le Chi-deux(CHI2), Odds Ratio(OR) et Term Strength(TS) [2] .

- Le calcul du poids des descripteurs, les codages utilisés sont : Codage booléen, Term Frequency(TF), Term FrequencyInverse Document Frequency (TF/IDF), TFC et l'entropie [3].

- La dernière étape le choix d'un classifieur, parmi les techniques proposées Rachio, K-Nearest Neighbor(KNN), Naïve Bayes(NB), Support Vector Machines(SVM), etc [4].

Les travaux consacrés dans le domaine de la classification des textes arabes sont :

Saleh Alsaleem a prouvé la performance du classifier Support Vector Machines(SVM) par rapport au classifieur Naïve Bayes (NB) sur un corpus de 5121 textes répartis entre 7 catégories(Culture, Économie, Général, Information Technologique, Politique, Social, Sport), il a choisi le mot simple comme descripteur. Les résultats de l'expérience indique que le classifieur SVM est plus performant que Naïve Bayes(NB), la moyenne de F-mesure pour SVM= 0.778 et égale a 0.74 pour NB) [5].

Pour éviter les problèmes provenant de la grande dimension de l'espace d'apprentissage et améliorer l'efficacité et le bon fonctionnement des algorithmes de classification. Mosleh Abdelwadood a testé six techniques de réduction (CHI2, NGL, GSS, IG, OR et MI) sur un corpus de 1445 textes et 9 catégories(Computer, Économie, Éducation, Ingénierie, Droit, Médecine, Politique, Religion, et Sport), de

1. voir la section 1.2.2
2. voir la section 1.2.3
3. voir la section 1.2.4
4. voir la section 1.3

taille=14MO.qui est confectionné par lui meme. Et pour faire a comparé le classifieur Support Vector Machines(SVM) avec le K-Nearest Neighbor(KNN) et Naïve Bayes(NB). Il a trouvé le Macro-moyenne de F-mesure pour le classifieur SVM égale 88,11 est meilleur, qui dépasse NB et le KNN [30].

Le meilleur choix du descripteur des textes reste toujours un sujet de recherche et de débat. Al-Shalabi, Rasha Obeidat ont montré que l'utilisation de N-grammes pour représenter chaque texte produit de meilleures performances que d'utiliser des mots simples. Et pour faire il ont cherché à comparer la performance de l'algorithme K-Nearest Neighbor(KNN)(avec mesure de similarité cosinus) en utilisant la représentation bag of words (mots simple), et N-grammes(N=1, N=2) sur le corpus de Mosleh Abdelwadood . Ils ont choisi la méthode Document Frequency (DF) pour réduire la taille du corpus ainsi que la méthode TFxIDF pour calculer les poids des termes. Les résultats montrent que l'utilisation de N-grammes pour représenter chaque texte produit de meilleures performances que d'utiliser des mots simples. La moyenne de la précision dans le premier cas est 0,7353 tandis que dans le seconde elle est 0.6688 [4].

Une autre utilisation des N-grammes par Laila Khreisat, qui a comparé deux méthodes de classification de textes basées sur une mesure de dissemblance/similarité entre le texte à classer et les catégories. Pour la dissimilarité elle a choisi la distance de Manhattan, et la méthode qui mesure la similarité s'appelle Dice. Le corpus utilisé est taille de 43K0 avec 4 catégories(Sport, Économie, Technologie, Météo). Pour classer un nouveau texte dans une des quatre catégories la première méthode affecte la catégorie ayant la moindre dissemblance avec le texte tandis que la deuxième cherche à affecter celle ayant la plus grande similarité. Les résultats obtenus confirment que la classification basée sur la « mesure de Dice » est meilleur, avec la moyenne de la Précision $\cong$ 0.88 et du rappel $\cong$ 0.83 contre seulement la moyenne de la précision $\cong$ 0.66du rappel $\cong$ 0.56 pour l'autre méthode. L'auteur indique que la faiblesse de la méthode de « Manhattan » est due à la richesse morphologique de la langue arabe et qu'une analyse morphologique pourrait améliorer les résultats [23].

Karima Abbidi, Elberrichi Zakaria, Tlili Guisssa Yamina ont montré l'efficacité de la représentation conceptuelle des textes pour la classification de textes arabes. Une étude comparative a été menée sur les différents descripteurs de textes mot simple, N-grammes (N=2,3,4) et concept sur le corpus Mesleh et avec le classifieur K-Nearest Neighbor(KNN) [20].

D'autres préfèrent utiliser la Racine pour représenter les textes. Rehab Duwairi a comparé trois classifieurs, le Naïve Bayes(NB), K-Nearest Neighbor(KNN) et un classifieur basé sur la distance Dice sur un corpus d'apprentissage composé de 1000 textes répartis entre 10 catégorie(Sports, Économie, Internet, Médecine, Art, Animaux, Technologie, Plantes, Religion, Politique). Pour extraire les racines elle a utilisée une méthode qui affecte des poids à chaque lettre du mot. Enfin, cette méthode considère uniquement les 3 lettres ayant les plus petits poids comme la racine du mot. Pour mesurer la performance des classifieurs l'auteur utilise la Précision, le Rappel, fallout et Error rate. Les résultats montrent que l'algorithme NB était le

plus performant parmi les trois algorithmes utilisés. L'algorithme KNN (avec k=50) était second tandis que l'algorithme basé sur la distance était troisième [12].

La racine est utilisée aussi dans les travaux de Mohamed El kourdi, Amine Bensaid et Tajje-eddine Rachidi, qui ont cherché à étudier le classifieur Naïve Bayes(NB) sur un corpus de 300 textes par catégories (Sport, business, Culture, Art, Sciences, Santé). Le résultat obtenu de la moyenne de la précision égale à 68,78 [24].

Raheel a mené une étude comparative sur différents descripteurs Mot simple, Racine, Lemme et N-grammes(N=3 et 4) sur un corpus de 1250 textes, répartis entre cinq catégories préalablement choisies dont chacune contient 250 textes distincts. Les cinq catégories sont (Politique, Économie, Médecine, Science et Technologie, Sports). Les classifieurs utilisés dans ces recherches sont le Multinomial Naïve Bayes (MNB), et Support Vector Machines(SVM). La méthode du calcul du poids utilisée est TFxIDF et les méthodes de réduction sont Information Gain(IG) et CHI2. Le résultat adopte les racines comme descripteurs avec le classifieur SVM et la méthode de réduction CHI2. Ce choix est dû au fait que la performance des classifieurs basés sur des corpus conçus à partir des racines est plus stable que ceux basés sur des corpus conçus à partir des N-grammes dit l'auteur [34].

Dans les recherches de Bassam Al-Salemi, Mohd. Juzaiddin Ab Aziz, les différentes techniques pour la réduction de l'espace d'apprentissage(CHI2, NGL, GSS, IG, MI) et les descripteurs(Light Stemming, et N-grammes(N=3,4,5)) ont été combiné pour améliorer la classification des textes basé sur le théorème bayésien. Le simple Naïve Bayes (NB), multi-variante Bernoulli Naïf Bayes (MBNB) et Multinomial Naïve Bayes (MNB) sur un corpus 3172 textes et 4 Catégories(Art, Économie, Sport, Politique) se sont utilisées. La meilleure valeur de Macro-moyenne de F-mesure égale a (0,941) réalisé par MBNB lorsque les termes sont représentés par Light Stemming et sélectionnés par la fonction CHI2 [2].

Thabtah, Wa'el Musa Hadi, Gaith Al-shammare, se sont intéressés aux différentes méthodes de pondération, pour calculer le poids des termes TF(Term Frequency), IDF(Inverse Document Frequency), TFIDF(Combine Term Frequency et Inverse Document Frequency), WIDF(Weighted Inverse Document Frequency), ITF(Inverse Term Frequency), et LOGTF(Logarithmic Frequency), sur le classifieur Nearest Neighbor(KNN) avec plusieurs types de similarités (cosine, dice, jaccard) et k=11.Les meilleurs résultats obtenus avec Dice basé TF.IDF et Jaccard basé TF.IDF est de moyenne de F-mesure(égale a 94,91) [39].

Al-Harbi, A.Almuhareb, A.Al-Thubaity,M.S.Khorsheed,A.Al-Rajeh ont comparé la performance de deux classifieurs Support Vector Machines(SVM) et Arbre de décision C5.0 sur sept corpus arabes (17,658 textes avec 11,500,000 mots). et comme descripteur pour les textes ils ont pris mot simple. L'algorithme C5.0 a prouvé sa performance sur l'algorithme SVM différence de 10% , la moyenne de la précision SVM est de 68,65%, tandis que la moyenne de la précision pour le $C5.0$ est 78,42%. l'auteur confirme que le seul inconvénient de l'algorithme C5.0, c'est qu'il est un algorithme de boîte noire, qui est disponible seulement dans le commerce [37].

d'après ces recherches on peut conclure :

Il est difficile de comparer ces recherches pour deux raisons, la première les chercheurs n'utilisent pas les mêmes jeux de données, le seul corpus qui était utilisé dans plusieurs recherches, le corpus Mosleh. La deuxième raison c'est que les auteurs ont utilisé différentes mesures d'évaluation du classifieur : Taux d'erreurs, Précision, Rappel, F mesure, etc .voir le tableau 2.4

Le classifieur Support Vector Machines(SVM) est comparé avec plusieurs techniques de classification, Arbre de décision, Naïve Bayes(NB), K-Nearest Neighbor(KNN), est déclaré performant dans plusieurs recherches [29], [37], [34], [5] .

La méthode de réduction CHI2 est utilisée dans plusieurs recherches [29], [37], [2], [34], [20]. Elle est comparée avec les autres méthodes de réduction (DF, NGL, GSS, IG, MI) dans les travaux de [30] et [2] elle produit les meilleurs résultats .

Le codage DFxIDF est utilisé dans la plupart des recherches[30], [24], [37], [4], [34]. Il été comparé avec les autres codages (TF(Term Frequency), IDF(Inverse Term Frequency), WIDF(Weighted Inverse Term Frequency), ITF(Inverse Term Frequency), et LOGTF(Logarithmic Frequency)), dans les travaux de [39], il est déclaré meilleur.

La racine comme descripteurs de textes est choisi dans plusieurs recherches [12], [24]. Il est comparé avec Mot simple, Lemme, N-grammes est déclaré meilleurs dans les travaux de [34].

La plupart des recherches sont faites sur un seul corpus, et généralement ces corpus sont de taille réduite, et nombre de catégories limité. Voir la tableau 2.4

TABLE 2.4: Un récapitulatif de l'état d'art

État	Corpus	Descripteur	Réduction	poids	Classifieur	Résultats
[23]	43ko 4catgs	N-gram (N=3)		TF	Manhattan, Dice	Dice Précision=0,88 Rapel=0,83
[30]	1445textes 9catgs 14MO	Mot Simple	CHI2,NGL, GSS,IG, OR, MI	TFxIDF	SVM,KNN, NB	SVM CHI2 F1=88,11
[12]	1000textes 10catgs	Racine		IDF	NB,KNN, DICE	NB
[24]	1800textes 6catgs	Racine		TFxIDF	NB	Précision=62
[37]	7corpus différents	Mot simple	CHI2	TFxIDF	SVM, C5.0	C5.0 Précision=78,42
[4]	Corpus Mesleh	Mot Simple, N-gram	DF	TFxIDF	KNN	KNN N-gram Précision=0,73
[39]	6catgs	Mot Simple		TF,IDF, TFxIDF, WIDF,ITF, LOGTF	KNN (Dice, Cosine, Jaccard)	KNN (Dice, Jaccard) TFxIDF F1 =94,91
[2]	3172textes 4catgs	Light Stemming, N-gram	CHI2,NGL, GSS,IG, MI,ODD	TF	NB,MBNB, MNB	MBNB Light stem CHI2 F1=0,941
[34]	1250textes 5catgs	Mot Simple, Racine, Lemme N-gram	CHI2 IG	TFxIDF	SVM, MNB	Racine SVM
[5]	5121textes 7catgs	Mot Simple			NB, SVM	SVM F1=0,778
[20]	Corpus Mosleh	Mot Simple, N-gram, Concept	CHI2		KNN	KNN Concept F1=0,74

Chapitre 3

Expérimentations et résultats

3.1 Introduction

Le problématique du travail que nous avons mené est de tester la classification automatique de textes arabes en utilisant :
- Trois corpus différents pour notre expérimentation ;
- L'algorithme d'apprentissage Support Vector Machines(SVM) comme classifieur ;
- utiliser le Light Stemming et le Stemming pour choisir le descripteur de texte ;
- La Précision, le Rappel et F-mesure pour évaluer la performance du classifieur.

3.2 Description des corpus utilisés

Un Corpus est un ensemble de textes étiquetés et répartis préalablement dans des catégories prédéfinies. Dans cette section seront présentés les différents corpus qui ont été utilisés au cours de nos expérimentations. Une attention a été portée au choix de ces corpus. Nous voulions introduire une certaine variété en ce qui concerne les catégories (nombre et thème), la taille de ces corpus qui est un aspect important pour que nos résultats soient statistiquement significatifs [34].

Le corpus Mosleh : a été conçu par Mesleh Abdelwadood à partir de différents journaux tels que « Al-Jazera, Al - Nahar, Al-Hayat et Al-Dostor », Il contient 964 textes repartis en 9 catégories(Politique, Éducation, Sport, Religion, Économie, Médecine, Ingénierie, Droit, Computer). La taille du corpus est 14MO voir la figure 3.1.

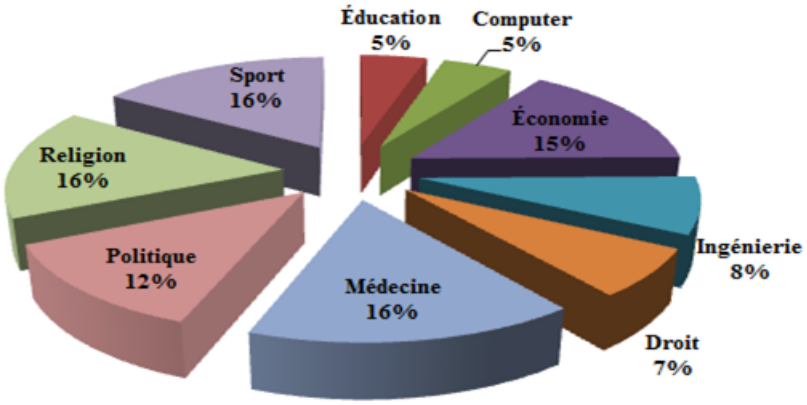

FIGURE 3.1: Le corpus Mosleh

Les corpus Watan et Khaleej : Sont construits par le Dr Mourad Abbas directeur du département de linguistique informatique et chercheur dans le traitement automatique de la langue arabe à crstdla Algérie. Ils ont été extrait à partir des milliers d'articles qui ont été téléchargés à partir d'un journal en ligne.

- Le corpus Watan 2004 contient 20291 textes organisés en 6 catégories(Sport, Économie, Culture, International News, Local News, Religion). La taille du corpus est 110 MO voir la figure 3.2.
- Le corpus Khaleej 2004 contient 5698 textes. Il est divisé en 4 catégories(Sport, Économie, International News, local News). La taille du corpus est 28,1 MO voir la figure 3.3.

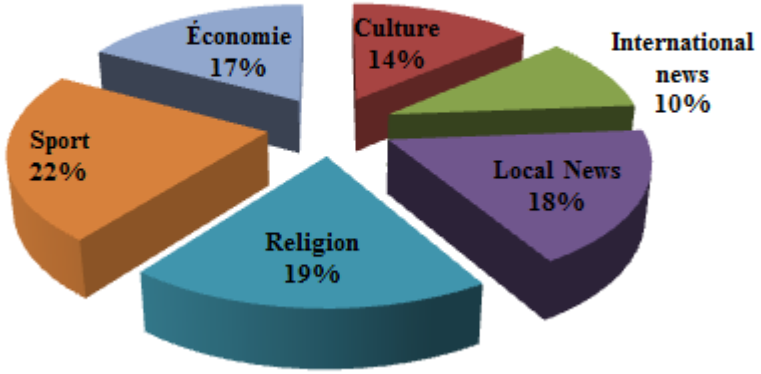

FIGURE 3.2: Le corpus Watan

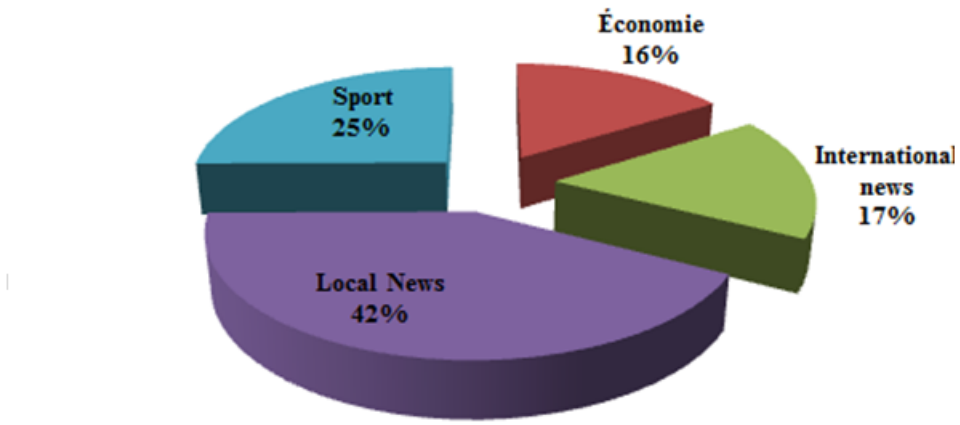

FIGURE 3.3: Le corpus Khaleej

Corpus	Nombre de catégories	Catégories	Nombre de textes	Taille
Mosleh	9	Politique, Éducation, Sport, Religion, Économie, Médecine, Ingénierie, Droit, Computer	964	14 MO
Watan	6	Sport, Économie, Culture, International News, Local News, Local News, Religion	20291	110 MO
Khaleej	4	Sport, Économie, International News, local News	5689	28,1 MO

TABLE 3.1: Un récapitulatif des corpus utilisés

On remarque bien que :

- Le premier corpus choisi est le corpus Mosleh largement utilisé dans les travaux de classification de textes arabes[30],[29],[4],[20].
- Les corpus utilisés dans l'état d'art, ont une taille en KO, sauf le corpus Mosleh , qui est de taille 14 MO, les autres corpus choisis sont de taille plus grande.
- Concernant les catégories, elles sont différentes en nombre et en thème.
- On voit bien que la répartition des textes est homogène dans les corpus, sauf pour la corpus Mosleh on trouve des catégories avec un nombre de textes faibles (Computer, Ingénierie, Éducation et Droit) .
- Les corpus sont confectionnés par différents concepteurs, le premier par Mesleh Abdelwadood, et les deux autres sont conçus par le même auteur Mourad Abbas. voir le tableau 3.1

3.3 Résultats et Discussion

3.3.1 Le Prétraitement des corpus :

C'est une phase très importante dans le processus de classification automatique de textes. Elle consiste à nettoyer les textes pour améliorer les résultats. Pour cette opération on a développé une application dans l'environnement NetBeans IDE 7.3.1 qui permet :

(a) Éliminer les caractères de séparation, les chiffres et les marques de ponctuations et les caractères non arabes ;

(b) Tous les mots écrits en caractères arabes partiellement voyellés sont dévoyellés ;

(c) Les différentes représentations morphologiques de la lettre (ا, Alef) (أ ,آ ,إ) sont normalisées et remplacées par (ا) ;

(d) Les différentes représentations morphologiques de la lettre (ى ,Yaa) (ي ,ئ) sont normalisées et remplacées par (ى) ;

(e) Remplacer le lettre (ة ,Taa Marbouta) par ه ;

(f) Tous les « mots vides » sont éliminés. Les « mots vides » sont par exemple, les prépositions, les articles, les conjonctions,...etc, qui ne portent pas de sens. Les figures 3.4 et 3.5 présente un exemple d'un texte pris du corpus Mosleh, catégorie computer avant et après l'opération de prétraitement.

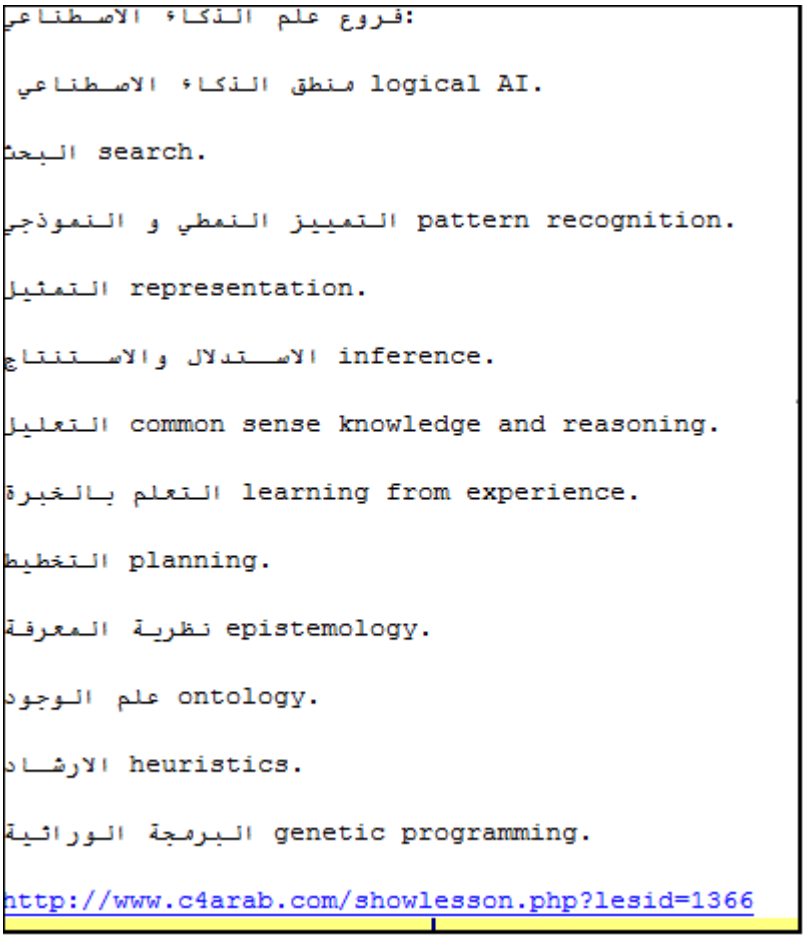

FIGURE 3.4: Un texte du corpus Mosleh (catégorie computer) avant le prétraitement

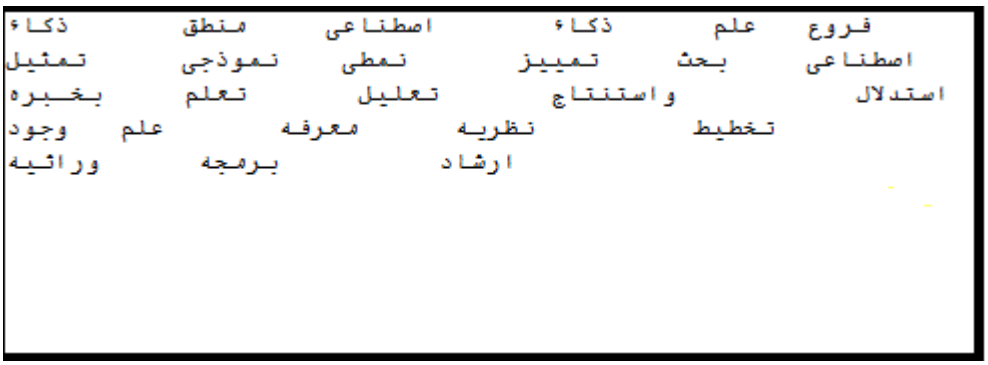

FIGURE 3.5: Un texte du corpus Mosleh catégorie computer après le prétraitement

– **Les résultats de prétraitement sur le corpus Mosleh** Après une phase de prétraitement morphologique et la suppression des mots vides, la moyenne des taux de réduction pour les catégories est égale à 35%. Avant cette opération, le nombre total de mots égale à 930158 pour le corpus entier et aprés le prétraitement, il a diminué de 333110 mots. Le tableau 3.2 et la figure 3.6 montrent le nombre de mots avant et après le prétraitement pour chaque catégorie .

Catégories	Nombre de textes	Nombre de mots avant le prétraitement	Nombre de mots après le prétraitement
Éducation	45	64239	42069
Computer	47	15970	8866
Économie	147	131711	89720
Ingénierie	77	141584	83777
Droit	65	144275	93177
Médecine	155	95776	63056
Politique	121	83215	56926
Religion	152	184895	111993
Sport	155	68493	47464
Total	**964**	**930158**	**597048**

TABLE 3.2: Corpus Mosleh avant et après le prétraitement

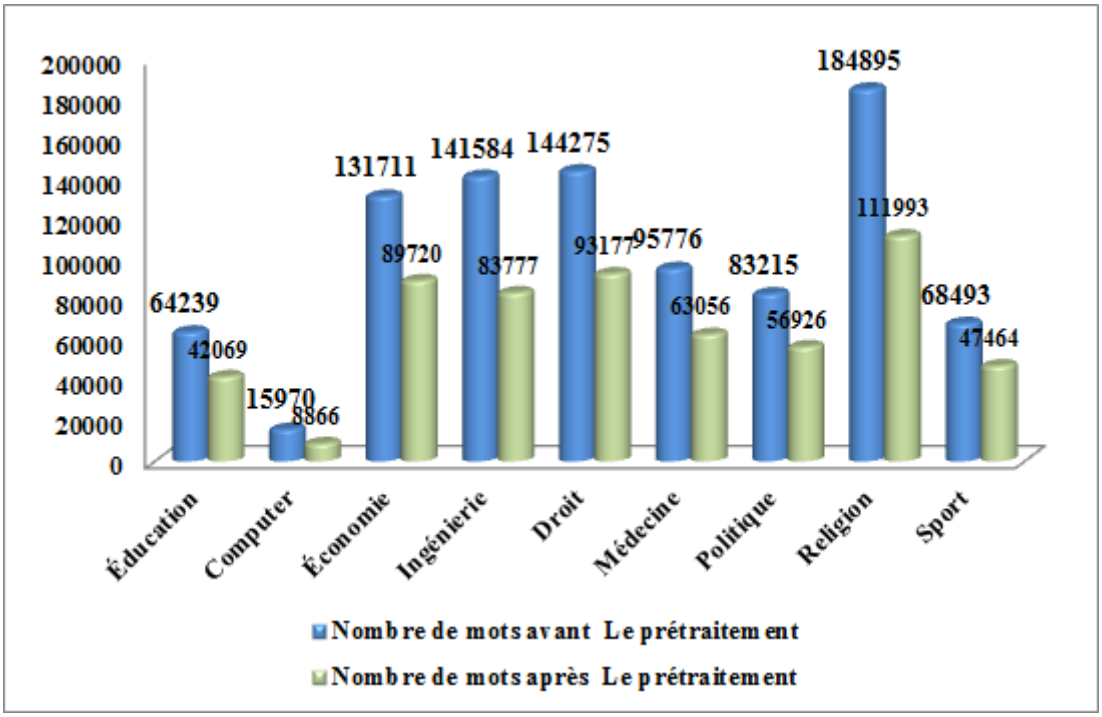

FIGURE 3.6: Corpus Mosleh avant et après le prétraitement

– **Les résultats de prétraitement sur le corpus Watan**
La moyenne des taux du réduction est égale à 30 % après le prétraitement sur le corpus Watan, au début le nombre de mots est égale à 10119405 sur le corpus entier et après le prétraitement il est égale à 6928767. Le tableau 3.3 et la figure 3.7 montrent la répartition du nombre des mots avant et après le prétraitement sur les catégories.

Catégories	Nombre de textes	Nombre de mots avant le prétraitement	Nombre de mots après le prétraitement
Culture	2782	1441923	994944
International News	2035	877227	618549
Local News	3596	1587715	1164622
Religion	3860	3235523	2022153
Sport	4550	1472756	1049294
Économie	3468	1504261	1079205
Total	**20291**	**10119405**	**6928767**

TABLE 3.3: Corpus Watan avant et après le prétraitement

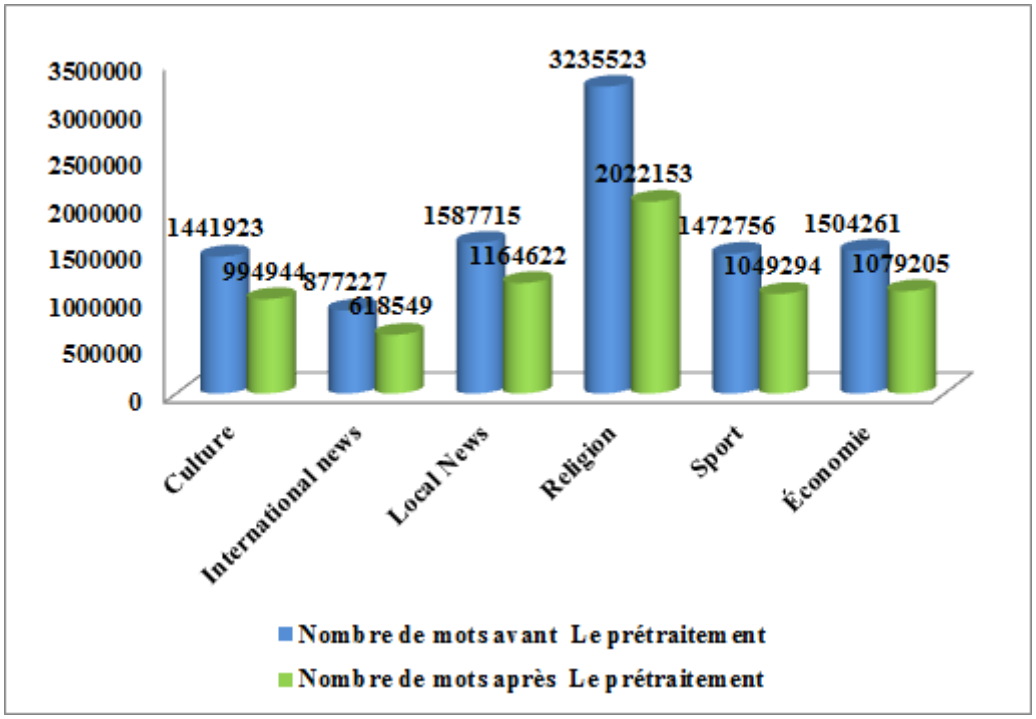

FIGURE 3.7: Le Corpus Watan avant et après le prétraitement

– **Les résultats de prétraitement sur le corpus Khaleej**
Sur le corpus Khaleej la moyenne des taux de réduction est égale à 26 %. Avant le prétraitement le nombre de mots sur le corpus entier est égale à 2471720 et après cette opération il est égale à 1817996. Le tableau 3.4 et la figure3.8 donnent le nombre de mots avant et après le prétraitement pour chaque catégories.

Catégories	Nombre de textes	Nombre de mots avant le prétraitement	Nombre de mots après le prétraitement
Économie	908	417935	304602
International News	953	534532	391161
Local News	2398	967525	715400
Sport	1430	551728	406833
Total	**5689**	**2471720**	**1817996**

TABLE 3.4: Corpus Khaleej avant et après le prétraitement

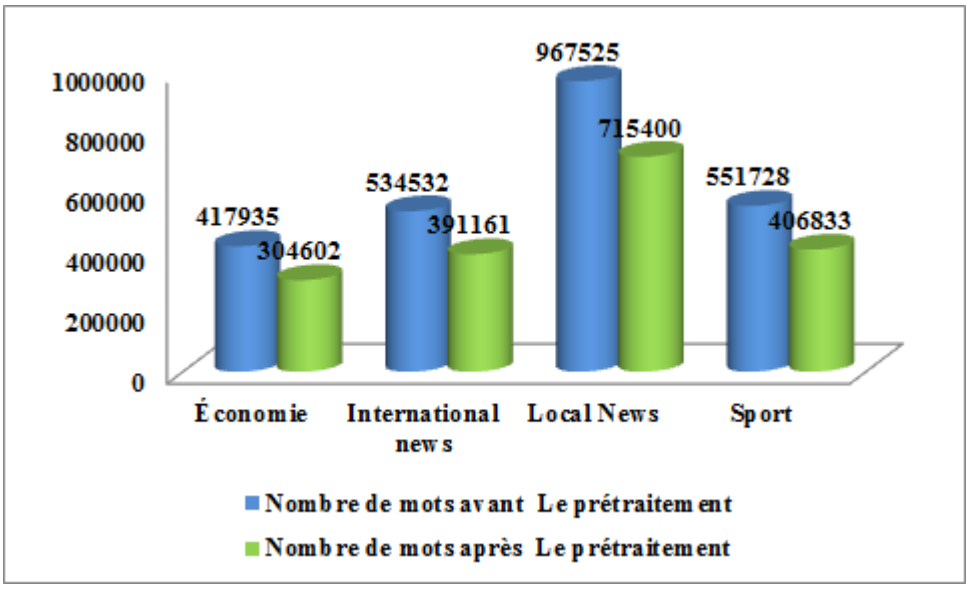

FIGURE 3.8: Le corpus Khaleej avant et après le prétraitement

– **Les résultats du prétraitement sur les trois corpus** Les résultats obtenus montrent l'effet de l'opération de prétraitement sur les trois corpus. Le prétraitemnet a réduit la taille du corpus d'environ 30 % voir la figure 3.9

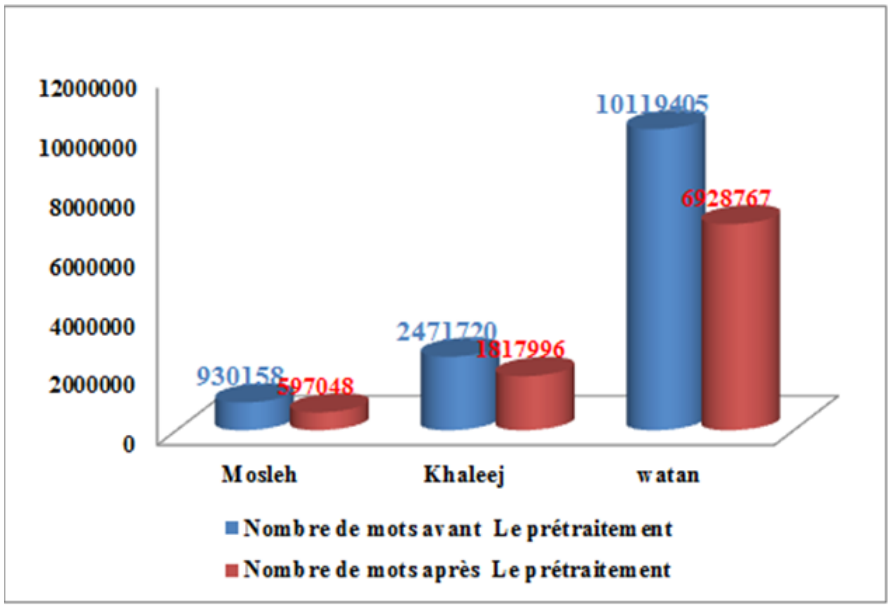

FIGURE 3.9: Les trois corpus avant et après le prétraitement

3.3.2 Le choix de descripteur du texte

La représentation des textes est une étape importante pour la catégorisation automatique des textes. Dans notre étude on a comparé trois approches pour choisir un descripteur de texte bag of words, Light Stemming et Stemming . Et pour faire on a utilisé les méthode Arabic Light Stemming et Khoja Stemming qui sont intégrés dans Weka. Ces deux méthodes sont décrites comme suit :

Algorithm 3 Arabic Light Stemming

Entrée : Mot d'entrée, liste des affixes ;
Sortie : Racine ;

1 : Éliminer les diacritiques ;
2 : Replacer (أ , آ ,إ) par ا;
3 : Replacer ة par ه;
4 : Replacer ي par ى;
5 : Supprimer les Préfixes, on utilisant la liste des affixes ;
6 : Supprimer les suffixes, on utilisant la liste des affixes ;

L'approche de l'algorithme 3 est principalement basée sur la normalisation et l'élimination des affixes. Une liste étendue d'affixes les plus fréquents est utilisée pour extraire la racine.

Algorithm 4 Khoja Stemming

Entrée : Mot d'entrée, liste des affixes, liste des Modèles, liste des racines ;
Sortie : Racine ;

1 : Éliminer les diacritiques ;
2 : Replacer (أ , آ ,إ) par ا;
3 : Replacer ة par ه;
4 : Replacer ي par ى;
5 : Supprimer les Préfixes, on utilisant la liste des affixes ;
6 : Supprimer les Suffixes, on utilisant la liste des affixes ;
7 : Compare le mot restant avec des motifs verbaux et nominaux(on utilisant la liste des modèles) pour l'extraction de la racine ;
8 : Valider cette racine par rapport à une liste de racines connues.

Khoja Stemming figure parmi les approches efficaces pour le Stemming du texte arabe [11]. L'approche de l'algorithme 4 consiste à enlever les affixes après une première étape de normalisation. Ensuite, le résultat est comparé avec une liste de modèles. Si une correspondance est trouvée, les lettres représentant la racine dans le modèle sont extraits. Ensuite, la racine ainsi extraite est validée dans une liste de racines connues.

l'effet des différents descripteurs sur la taille des corpus D'après les résultats obtenus dans le tableau 3.5, la réduction de la taille des trois corpus augmente de plus en plus en appliquant les différentes représentations Bag of Words , le Light Stemming et le Stemming voir la figure 3.10

Corpus	Nombre de mots		
	Mot	Light Stemming	Stemming
Mosleh	5176	4231	2650
Watan	3143	2671	1765
Kaleej	2404	2139	1558

TABLE 3.5: Le choix de descripteur sur les trois corpus

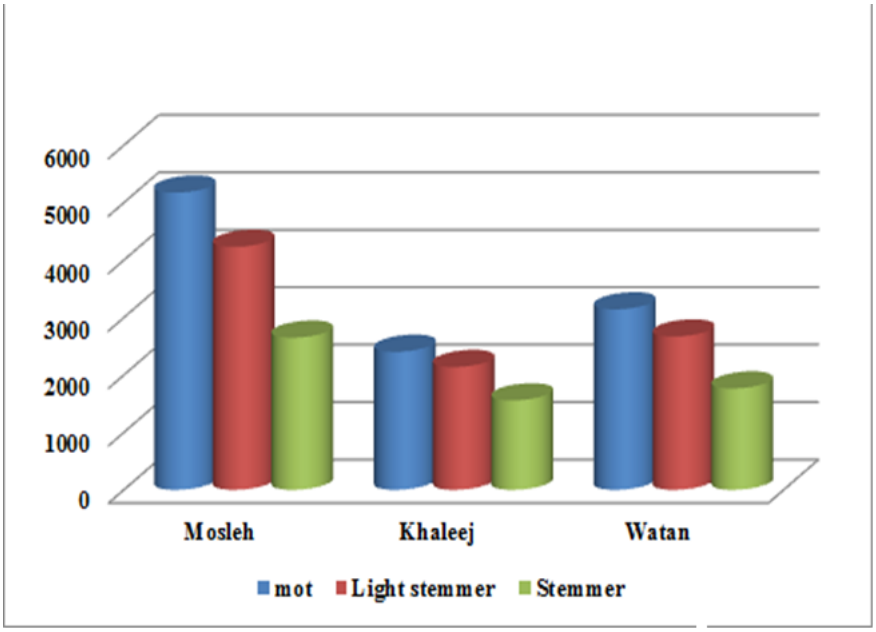

FIGURE 3.10: Le choix de descripteur sur les trois corpus

3.3.3 Évaluation du classifieur SVM sur le trois corpus

Il existe plusieurs outils et librairies Java qui permettent de mettre en œuvre l'algorithme SVM parmi les outils les plus populaires nous trouvons Weka, un logiciel développé à l'université de Waikato en Nouvelle-Zélande. Il dispose de plusieurs algorithmes d'apprentissage automatique et d'analyse de données. Nous avons choisi d'utiliser Weka parce qu'il est écrit en Java et il est simple à intégrer dans nos programmes java. En plus, il dispose d'une interface graphique qui nous permet de tester et d'évaluer les résultats avant de l'appliquer.

Pour évaluer le classifieur SVM sur les corpus on a utilisé trois mesures :
– *La précision* mesure la capacité du système à refuser les solutions non-pertinentes. Une précision de 1 correspond à l'absence totale de faux positifs. Une précision nulle indique un résultat ne contenant aucun document pertinent.
– *Le rappel* la capacité du système à donner toutes les solutions pertinentes.
– *F-mesure* moyenne harmonique de la précision et du rappel. Mesure la capacité du système à donner toutes les solutions pertinentes et à refuser les autres.
pour plus de détail sur ces mesures voir la section 1.4 .

a) **Évaluation du classifieur SVM sur le Corpus Mosleh** Les résultats des tests sur le corpus Mosleh sont résumés dans le tableau 3.6 :

Catégorie	Descripteur								
	Mot Simple			Light Stemming			Stemming		
	P	R	F1	P	R	F1	P	R	F1
Éducation	0,900	0,500	0,643	0,818	0,500	0,621	0,929	0,722	0,813
Computer	1,000	0,100	0,182	0,500	0,200	0,286	1,000	0,600	0,750
Économie	0,765	0,456	0,571	0,778	0,491	0,602	0,736	0,684	0,709
Ingénierie	0,657	0,263	0,161	0,182	0,629	0,282	0,277	0,657	0,390
Droit	0,667	0,333	0,444	0,778	0,389	0,519	0,750	0,500	0,600
Médecine	0,781	0,472	0,588	0,778	0,528	0,629	0,971	0,642	0,773
Politique	0,893	0,658	0,758	0,743	0,684	0,712	0,744	0,763	0,753
Religion	0,933	0,596	0,727	0,906	0,617	0,734	0,974	0,787	0,871
Sport	1,000	0,846	0,917	1,000	0,846	0,917	1,000	0,923	0,960
Moyenne	**0,789**	**0,570**	**0,625**	**0,758**	**0,595**	**0,641**	**0,821**	**0,726**	**0,754**

TABLE 3.6: SVM sur le corpus Mosleh

Les meilleurs résultats de la précision sont dans la grande majorité des cas obtenus par l'utilisation de l'approche Stemming pour choisir le descripteur du texte. La moyenne de la précision est égale à 0,821 pour Stemming, 0,758 pour le Light Stemming et 0,789 pour Bag of Words voir la figure 3.11.

les rappels sont faibles pour les catégories éducation, computer, ingénierie et droit, voir la figure 3.12, ceci s'explique par le fait que ces catégories contiennent un nombre réduit de textes voire la figure 3.1. L'utilisation de Stemming a amélioré le rappel qui est égale à 0,726 .

Les résultats de F-mesure se sont améliorés en utilisant le Stemming avec une moyenne égale à 0,754 voir la figure 3.13. Une légère amélioration en comparaison avec les résultats de Bag of Words et Light Stemming .

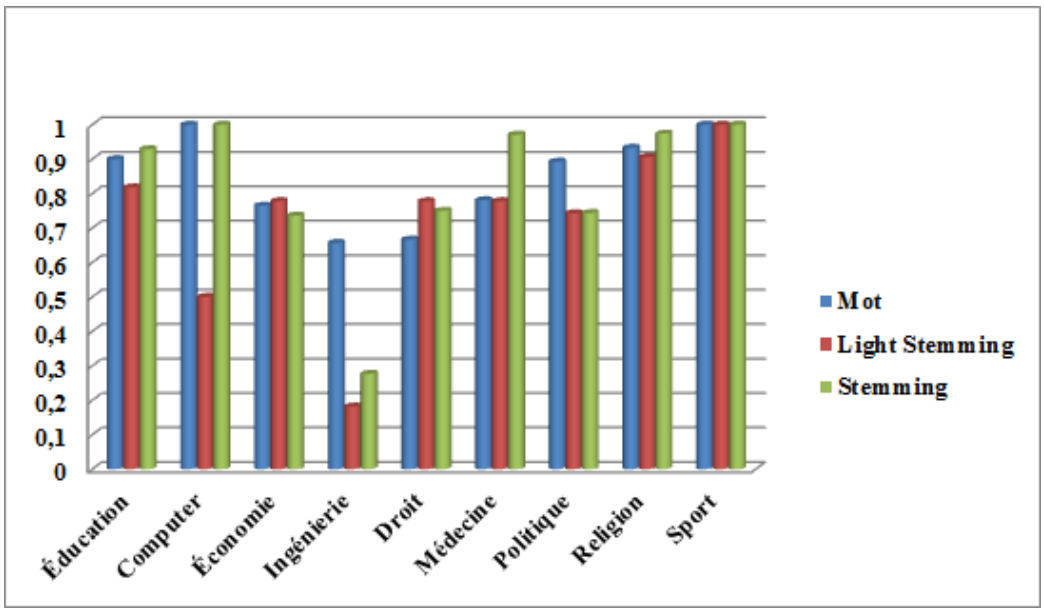

FIGURE 3.11: La Précision sur le corpus Mosleh

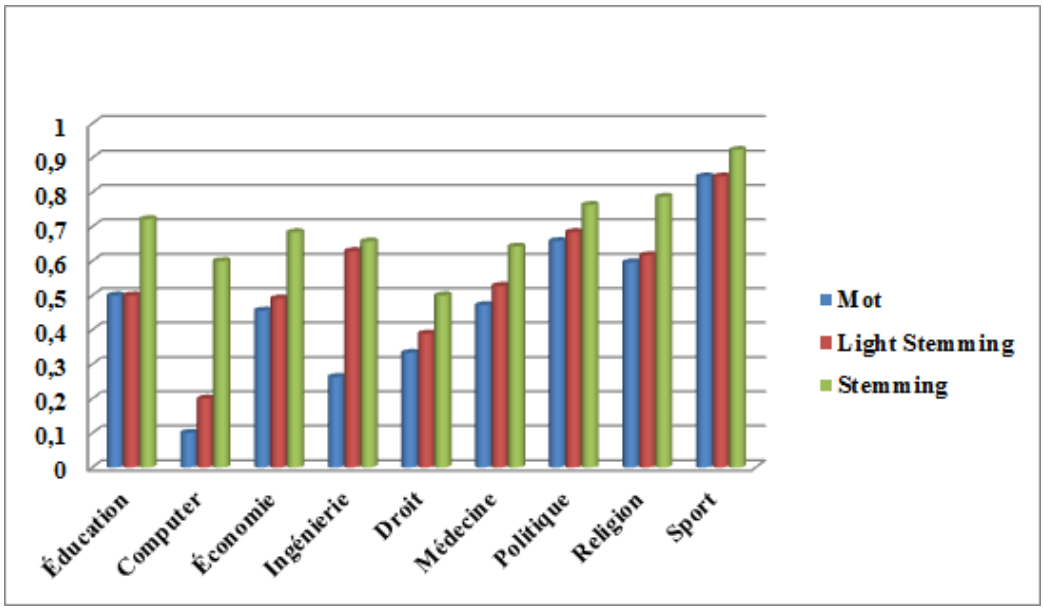

FIGURE 3.12: Le Rappel sur le corpus Mosleh

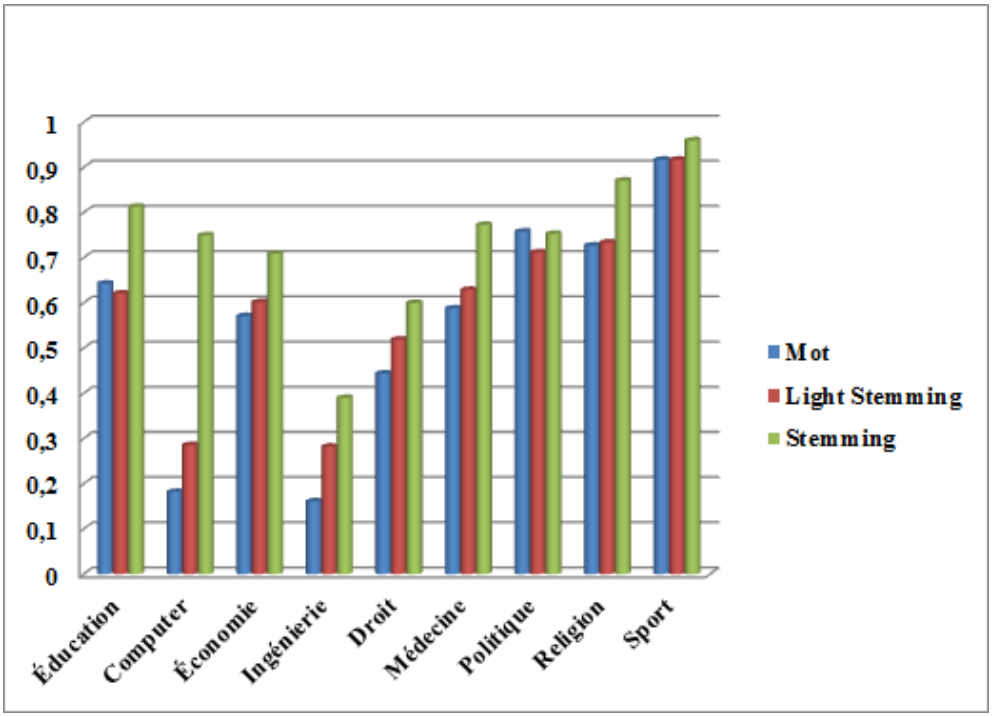

FIGURE 3.13: F-Mesure sur le corpus Mosleh

b) **Évaluation du classifieur SVM sur le Corpus Watan** Les résultats des tests sur le corpus Watan sont résumés dans le tableau 3.7 :

Catégorie	Descripteur								
	Mot Simple			Light Stemming			Stemming		
	P	R	F1	P	R	F1	P	R	F1
Culture	0,835	0,912	0,872	0,897	0,901	0,899	0,899	0,912	0,906
International news	0,952	0,890	0,920	0,955	0,920	0,937	0,949	0,920	0,934
Local news	0,871	0,865	0,868	0,852	0,860	0,856	0,873	0,877	0,875
Religion	0,989	0,997	0,993	0,987	0,998	0,993	0,986	0,998	0,992
Sport	0,987	0,979	0,983	0,985	0,983	0,984	0,989	0,985	0,987
Économie	0,896	0,873	0,885	0,874	0,873	0,874	0,892	0,886	0,889
Moyenne	**0,928**	**0,927**	**0,927**	**0,929**	**0,929**	**0,929**	**0,936**	**0,935**	**0,935**

TABLE 3.7: SVM sur le corpus Watan

Le classifieurs SVM sur le corpus Watan donne une Précision égale à 0,936 voir la figure 3.14. Une légère amélioration est signalée en utilisant le Stemming par apport au mot simple et Light Stemming.

Les résultats du rappel ont été légèrement améliorés en utilisant la méthode Stemming voir la figure 3.15.

Le SVM basé sur le Stemming comme attribut est toujours dans la première position avec la moyenne de F-mesure égale à 0,935 voir la figure 3.16.

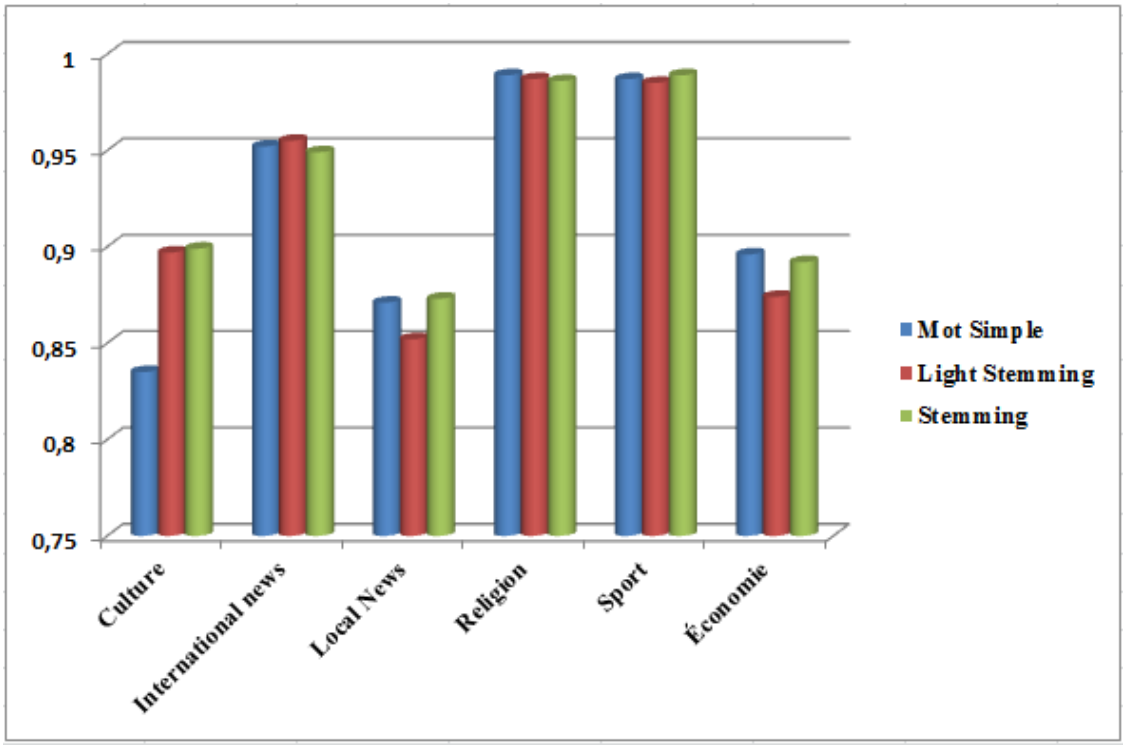

FIGURE 3.14: La Précision sur le corpus Watan

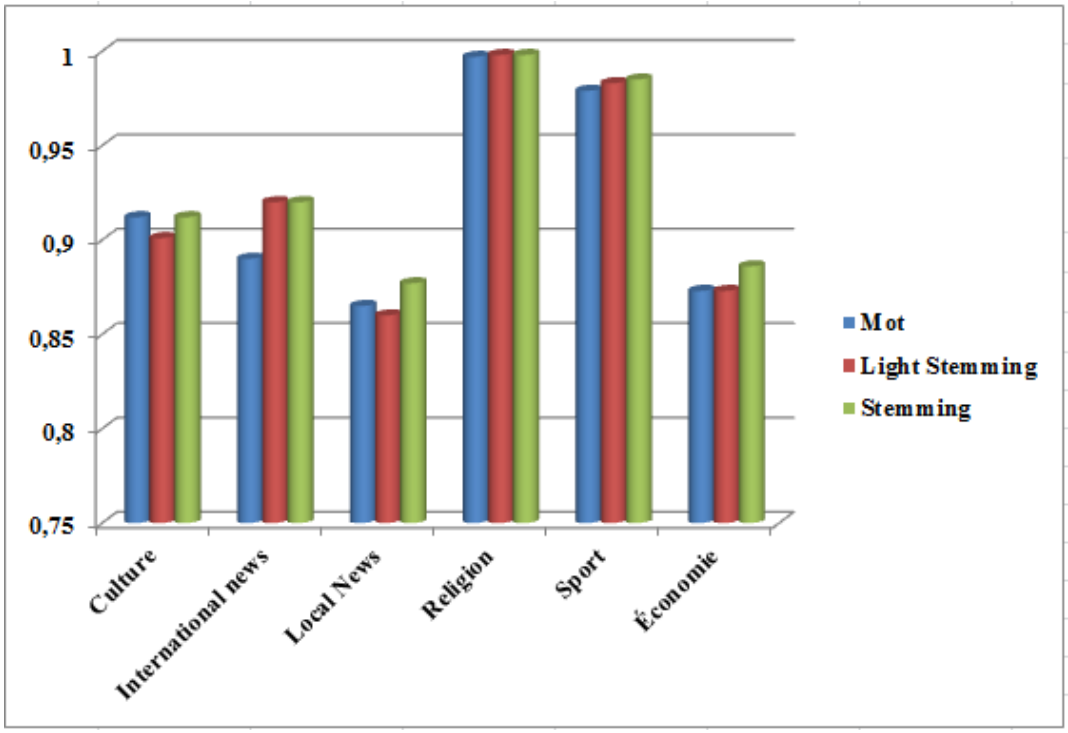

FIGURE 3.15: Le Rappel sur le corpus Watan

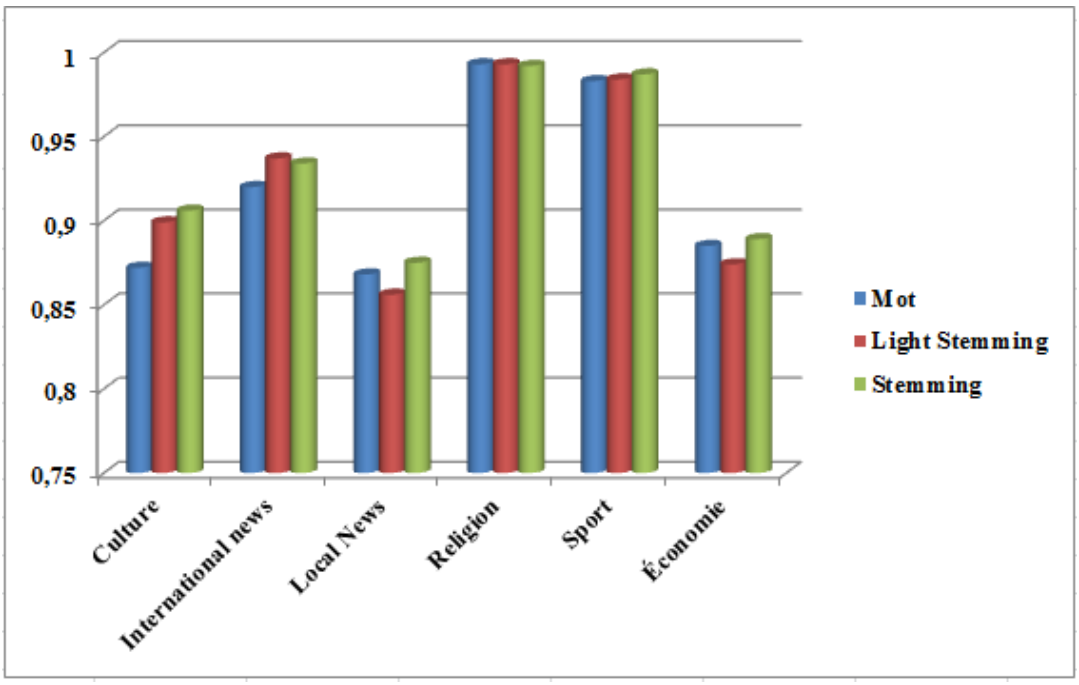

FIGURE 3.16: F-Mesure sur le corpus Watan

c) **Évaluation du classifieur SVM sur le Corpus Khaleej** Les résultats des tests sur le corpus Khaleej sont résumés dans le tableau 3.6 :

Catégorie	Decripteur								
	Mot Simple			Light Stemming			Stemming		
	P	R	F1	P	R	F1	P	R	F1
Économie	0,898	0,758	0,822	0,867	0,782	0,823	0,889	0,788	0,836
International news	0,973	0,883	0,926	0,970	0,893	0,930	0,970	0,905	0,937
Local news	0,834	0,948	0,888	0,849	0,934	0,890	0,858	0,941	0,897
Sport	0,978	0,925	0,951	0,979	0,937	0,957	0,979	0,945	0,962
Moyenne	**0,905**	**0,899**	**0,899**	**0,906**	**0,902**	**0,902**	**0,913**	**0,910**	**0,910**

TABLE 3.8: SVM sur le corpus Khaleej

La précision est bonne sur toutes les catégories du corpus Khaleej, avec une moyenne égale à 0,913 voir la figure 3.17.

Les valeurs de rappel obtenues sont meilleures sur toutes les catégories du corpus Khaleej, avec une moyenne égale à 0,913 voir la figure 3.18.

La moyenne de F-Mesure égale à 0,913 voir la figure 3.19 .

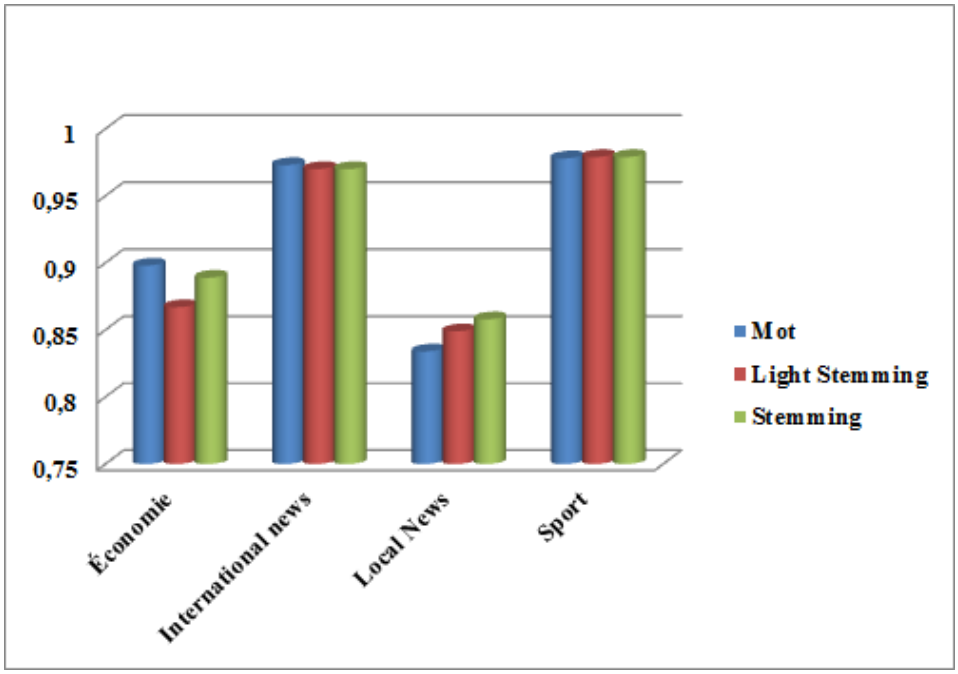

FIGURE 3.17: La Précision sur le corpus Khaleej

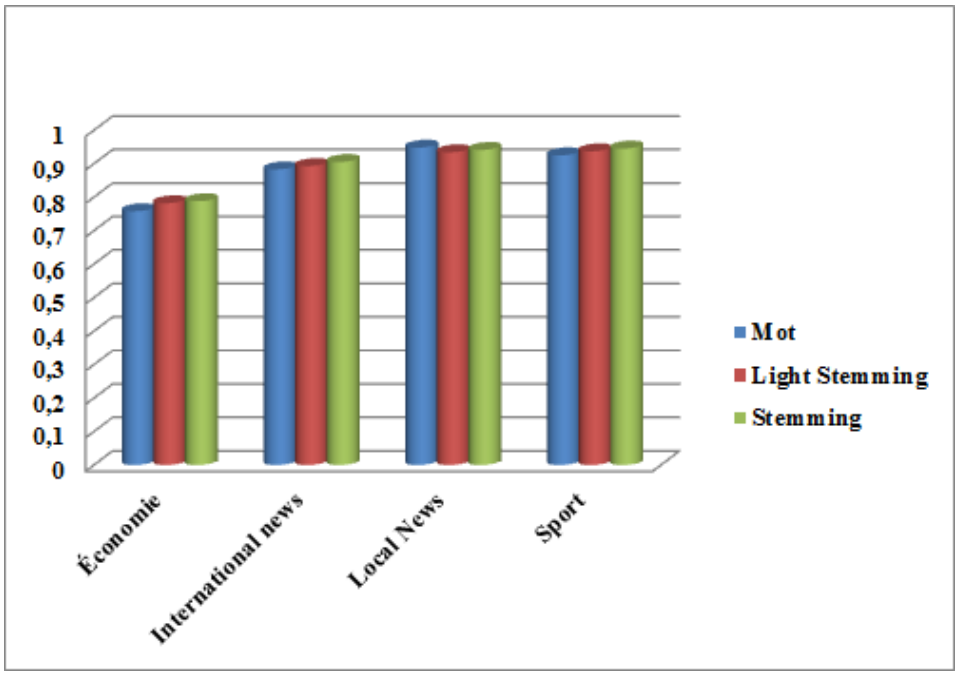

FIGURE 3.18: Le Rappel sur le corpus Khaleej

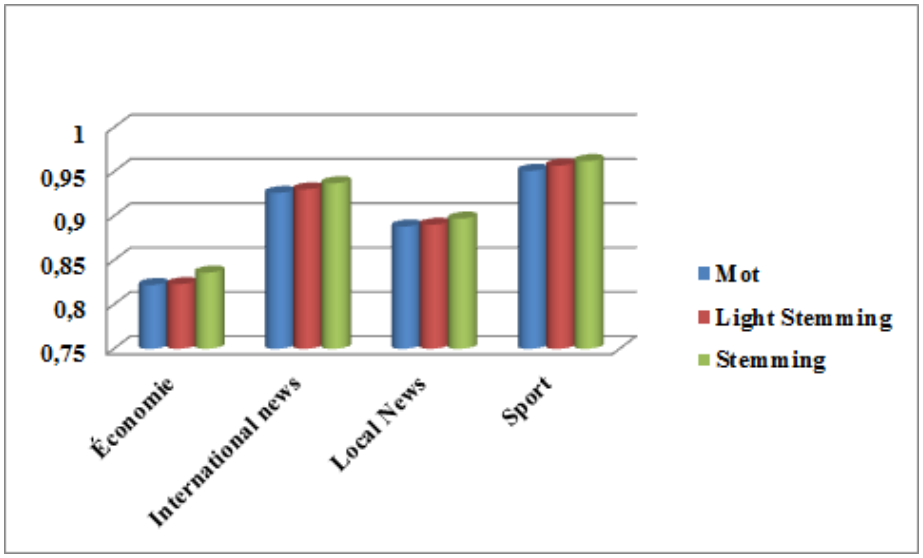

FIGURE 3.19: F-Mesure sur le corpus Khaleej

d) Évaluation du classifieur SVM sur les trois corpus

Sur les trois corpus les résultats de la précision sont meilleurs, ils sont égales à 0,821 sur le corpus Mosleh et 0,936 sur le corpus Watan et 0,913 sur le corpus khaleej voir la figure 3.20.
en comparant l'utilisation de Bag of Words, Light Stemming et le Stemming, on remarque une légère amélioration avec le Stemming. La figure 3.21 représente le rappel obtenu pour chaque corpus. Les résultats permettent de montrer que le rappel pour la corpus Mosleh égale à 0,726, pour le corpus Watan égale à 0,935 et pour le corpus Khaleej égale à 0,910 .

Une petite amélioration dans les résultats, on utilisant le Stemming, par rapport aux autres approches.
Les valeurs de F-Mesure sont égales à 0,754 sur le corpus Mosleh et 0,936 sur le corpus Watan et 0,910 sur le corpus Khaleej voir la figure 3.22.
Une légère différence entre les valeurs de F-mesure on appliquant le Light Stemming et le Stemming.

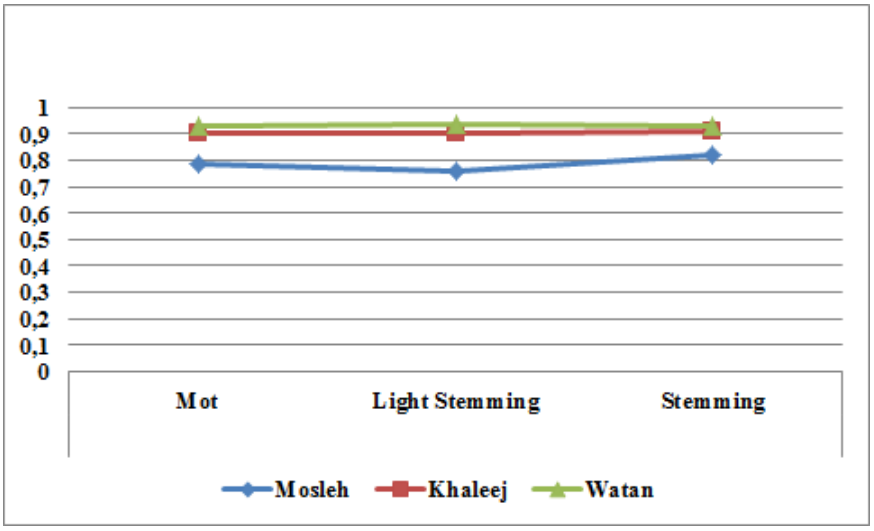

FIGURE 3.20: La Précision sur les trois corpus

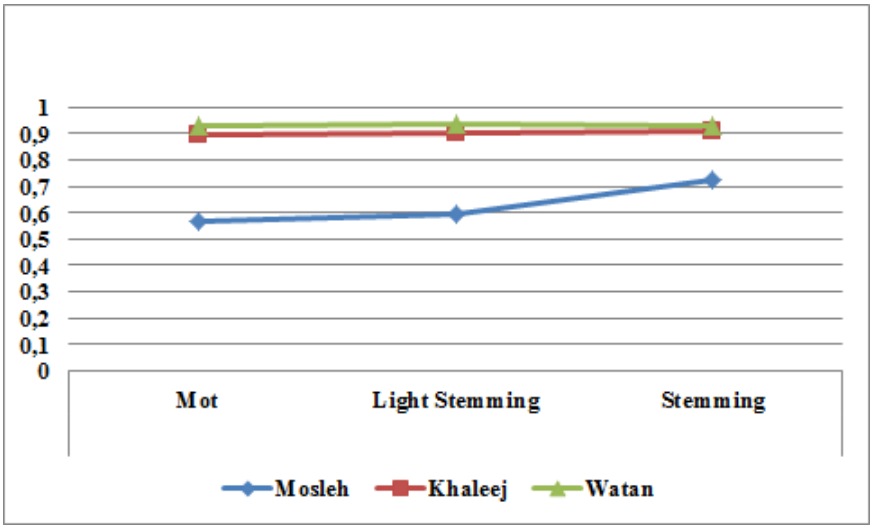

FIGURE 3.21: Le Rappel sur les trois corpus

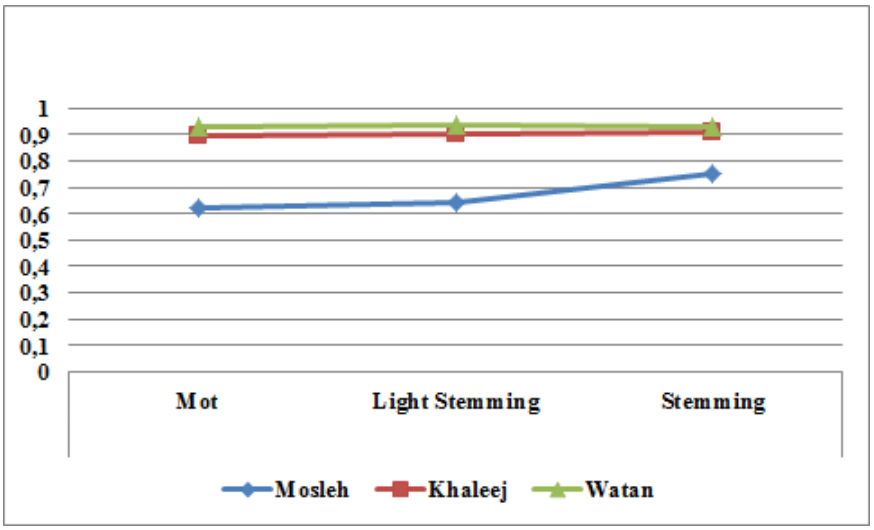

FIGURE 3.22: F-Mesure sur les trois corpus

Le but de nos expérimentations est de tester l'algorithme SVM pour la classification automatique des textes arabes avec différents corpus et différents descripteurs. Nous avons mené une étude comparative sur trois approches pour choisir un descripteur de textes Bag of Words, Light Stemming et Stemming en utilisant trois corpus différents.
D'après Les résultats obtenus, on peut conclure que :

- Les meilleurs résultats sont obtenus sur les corpus Watan et Khaleej qui sont de taille importante par rapport au corpus Mosleh. La précision est égale à 0,936, le rappel est égale à 0,935, et F-mesure est égale à 0,935 sur le corpus Watan avec le Stemming.
- utiliser le Stemming pour choisir le descripteur de textes a donné une légère amélioration, une différence de 0,09 par rapport aux autres représentations.
- utiliser des catégories avec un nombre de textes réduit cause de problème dans la classification, on a obtenu un faible rappel pour les catégories éducation, computer, ingénierie et droit sur le corpus Mosleh.

Ce fait confirmé par Jalam *"Dans la pratique, les effectifs des classes sont souvent déséquilibrés et, pour certaines classes, le nombre d'exemples positifs est faible comparé à celui des exemples négatifs. Ceci crée une difficulté supplémentaire car les classes peu nombreuses sont mal représentées"* [18].

Conclusion

La catégorisation des textes consiste à classer automatiquement un ensemble de documents selon des catégories prédéfinies. Cette approche est attractive du moment qu'elle décharge les organisations des tâches fastidieuses de classification manuelle des documents. Afin d'atteindre cet objectif un ensemble d'étapes est habituellement suivi.

Les chercheurs en catégorisation de textes, comme dans d'autres domaines expérimentaux, utilisent un ensemble de jeux de données qui aident à valider et à comparer les performances des classifieurs proposés. Dans notre travail on a utilisé trois corpus différents. Le premier corpus choisi est le corpus Mosleh largement utilisé dans les travaux de classification de textes arabes[30], [29], [4], [20]. Les deux derniers sont le corpus Watan et le corpus Khaleej construits par le Dr Mourad Abbas et de taille plus grande que le premier corpus.

Le choix de descripteurs est une autre étape très importante. Ces derniers constituent la structure de l'espace dans lequel seront représentés les textes. Ils doivent être le plus discriminant possible. Ces termes ne devraient pas être trop nombreux car ce sont eux qui vont déterminer la taille de l'espace vectoriel.

L'arabe est une langue morphologiquement très riche(sa nature agglutinante, sa richesse flexionnelle et l'absence de voyellation de la majorité des textes arabes écrits), qui présente de vrais défis à la classification automatique des textes. Pour cela nous avons traité dans ce mémoire la question de la classification automatique des textes écrits en caractères arabes en appliquant un traitement morphologique basé sur le Stemming.

Dans ce contexte nous avons choisi trois approches Bag of Words, le Light Stemming et le Stemming concernant le choix du descripteur à adopter pour la représentation vectorielle des textes. La premiere approche exclut toute analyse grammaticale des termes, les documents sont représentés par des vecteurs de dimension égale à la taille du vocabulaire, qui est en général assez grand. Par contre la deuxième approche Light Stemming est principalement basée sur la normalisation et l'élimination des affixes. Une liste étendue d'affixes les plus fréquents est utilisée pour extraire la racine. Ensuite, le Stemming, on a utilisé Khoja Stemming qui figure parmi les approches efficaces pour le Stemming du texte arabe [11]. L'approche consiste à enlever les affixes après une première étape de normalisation. Ensuite, le résultat est comparé avec une liste de modèles. Si une correspondance est trouvée, les lettres représentant la racine dans le modèle sont extraits. Ensuite, la racine ainsi extraite est validée dans la liste de racines connues.

On a commencé nos expérimentation par une phase de prétraitement morphologique et la suppression des mots vides qui consiste à nettoyer les textes pour améliorer les résultats. Cette démarche nous a permis de diminuer significativement la taille des corpus utilisés. Les résultats obtenus montrent un taux de réduction d'environ 30% sur les trois corpus. Ce qui permet de résoudre les difficultés concernant la dimension extrêmement élevée de l'espace d'apprentissage. En appliquant les deux approches Light Stemming et le Stemming pour l'extraction de la racine, on obtient de meilleurs taux de réduction de la taille des trois corpus.

La dernière étape dans notre expérimentation est l'évaluation du classifieur SVM sur les trois corpus. Et pour ce faire on a utilisé trois mesures la précision, le rappel et F-mesure. Les meilleurs résultats de la précision, le rappel et F-mesure sont dans la grande majorité des cas obtenus par l'utilisation de l'approche Stemming Sur les corpus Watan et Khaleej qui ont une taille importante par rapport au corpus Mosleh. La valeur de la précision est égale à 0,936, le rappel est égale à 0,935, et F-mesure est égale à 0,935 sur le corpus Watan avec le Stemming. Pour cela, nous concluons que l'approche basée sur le Stemming est la plus performante avec le classifieur SVM sur des corpus de taille importante.

Les catégories avec un nombre de textes réduit cause des problèmes dans la classification automatique de textes. On a obtenu un faible rappel pour les catégories éducation, computer, ingénierie et droit du corpus Mosleh. Le rappel est égale à 0,333 avec l'approche Bag of Words, est égale à 0,389 avec le Light Stemming et il est égale à 0,500 avec le Stemming sur la catégorie Droit.

Les perspectives

- La première perspective se présente dans le choix du descripteur pour représenter l'espace d'apprentissage. On veut tester l'impact de l'approche conceptuelle sur la classification automatique de textes arabes. Cette approche se base non pas sur les termes présents sur le texte à traiter mais sur les concepts correspondants. Ainsi, au lieu de définir un espace vectoriel dont chaque composante représente un terme(mot, stem, lemme, N-grammes), on projette l'ensemble de termes du texte sur un ensemble fini de concepts(liste de concepts). Cette liste peut être décrite dans un thésaurus, une ontologie, une hiérarchie de concepts...etc.
- La deuxième perspective consiste à améliorer la classification automatique de textes dans les catégories de taille réduite par l'exploration des méthodes qui prennent en charge cet aspect dans leur conception.

Bibliographie

[1] Ramzi Abbes. La conception et la réalisation d'un concordancier électronique pour l'arabe. *Thése Doctorat*, 2004.

[2] Bassam Al-Salemi and Mohd. Juzaiddin Ab Aziz. Statistical bayesian learning for automatic arabic text categorization. *Department of Computer Science, Faculty of Information Technology,University Kebangsaan Malaysia, Bangi, 43600, Selangor, Malaysia, Journal of Computer Science 7 (1) : 39-45, 2011ISSN 1549-3636*, 2011.

[3] Riyad Al-Shalabi, Ghassan Kanaan, and Manaf H.Gharaibeh. Arabic text categorization using knn algorithm. 2008.

[4] Riyad Al-Shalabi and Rasha Obeidat. Improving knn arabic text classification with n-grams based document indexing. *Faculty of Computers Information-Cairo University*, 2008.

[5] Saleh Alsaleem. Automated arabic text categorization using svm and nb. *Shaqra University, Saudi Arabia, International Arab Journal of e-Technology, Vol. 2, No. 2*, June 2011.

[6] Ion Androutsopoulos, John Koutsias, Konstantinos V. Chandrinos, George Paliouras, and Constantine D. Spyropoulos. An evaluation of naive bayesian anti-spam filtering. *Proceedings of the workshop on Machine Learning in the New Information Age, G. Potamias, V. Moustakis and M. van Someren (eds.), 11th European Conference on Machine Learning, Barcelona, Spain, pp. 9-17*, 2000.

[7] Sujeevan Aseervatham. Apprentissage à base de noyaux sémantiques pour le traitement de données textuelles. *thèse doctorat*, 2007.

[8] Laurent Audibert. Outils d'exploration de corpus et désambiguïsation lexicale automatique. *thèse doctorat*, 2003.

[9] Lamia Hadrich Belguith and Nouha Chaâben. Analyse et désambiguïsation morphologiques de textes arabes non voyellés. *Faculté des Sciences Économiques et de Gestion de Sfax – Laboratoire LARIS*, 2006.

[10] Siham Boulaknadel. Traitement automatique des langues et recherche d'information en langue arabe dans un domaine de spécialité :apport des connaissances morphologiques et syntaxiques pour l'indexation. *thèse doctorat*, 2008.

[11] Abderrezak Brahimi. Contribution à la recherche intelligente sur le web : Indexation sémantique des textes non-structurés. *Thése Doctorat*, 2013.

[12] Rehab Duwairi. Arabic text categorization. *the internationnal arab journal of information technology,vol4,No2*, April 2007.

[13] Alaa M. El-Halees. Arabic text classification using maximum entropy. *The Islamic University Journal (Series of Natural Studies and Engineering , Vol. 15, No.1, pp 157-167, ISSN 1726-6807, Department of Computer Science, The Islamic University of Gaza ,P.O. Box 108, Gaza, Palestine*, 2007.

[14] Yann Guermeur. Svm multiclasses,théorie et applications. *thèse doctorat*, 2007.

[15] Mena Badieh Habib. An intelligent system for automated text categorisation. *thèse Master*, 2008.

[16] Mohamadally Hasan and Fomani Boris. Svm : Machines à vecteurs de support ou séparateurs à vastes marges. *BD Web, ISTY3. Versailles : Université de Versailles St Quentin*, 16 janvier 2006.

[17] Majed Ismail Hussien, Minwer Al-dwan Fekry Olayah, and Ahlam Shamsan. Arabic text classification using smo,naive bayesian, j48 algorithms. *IJRRAS 9 (2)*, November 2011.

[18] Radwan Jalam. Apprentissage automatique et catégorisation de textes multilingues. *thèse doctorat*, 2003.

[19] Thorsten Joachims. Text categorization with support vector machines : Learning with many relevant features. *University of Dortmund LS8 Report 23*, 19 April 1998.

[20] Abidi Karima, Elberrichi Zakaria, and Tlili Guisssa Yamina. Arabic text categorization : A comparative study of different representation modes. *Journal of Theoretical and Applied Information Technology . Vol. 38 No.1*, 15 April 2011.

[21] Laurent Kevers. Accde sémantique aux bases de donées documentaires .techniques symboliques de traitement automatique du langage pour l'indexation thématique et l'extraction d'informations temporelles. *thèse doctorat*, 2010.

[22] Jamal Kharroubi. Etude de techniques de classement " machines à vecteurs supports "pour la vérification automatique du locuteur. *thèse doctorat*, 2002.

[23] Laila Khreisat. Arabic text classification using n-gram frequency statistics a comparative study. *thèse doctorat*, 2006.

[24] Mohamed El kourdi, Amine Bensaid, and Tajje eddine Rachidi. Automatic arabic document categorization based on the naïve bayes algorithm. *School of Science Engineering Alakhawayn University P.O. Box 104, Ifrane 53000, Morocco*, 2007.

[25] Man Lan. A new term weighting method for text categorization. *thèse doctorat*, 2006.

[26] Fabien Lauer. Machines à vecteurs de support et identification de systèmes hybrides. *thèse doctorat*, 2008.

[27] David D. Lewis. Naive bayesian text classification for spam filtering. *the Spring Conference of the American Statistical Association Chicago Chapter, Loyola University of Chicago*, 7 May 2004.

[28] Ludovic Mercier. Les machines à vecteurs support pour la classification en imagerie hyperspectrale :implémentation et mise en oeuvre. *Travail d'Etude et de Synthèse Technique en Informatique*, 11 février 2010.

[29] Abdelwadood Moh'd Meseleh. Chi square feature extraction based svms arabic language text categorization system. *Faculty of Information Systems and Technology, Arab Academy for Banking and Financial Sciences, Amman, Jordan*, 2007.

[30] Abdelwadood Moh'd Meseleh. Support vector machines based arabic language text classification system : Feature selection comparative study. *Faculty of Information Systems and Technology, Arab Academy for Banking and Financial Sciences, Amman, Jordan*, 2007.

[31] Slim Mesfar. Analyse morpho-syntaxique automatique et reconaissance des entites nommées en arabes standard. *thèse doctorat*, 2008.

[32] Didier Nakache. Extraction automatique des diagnostics à partir des comptes rendus médicaux textuels. *thèse doctorat*, 2007.

[33] Eric Ngouana and Serge Mayaya. Classifcation bayesienne naive de textes. *Faculté Polytechnique de Mons, 5ième Electricité, Certificat Applicatifs Multimédia*, 13 décembre 2005.

[34] Saeed Raheel. L'apprentissage artificiel pour la fouille de données multilingues : Application à la classification automatique des documents arabes. *thèse doctorat*, 2010.

[35] Mathieu Ramona. Classification automatique de flux radiophoniques par machines à vecteurs de support. *thèse doctorat*, 2010.

[36] Simon Réhel. Catégorisation automatique de textes et cooccurrence de mots provenant de documents non étiquetés. *thèse doctorat*, 2005.

[37] S.Al-Harbi, A.Almuhareb, A.Al-Thubaity, M.S.Khorsheed, and A.Al-Rajeh. Automatic arabic text classification. *JADT 2008 : 9es Journées internationales d'Analyse statistique des Données Textuelles*, 2008.

[38] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys, Vol. 34, No. 1,pp. 1–47.*, 2002.

[39] Fadi Thabtah, Wa'el Musa Hadi, and Gaith Al-shammare. Vsms with k-nearest neighbour to categorise arabic text data. *Proceedings of the World Congress on Engineering and Computer Science 2008 WCECS 2008, San Francisco, USA*, 22 - 24 October 2008.

[40] Romain Vinot. Classification automatiques de textes dans des catégories non thématiques. *thèse doctorat*, 2004.

[41] Romain Vinot, Natalia Grabar, and Mathieu Valette. Application d'algorithmes de classification automatique pour la détection des contenus racistes sur l'internet. *TALN 2003, Batz-sur-Mer*, 11–14 juin 2003.