REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

Université Djillali Liabes de Sidi Bel Abbes
Faculté de Technologie
Département d'Informatique

THESE

Pour l'obtention du diplôme de
DOCTORAT EN SCIENCES
Spécialité : Informatique
Option : Informatique

Par
Mr BENTAALLAH Mohamed Amine

Thème :

Utilisation des Ontologies dans la Catégorisation de Textes Multilingues

Devant le jury

Mr. BENSLIMANE Sidi Mohamed	Prof	UNIV-SBA	Président
Mr. MALKI Mimoun	Prof	UNIV-SBA	Directeur de thèse
Mr. ATMANI Baghdad	Prof	UNIV-ORAN1	Examinateur
Mr. FARAOUN Kamel Mohamed	Prof	UNIV-SBA	Examinateur
Mr. ABDERRAHIM Mohamed El Amine	M.C.A	UNIV-TLEMCEN	Examinateur
Mr. BOUCHIHA Djalloul	M.C.A	CU-NAAMA	Examinateur

Résumé

L'objectif de la catégorisation de textes multilingues est de permettre l'assignation d'une ou plusieurs catégories parmi une liste prédéfinie aux textes multilingues. La solution la plus directe consiste à traduire tout les documents vers une seule langue. Néanmoins, cette solution souffre de l'imprécision des techniques de traduction automatique. Les travaux de cette thèse s'inscrivent dans la problématique générale liée à l'utilisation des ontologies pour la catégorisation de textes multilingues. Dans notre première contribution, nous proposons une nouvelle approche de catégorisation multilingue intégrant les ontologies dans la phase de représentation comme moyen pour réduire les distorsions d'information causées par les traducteurs. L'idée consiste à mapper les traductions générées via les concepts de l'ontologie monolingue utilisée. Dans la deuxième contribution, nous proposons une extension de la première approche qui se base sur l'utilisation des ontologies multilingues afin d'éviter l'utilisation des techniques de traduction automatique. Dans nos expérimentations, nous utilisons le Princeton WordNet ainsi que le WordNet Espagnol pour évaluer les deux approches proposées sur deux corpus bilingues Anglais-Espagnol. Les résultats obtenus montrent une nette amélioration par rapport à l'approche basée sur la traduction automatique.

Mots Clés : Fouille de Textes, Catégorisation de Textes Multilingues, Ontologies, Traduction Automatique, Multilinguisme, Représentation Conceptuelle, WordNet.

Remerciement

Je remercie ALLAH, le tout puissant, le Clément et le Miséricordieux, de m'avoir donné la santé, le courage et le moral pour aboutir à ce modeste travail.

J'adresse mes plus sincères remerciements au Prof. Mimoun MALKI, mon directeur de thèse pour sa disponibilité, son aide et tout les conseils qu'il m'a apportés durant toutes ces années et surtout pour la confiance et les encouragements qu'ils m'a procurés pour achever ce travail.

Je suis redevable aux membres de mon jury : Prof. BENSLIMANE Sidi Mohamed qui a bien voulu présider la soutenance, Prof. ATMANI Baghdad, prof. FARAOUN Kamel Mohamed, Dr. ABDERRAHIM Mohamed El Amine et Dr. BOUCHIHA Djalloul pour leur participation en tant qu'examinateur.

Merci à ma famille qui m'a toujours encouragée et soutenue dans la poursuite de mes études.

A mes Parents,

Ma femme,

Mes enfants Mohamed Aymen et Imène

Ma soeur

Table des matières

Liste des tableaux

Table des figures

Chapitre 1

Introduction Générale

Face à la prolifération des documents accessibles sur le web, l'utilisateur est devenu incapable de traiter ces informations d'une façon manuelle et de sélectionner l'information pertinente dans cette gigantesque base documentaire. Cette incapacité a rendu indispensable, la construction de systèmes permettant d'automatiser le processus de recherche d'information. Le domaine de la fouille de textes est le domaine qui s'intéresse à la résolution de telles problèmes. Le domaine de la fouille de textes est apparu dans les années 90 comme une nouvelle discipline permettant l'extraction des connaissances à partir d'un ensemble de textes. Les trois principales tâches de la fouille de textes sont la classification de textes, la recherche d'information ainsi que l'extraction d'information [149]. La tache traitée dans cette thèse concerne la classification de textes et plus précisément la catégorisation de textes.

La catégorisation de textes est une classification de type supervisée qui consiste à utiliser un ensemble de documents pré-étiquetés pour pouvoir classer de nouveaux documents. Le besoin de catégoriser des textes remonte au début des années 60 mais ce n'est qu'aux années 90 que la catégorisation de textes est devenu un domaine à part entière vu sa grande sollicitation dans de nombreux applications nécessitant l'organisation de

documents tels que le filtrage de document, l'organisation de documents, l'indexation documentaire, etc [123].

La représentation de textes est l'une des étapes les plus importantes dans le processus de catégorisation de textes. La méthode de représentation la plus utilisée est la représentation dite "sac de mot" qui consiste à représenter le document sous forme d'un vecteur de mots[150]. L'inconvénient majeur de cette méthode de représentation réside dans le fait qu'elle ne prend pas en considération les relations entre les mots de la langue. De plus, la majorité des problèmes rencontrés dans le domaine de la catégorisation de textes sont celles posées par les langues[148] tels que :

- la graphie où un mot peut s'écrire de plusieurs façon (par exemple : Sheikh, Sheik, etc),
- la variation grammaticale où le mot peut avoir plusieurs catégories grammaticale. par exemple, "or" ne doit pas être considéré comme une conjonction de coordination dans la phrase "mine d'or".
- la variation morphologique où les marques de nombres, de genre et les conjugaisons peuvent changer la graphie d'un mot (par exemple : chevaux, cheval).
- le problème de composition lexicale où un mot peut être formé de plusieurs mots.

Durant cette dernière décennie, plusieurs travaux se sont focaliser sur l'utilisation des ontologies pour la proposition de nouvelles méthodes de représentation basées sur les sens de mots plutôt que sur les mots eux même [61]. Les travaux menés dans [163] ont montré l'utilité d'utilisation des ontologies pour améliorer les performances de l'indexation.

Le phénomène de multiliguisme s'est fait ressentir de plus en plus ces dernières années sur le web. Ceci est du aux raisons suivantes [75] :

- La disponibilité croissante de collections numériques qui a créé chez l'utilisateur de nouveaux besoins de retrouver l'information désirée quelque soit la langue dont l'information est rédigée.
- Le recul de la domination de l'Anglais comme langue du Web.

— L'apparition de plusieurs globalisation et de pays unifiés.

Ce phénomène de multilinguisme a donnée naissance à un sous domaine de la catégorisation de textes qui est bien la catégorisation de textes multilingues. La majorité des approches proposées pour la catégorisation multilingue se basent sur l'utilisation de la traduction comme moyen pour transformer la catégorisation multilingue en une catégorisation monolingue. Néanmoins, l'utilisation de la traduction génère une distorsion et une perte d'information qui influencera négativement sur le reste du processus de catégorisation. Ces dans ce contexte là que nos travaux se positionnent en proposant une contribution à la résolution de tel problème. Plus précisément, l'objectif de cette thèse est d'évaluer l'utilisation des ontologies comme moyen pour éliminer ou réduire les effets néfastes de la traduction. Tenant compte de l'objectif de cette thèse, nous avons proposés deux approches basées sur l'utilisation des ontologies :

1. La première proposition [11, 10]consiste à utiliser les ontologies comme moyen pour remédier aux insuffisances des techniques de traduction automatique. Cette combinaison entre ontologies et traducteurs automatiques offre les avantages suivants :
   — Sans utilisation des techniques de traduction automatique, il devient nécessaire d'incorporer une ontologie pour chaque langue, ce qui est extrêmement difficile dans le cas des langues les moins populaires.
   — L'utilisation d'une ontologie assez riche permettra de réduire les distorsions d'information causées par l'utilisation des techniques de traduction automatique.
2. La deuxième proposition [12] exclut l'utilisation des techniques de traduction automatique en incorporant une ontologie pour chaque langue. L'avantage réside dans l'absence de la distorsion causée par l'utilisation des techniques de traduction automatique. Tout de même, les résultats de cette proposition dépendront de la richesse des ontologies utilisées ainsi que de la qualité d'alignement entre les différentes ontologies utilisées.

Ce manuscrit est constitué de sept chapitres et une conclusion. Dans le deuxième chapitre, nous définissons la catégorisation de textes, nous ferons un rapide tour histo-

rique sur la catégorisation puis nous décrivons le processus générale de la catégorisation de textes avec tout ces étapes et nous montrons les problèmes spécifiques aux textes lors de l'apprentissage automatique. Enfin, nous terminons ce chapitre en présentant les jeux de données habituellement utilisés dans la littérature pour évaluer les systèmes de catégorisation de textes. Dans le troisième chapitre, nous allons discuter les méthodes proposées pour :

- le choix de termes qui vont servir à représenter les catégories avec leurs documents, c'est un choix primordial et important pour la catégorisation de textes.
- la pondération des termes qui va fournir aux algorithmes de classification un modèle qui sera utilisé pour mesurer les similarités. Le choix de la méthode de pondération influe sur la suite du processus de la catégorisation de textes.
- la réduction de dimensionnalité qui va servir à diminuer la taille du vocabulaire avant d'appliquer les techniques de classification les plus complexes.

Le quatrième chapitre présente les méthodes de classification les plus utilisées dans la littérature en insistant sur les caractéristiques, les avantages et les limites de chaque méthode, puis nous présentons comment évaluer ces méthodes dans la catégorisation de textes. Le cinquième chapitre est consacré à la notion de l'ontologie. Nous présentons en premier lieu sa définition dans la communauté *Ingénierie de Connaissance*, puis nous donnons une définition formelle de l'ontologie dans la communauté *Catégorisation de Textes*. Par la suite, nous détaillons les constituants de l'ontologie. Nous nous attaquons dans le sixième chapitre au problème de la catégorisation de textes multilingues. Nous définissons en premier lieu la catégorisation multilingue, puis nous citons les différents travaux effectués dans ce domaine. Le septième chapitre est dédié à la description des approches proposées dans le cadre de cette thèse et la présentation des résultats ainsi que leurs discussions.

Chapitre 2

Catégorisation de Textes

2.1 Introduction

Le volume d'information accessible électroniquement ne cesse de croitre continuellement. Un meilleur exemple de cette croissance est la base documentaire de Google qui est passé de 25 millions de pages durant ces premiers jours en novembre 1998 pour atteindre environ 30 trillions de pages en aout 2012 (voir http ://www.google.fr). Face à cette croissance extrêmement rapide ainsi que l'impossibilité de traiter manuellement cette gigantesque base documentaire. Il est devenu indispensable de concevoir des outils efficaces, permettant à l'utilisateur de rechercher les documents pertinents répondant à son besoin ou de filtrer les documents désirés à partir d'un flux de documents. Ces outils proviennent de deux domaines à savoir la *Recherche d'Information* et la *Catégorisation de Textes*.

La catégorisation de textes consiste donc à trouver, dans un flux de documents, ceux qui sont relatifs à un sujet défini par avance. En d'autre terme, il s'agit de trouver la catégorie thématique d'un document en le comparant avec les documents de

chaque catégorie. Étant donnée que les catégories sont déjà prédéfinies, la catégorisation de textes est un problème de classification supervisée nécessitant l'implication des techniques d'apprentissage automatique afin de construire un classifieur à partir d'un ensemble de documents pré-étiquetés dans chaque catégorie. Ce classifieur aura comme objectif l'assignation d'une ou plusieurs catégories à un document non-étiqueté.

la différence entre la *Recherche d'Information* et la *Catégorisation de Textes* réside dans le type de la base documentaire ainsi que son interrogation. En effet, dans la catégorisation de textes, la base documentaire est variable et son interrogation est fixe, alors que, pour la recherche d'information la base documentaire est fixe et son interrogation est dynamique. La catégorisation de textes est apparu dans les années 60, mais ce n'est qu'à partir des années 90 qu'elle est devenu un domaine de recherche très important et actif en raison de l'intérêt majeur de ces divers applications telles que l'indexation documentaire avec vocabulaire contrôlé, le filtrage de spams, la détection de genre de textes, la gestion des E-mails et bien d'autres applications [58].

Dans ce chapitre, nous allons nous intéresser au domaine de la catégorisation de textes en présentant en premier lieu une définition formelle puis nous décrivons le processus général de la catégorisation de textes et nous montrons les problèmes spécifiques aux textes lors de l'apprentissage automatique et nous terminons ce chapitre en présentant les jeux de données habituellement utilisés dans la littérature.

2.2 Définition de la Catégorisation de texte

La Catégorisation de Texte (C.T) est le processus qui consiste à affecter un document à une ou plusieurs catégories parmi une liste prédéfinie.

F. Sebastianni [44] définit formellement la catégorisation de texte comme étant le

processus qui consiste à associer une valeur booléenne à chaque paire (d_j, c_i) dans $D \times C$, où D est l'ensemble des textes et C est l'ensemble des catégories. La valeur V (Vrai) est 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. Plus formellement, il s'agit de construire une fonction $F : D \times C \rightarrow \{V, F\}$ qui associe une ou plusieurs catégories à un document d_j. L'objectif attendu de cette fonction et d'approximer le plus possible la fonction $E : D \times C \rightarrow \{V, F\}$, modélisant les assignations correctes entre les documents et les catégories. L'assignation des catégories aux différents documents doit respecter deux critères importants. D'une part, aucune connaissance supplémentaire sur les sens des catégories n'est exigée. Ainsi, on a à faire à des catégories symboliques. Par conséquent, les noms des catégories n'interviendront pas dans le processus d'assignation. D'autre part, l'assignation des catégories à un document doit être établie en se basant sur le contenu du document lui même et non sur les méta-données associées au documents (tels que date de publication,source de publication, type de document, etc.).

Selon T. Saracevic [151], l'assignation d'un document à une catégorie ne peut être décidée de manière déterministe à cause de la subjectivité de la sémantique d'un document. Le phénomène d'inter-indexeur d'incohérence présenté dans [21] montre qu'il peut y avoir fréquemment désaccord entre deux experts humains quant à l'assignation d'un document à une catégorie.

Une confusion existe entre la définition de la catégorisation et celle de la classification. D. Nakache [33] propose de définir la classification comme étant l'action d'organiser un ensemble en structures ordonnées ou hiérarchisées et la catégorisation comme étant l'action d'affecter des éléments, qui possèdent des caractéristiques communes, à des catégories pré-établies, sans relation d'ordre.

La catégorisation de textes peut être réalisé manuellement ou automatiquement à partir d'un ensemble de couples document-catégorie. le cas manuel est marqué par l'intervention des experts qui analysent le corpus afin de fournir des règles permet-

tant de déterminer automatiquement la catégorie associée à un document. Le système CONSTRUE [62], construit par le Carnegie Groupe pour l'agence de presse Reuter est un système qui classe automatiquement un flux de dépêches de presse dans une ou plusieurs catégories en utilisant des règles de classification de type "si–alors". La construction manuelle des règles par un expert du domaine génère les inconvénients suivants :

— Intervention de l'expert du domaine à chaque mise à jour de l'ensemble des catégories ;
— Impossibilité d'utiliser le classifieur pour d'autres domaines. Ainsi, il faut construire un nouvel ensemble de règles.

Dans le cas automatique, un processus inductif se charge de la création d'un classifieur pour une catégorie en observant les caractéristiques d'un ensemble de documents classés manuellement par un expert du domaine. Dans ce cas, l'expert du domaine n'a pas à spécifier les règles mais tout simplement à fournir l'ensemble de documents classifiés. Un autre avantage est la portabilité du classifieur à tout autres domaines car les règles seront déduite automatiquement par le classifieur.

2.3 Les paradigmes de la catégorisation de textes

Comme montré dans la figure 2.1, trois types de catégorisation peuvent être distinguées : la cas binaire, le cas multi-classes et le cas multi-label.

— Le cas binaire consiste à assigner le document à un seul ensemble parmi deux ensembles possibles.
— le cas multi-classes consiste à assigner le document à une seule classe parmi les m classes possibles.
— le cas multi-label consiste à assigner le document à plusieurs classes en même temps.

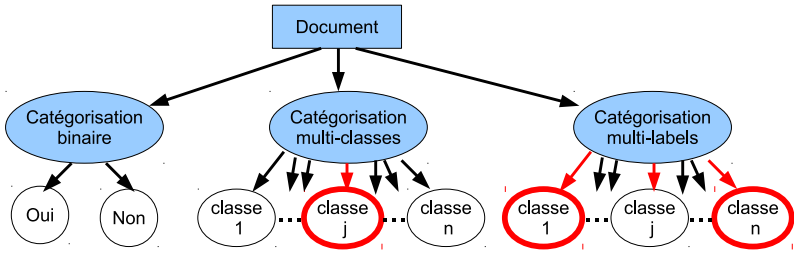


FIGURE 2.1 – *Paradigmes de la C.T*

Dans la catégorisation binaire, les deux ensembles possibles peuvent être référencés comme l'ensemble des documents appartenant (positive) et l'ensemble des documents non-appartenant (négative) respectivement (un contre tous), ou comme référençant deux classes disjointes (un contre un). Des algorithmes d'apprentissage tels que le Naive-Bayes, Régression Linéaire ou Support Vector Machine sont des exemples de ce paradigme. La catégorisation binaire est la base des deux autres paradigmes. En effet, l'approche traditionnelle consiste à appliquer une catégorisation binaire pour chaque classe en renvoyant une mesure reflétant le degré d'appartenance du document à la classe. Cette mesure sert à assigner par la suite le document à la classe ayant le plus fort degré dans le cas multi-classe ou aux classes ayant des degrés élevés dans le cas multi-label.

2.4 Les étapes de la catégorisation de textes

Comme montrée dans la figure 2.2, La construction d'un système de catégorisation de textes nécessite le passage par plusieurs étapes. Ces étapes varient d'un système à un autre selon les besoins. Néanmoins, tout système de catégorisation de textes peut être construit à partir de ces étapes :

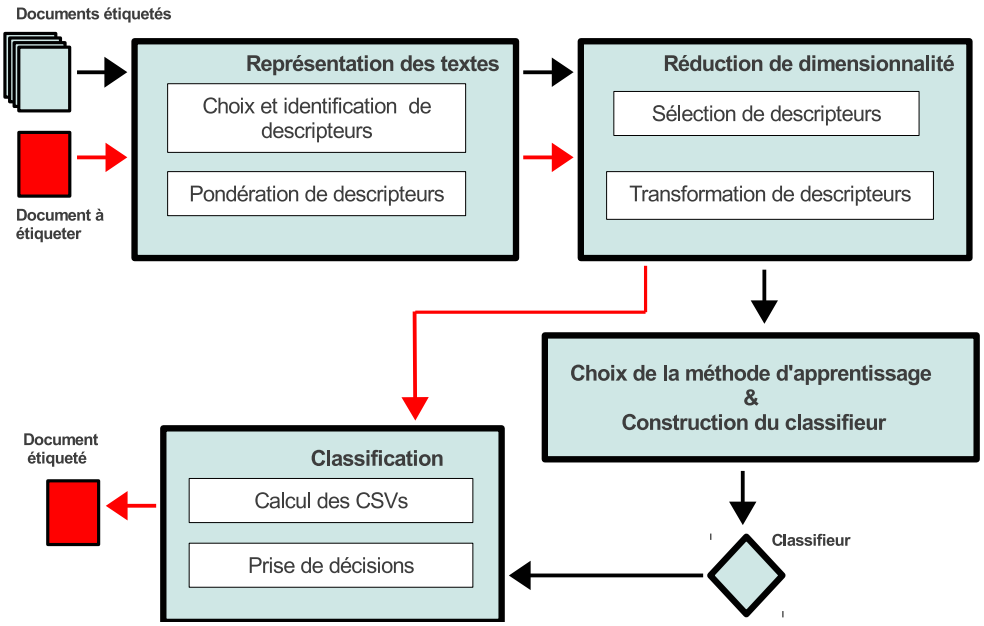


FIGURE 2.2 – *Les étapes de la C.T*

2.4.1 Représentation des textes

Comme les documents textuels sont des documents non structurés, un codage préalable du texte est nécessaire. Cette première étape consiste à représenter les documents textuels sous une forme exploitable par la machine. La forme la plus utilisée dans la C.T consiste à transformer l'ensemble des textes en un tableau croisé où :

- La $j^{ième}$ ligne correspond au document d_j.
- La $i^{ième}$ colonne correspond au descripteur t_i.
- Le croisement entre la $j^{ième}$ ligne et la $i^{ième}$ colonne représente le poids du descripteur t_i dans le document d_j.

La construction d'un tel tableau nécessite les étapes suivantes :

1. **Choix et identification des descripteurs :** Cette étape consiste à identifier la nature des descripteurs les plus représentatifs du document textuel. En effet, les descripteurs peuvent être des mots, des lemmes, des phrases, des concepts ou tout

autre entité représentant au mieux le document textuel. Salton et al. [133] et Aas [1] utilisent, à titre d'exemples, les mots comme descripteurs, tandis que d'autres préfèrent utiliser les lemmes (racines lexicales) [130] ; ou encore des stemmes (la suppression d'affixes)[97]. Cavnar [18] utilise les n-grams comme méthode pour la représentation des textes. Hotho [15] propose une approche hybride basée sur les stemmes des mots communs qui sont enrichis par les concepts extraits à partir d'une ontologie. De même, Peng [111] utilise les concepts pour représenter les documents.

2. **Pondération des descripteurs :** Une fois les descripteurs identifiés, cette étape consiste à calculer les poids des descripteurs dans les documents. Ces poids mesurent l'importance des descripteurs dans les documents. Plusieurs méthodes existent pour la pondération des descripteurs. Dans le troisième chapitre, nous détaillerons ces différentes méthodes en citant leurs avantages et inconvénients (voir section 3.3).

2.4.2 Réduction de dimensionnalité

La C.T ainsi que la R.I souffrent du problème de la grande dimensionnalité. En effet, pour un corpus de taille raisonnable, le tableau "$descripteurs \times textes$" peut avoir des centaines de milliers de colonnes (descripteurs) . Même si l'application des prétraitement tels que l'élimination des mots vides et lémmatisation réduisent le nombre de descripteurs, ce nombre reste tout de même un inconvénient majeur pour l'application des techniques d'apprentissage en C.T. Afin de résoudre ce problème, plusieurs méthodes ont été proposé pour réduire l'ensemble des descripteurs en les ordonnant selon leurs capacités à caractériser une catégorie ou un documents. Ces méthodes réduisent le nombre de descripteurs selon deux approches différentes :

1. **Sélection des termes :** Selon cette approche, la réduction est réalisée en éliminant un sous ensemble de descripteurs qui sont jugés inutiles pour la discri-

mination des catégories. Des méthodes tel que *mutual information*, *information gain*, *Chi square* ou *document frequency* (voir [47]) sont des exemples de cette approche.

2. **Transformation des termes :** Selon cette approche, la réduction est réalisée en remplaçant les descripteurs par de nouvelles entités. Ces entités regroupent sémantiquement les descripteurs. Ainsi, plusieurs descripteurs seront représentés par une seule entité. Des méthodes à savoir le *clustring suppervisé* proposé dans [23] et la *L.S.I (Latent Semantic Indexing)* [145, 46] utilisent cette approche.

Quelque soit l'approche utilisée pour la réduction, nous aboutissons à un tableau "$descripteurs \times textes$" avec un nombre limité de colonnes permettant ainsi l'utilisation des techniques d'apprentissage par la suite.

2.4.3 Choix de la méthode d'apprentissage et construction de classifieur

Cette étape est la plus importante dans le processus de catégorisation. Elle consiste à construire un classifieur autonome en utilisant les méthodes d'apprentissage supervisée. Parmi les méthodes d'apprentissage les plus souvent utilisées figurent l'analyse factorielle discriminante [86], la régression logistique [72], les réseaux de neurones [146], les plus proches voisins [167], les arbres de décision [4], les réseaux bayesiens [19], les machines à vecteurs supports [64] et, plus récemment, les méthodes dites de boosting [96]. Les classifieurs se différencient selon leurs modes de construction (les classifieurs sont-ils construits manuellement, ou bien automatiquement par induction à partir des données ?) et selon leurs caractéristiques (le modèle appris est-il compréhensible, ou bien s'agit-il d'une fonction numérique calculée à partir de données servant d'exemples ?). Plusieurs travaux de recherche ont ciblé la comparaison entre les classifieurs [70, 113, 91]. Les plus performants ont été généralement basés sur les réseaux bayésiens, les machines à vecteurs supports [24], ou les méthodes de boosting [43]. Dans le chapitre 4, nous

détaillerons ces différentes techniques de classification.

2.4.4 Classification

Cette étape consiste à assigner le document à catégoriser à une ou plusieurs catégories en se basant sur le classifieur construit dans l'étape précédente à partir des documents étiquetés. A cet effet, deux étapes sont nécessaires :

1. **Calcul des CSVs :** Afin de pouvoir décider de l'appartenance d'un document à une catégorie ou une autre, le CSV (Classification Status Value) doit être calculé entre le document à catégoriser et les différentes catégories. Le CSV permet de déterminer le degré d'appartenance du document à une catégorie.
2. **Prise de décision :** Cette étape consiste à utiliser les CSVs calculés afin de décider sur l'appartenance du documents au différentes catégories. Cette prise de décision peut être effectuée selon trois stratégies :
   — Stratégie RCut : Dans cette stratégie, pour chaque document d, les t classes ayant obtenues les CSVs les plus élevés pour d sont sélectionnées pour l'étiqueter . Le paramètre t est soit prédéfini, soit fixé par validation croisée.
   — Stratégie PCut : Dans cette stratégie, pour chaque catégorie c_i, les k_i documents obtenant les meilleurs CSVs seront étiquetés avec c_i, puis la procédure est répétée pour les autres classes. Il s'agit ici d'un tirage avec remise.
   — La stratégie SCut prévoit de traiter chaque catégorie séparément. Ainsi, pour chaque catégorie c_i, un seuil s_i est défini. Le document d est affecté à toutes les classes c_k vérifiant $CSV(d, c_k) >= s_k$. Les seuils s_i doivent être optimisés individuellement sur un ensemble de test par validation croisée.

2.5 Évaluation de la qualité des classifieurs

Afin de s'assurer que le classifieur construit est généralisable à d'autres textes. Il est nécessaire d'évaluer la performance du classifieur. Plusieurs mesures d'évaluation originaire de la RI sont utilisées dans la C.T. L'évaluation de la C.T est basée sur la comparaison entre les assignations effectuées par le systèmes avec celles effectuées par un expert. Ainsi, il s'agit de construire une table de contingence pour chaque catégorie. Comme illustré dans la figure 2.3, il s'agit de calculer les quatre valeurs suivantes :

Catégorie Ci		*Jugement Expert*	
		Oui	Non
Jugement classifieur	Oui	$\mathbf{VP_i}$	FP_i
	Non	FN_i	$\mathbf{VN_i}$

FIGURE 2.3 – *Table de contingence*

— VP_i **(vrai Positif) :** Le nombre de documents correctement assignés à la catégorie C_i.

— FP_i **(Faux Positif) :** Le nombre de documents incorrectement assignés à la catégorie C_i.

— FN_i **(Faux Négatif) :** Le nombre de documents incorrectement rejetés de la catégorie C_i.

— VN_i **(Vrai Négatif) :** Le nombre de document correctement rejetés de la catégorie C_i.

En se basant sur ces 4 valeurs, plusieurs mesures ont été proposé pour évaluer les performances d'une C.T.

2.5.1 Précision et rappel

Initialement conçu pour l'évaluation des systèmes de recherche d'information, la précision et le rappel sont les deux mesures les plus couramment utilisées pour évaluer la C.T. Ces deux mesures permettent de formuler deux notion importante pour décider de la performance d'un système de catégorisation à savoir le bruit et le silence (voir Figure 2.4). Pour chaque catégorie, on aura à faire à deux ensembles de documents.

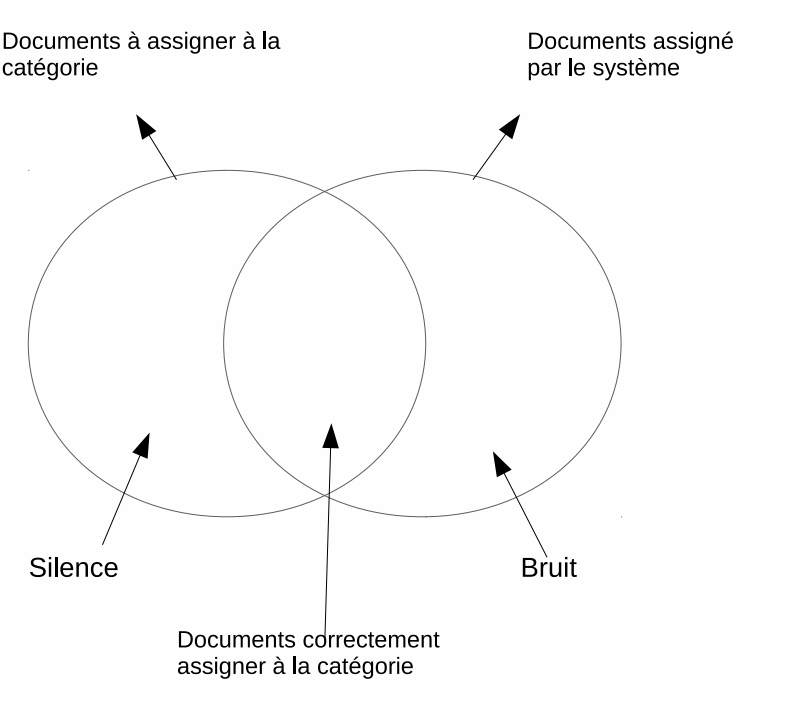


FIGURE 2.4 – *Relation entre bruit et silence*

Le premier ensemble contient les documents que doivent être assigner à la catégorie tandis que le deuxième ensemble contient les documents assignés par le système à la catégorie. Plus l'intersection entre ces deux ensembles est peuplée plus les performances sont meilleurs, ce qui revient à minimiser le bruit et le silence. Le rappel représente la proportion de documents correctement classés par rapport à tous les documents de la classe C_i.

$$R(c_i) = \frac{VP(c_i)}{VP(c_i) + FN(c_i)} \tag{2.1}$$

Le rappel mesure la capacité d'un système de classification à détecter les documents de chaque catégorie. La valeur de rappel est toujours comprise entre 0 et 1. Plus la valeur s'approche de 1 et plus le système catégorise bien les documents de la catégorie. La précision représente la proportion de documents correctement classés parmi ceux classés par le système comme appartenant à la catégorie C_i.

$$P(c_i) = \frac{VP(c_i)}{VP(c_i) + FP(c_i)} \tag{2.2}$$

La précision mesure la capacité d'un système à ne pas assigner à une catégorie les documents qui ne lui appartient pas. Elle peut être interprétée par la probabilité conditionnelle qu'un document choisi aléatoirement dans la classe soit bien classé par le système.

2.5.2 F-Mesure

Afin de mieux évaluer un système de C.T, il est indispensable de fusionner les deux mesures *précision* et *rappel* vu leur complémentarité. Plusieurs mesures ont été développées afin de fusionner les deux mesures. La mesure F décrite dans [125] est la mesure de synthèse communément adoptée depuis les années 80 pour évaluer la C.T à partir de la précision et du rappel. Elle permet de combiner, selon un paramètre β, rappel et précision et est calculé comme suit :

$$F_\beta(c_i) = \frac{(\beta^2 + 1) \times P(c_i) \times R(c_i)}{\beta^2 \times P(c_i) + R(c_i)} \tag{2.3}$$

Le paramètre β permet de choisir l'importance relative que l'on souhaite donner à chaque mesure. On choisit en général de donner la même importance aux deux mesures, donc habituellement, la valeur de β est fixée à 1 et la mesure est ainsi notée F_1 (noté F) qui s'écrit :

$$F_1(c_i) = \frac{2 \times P(c_i) \times R(c_i)}{P(c_i) + R(c_i)} \tag{2.4}$$

2.5.3 Micro-Moyenne et Macro-Moyenne

La micro-moyenne (micro-averaging) calcule les mesures *rappel* et *précision* de façon globale en sommant les valeurs VP, FP, FN et VN de chaque catégorie. A partir de la

		Expert	
	$C = \{c_1, c_2, ..., c_{\|C\|}\}$	C_i	$\neg C_i$
Classifieur	C_i	$VP = \sum_{i=1}^{\|C\|} VP_i$	$FP = \sum_{i=1}^{\|C\|} FP_i$
	$\neg C_i$	$FN = \sum_{i=1}^{\|C\|} FN_i$	$VN = \sum_{i=1}^{\|C\|} VN_i$

FIGURE 2.5 – *Table de contingence globale*

table de contingence globale (voir Figure 2.5), la *précision micro-moyenne* et le *rappel micro-moyenne* sont estimés comme suit :

$$P = \frac{\sum_{i=1}^{|C|} VP_i}{\sum_{i=1}^{|C|} VP_i + FP_i} \tag{2.5}$$

$$P = \frac{\sum_{i=1}^{|C|} VP_i}{\sum_{i=1}^{|C|} VP_i + FN_i} \tag{2.6}$$

La macro-moyenne (macro-averaging) se base sur une évaluation indépendante de chaque catégorie en calculant localement les précisions et rappels. L'évaluation globale est faite par la suite à travers le calcul de la moyenne des mesures locaux. La *précision* et le *rappel macro-moyenne* sont calculés comme suit :

$$P = \frac{\sum_{i=1}^{|C|} P_i}{|C|}, R = \frac{\sum_{i=1}^{|C|} R_i}{|C|} \tag{2.7}$$

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.

2.6 Domaines d'application de la C.T

La C.T est utilisée dans de nombreuses applications dont les données textuelles sont les seules ou les plus dominantes données à traiter. Ces applications partagent les points suivants :

— Le besoin d'organiser des documents dont le contenu textuel est l'unique , ou le plus dominant ou le plus simple contenu à traiter.

— Le besoin d'organiser de larges bases documentaires dont leur organisation manuelle devient quasiment impossible.

— Le fait que l'ensemble des catégories est fixé au préalable et dont la variation est qualifiée de rare.

2.6.1 Indexation automatique par les vocabulaires contrôlés

La C.T était utilisé depuis les années 70 dans l'indexation automatique des documents pour les systèmes de recherche d'information par le biais de vocabulaires contrôlés. Dans ces systèmes, chaque document est représenté par un ou plusieurs mots clés décrivant son contenu. Ces mots clés font partie d'un ensemble fini appelé "vocabulaire controlé" formant ainsi un thésaurus hiérarchique (comme le thésaurus NASA pour le domaine aérospatial ou le thésaurus MESH pour le domaine médical). Généralement, l'assignation des mots clés au différents documents est extrêmement couteuse en termes de temps. En prenant comme catégories les entrées du thésaurus, la C.T peut être utilisée à ce niveau pour assigner un document à une ou plusieurs entrées du thésaurus à savoir les mots clés. Ainsi, plusieurs travaux ont utilisé la C.T pour indexer automatiquement les documents ([152, 45, 46]). La C.T est utilisée aussi pour la génération automatique de méta-données. Dans les bibliothèques numériques, les documents sont marqués 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 par le biais des codes bibliographiques, des mots-clés ou des phrases-clés. La génération de ces méta-données peut être vu comme un problème d'indexation automatique par vocabulaire contrôlé et par conséquent peut être traitée par la C.T. Un meilleur exemple de système de génération automatique de méta-données par la C.T est celui du système Klarity[55]

2.6.2 Organisation de documents

La C.T est largement utilisée pour organiser et classer des documents que ce soit pour des organisations personnelles ou la structuration d'un document de base d'entreprise. Un meilleur exemple est celui des systèmes de catégorisation d'annonces des journaux. En effet, plus le nombre d'annonces s'accroit plus il devient nécessaire d'avoir un système permettant de classer les annonces dans leurs catégories les plus appropriées. Un exemple d'application concret est présenté dans [84] qui consiste à organiser cinq millions brevets de l'USPTO (U.S. Patent and Trademark Office)en catégories afin de faciliter leur recherche.

2.6.3 Filtrage de document

Le filtrage des textes est un autre exemple d'application où la C.T est utilisée pour classifier un flux de documents expédiés de manière asynchrone par un producteur d'information à destination d'un consommateur d'information. Ces systèmes de filtrage permettront par exemple d'empêcher la livraison d'un document n'intéressant pas un journal ayant une catégorie thématique bien précise (sport par exemple). ce filtrage peut être considéré comme un problème de C.T binaire avec deux catégories disjointes à savoir la catégories des documents pertinents et celle des documents non pertinents. Plusieurs travaux de recherche ont utilisé la C.T pour filtrage des spams et le classement

des messages dans des catégories thématiques pour l'utilisateur [22, 3]. Un système de filtrage peut être installé soit chez le producteur soit chez le consommateur. Dans le premier cas, il s'agit de livrer les document aux consommateurs intéressés en construisant un profil caractérisant chaque consommateur[94]. Par contre, dans le deuxième cas, il s'agit de construire un seul profil pour bloquer la livraison de documents jugés sans intérêt pour le consommateur.

2.6.4 Désambiguïsation sémantique

La Désambiguïsation sémantique (WSD : Word Sense Disambiguation) consiste à sélectionner le sens adéquat d'un mot dans un texte parmi l'ensemble de sens que peut avoir le mot dans la langue. La WSD est très importante pour de nombreuses applications, y compris le traitement du langage naturel et l'indexation des documents par le sens des mots. La WSD peut être considérée comme une tâche de T.C [48, 65] si nous considérons le contexte d'occurrence des mots comme un document et le sens du mot comme une catégorie. La WSD est juste un exemple du problème plus général consistant à lever les ambiguïtés du langage naturel, un des problèmes les plus importants en linguistique computationnelle.

2.7 Problèmes de la catégorisation de textes

Comme n'importe quel domaine traitant des documents textuels, la C.T doit faire face au problèmes liés au langage naturel.

Un des problèmes majeurs du langage naturel est celui de l'ambiguïté avec ces différents niveaux. En effet, l'ambiguïté peut être lexique dans le cas des mots ayant la même forme mais des sens différents comme elle peut être syntaxique suite à une dé-

duction erronée de l'agencement des mots dans la phrase. L'ambiguïté peut être aussi de niveau sémantique où un mot peut avoir plusieurs sens possible selon son contexte d'utilisation. Il existe aussi de multiples manières d'exprimer la même réalité, avec des nuances diverses. Deux mots ou expressions seront dits synonymes s'ils ont le même sens ; exemples : *mon chat mange un oiseau*, *mon gros matou croque un piaf* et *mon félin préféré dévore une petite bête à plumes* [88]. On voit bien qu'il s'agit d'un chat qui mange un oiseau mais pourtant les trois textes ne partagent aucun mot autre que des mots-outils (mon, un). Hormis les problèmes liés au langage naturel, la C.T dépend aussi de la taille des corpus manipulés. Ainsi, pour un corpus de taille raisonnable, on se retrouve face à une gigantesque matrice $documents * descripteurs$. Cette grande dimensionnalité peut réduire l'efficacité des algorithmes d'apprentissage sophistiqués dont leurs complexité dépend du nombre de descripteurs à traiter et peut aussi augmenter le risque de sur-apprentissage. En effet, on se retrouve avec des descripteurs dont la majorité se répètent rarement dans la collection. par conséquent, le classifieur aura tendance à classe correctement les exemples d'apprentissage, mais classe mal de nouveaux exemples. La répartition des documents du corpus traité sur les différents catégories est un facteur très important pour garantir une bonne catégorisation. En effet, il est nécessaire d'avoir un certain équilibre entre les différents catégories en termes du nombre de documents affectés. Généralement, les performances de catégorisation se dégrade pour les catégories moins peuplées que les autres catégories car elles seront mal représentées par rapport aux autres catégories.

2.8 Jeu de données utilisé pour l'évaluation

La comparaison entre différents systèmes de C.T ne peut être fiable qu'en utilisant les mêmes corpus. A cet effet, plusieurs corpus ont été proposé comme corpus standards pour la C.T. Dans ce qui suit, nous allons présenter les corpus les plus utilisés :

2.8.1 Le corpus Reuters

C'est un corpus en langue anglaise contenant la classification de dépêches de l'agence de presse Reuters. Une première version nommée Reuters-22173 a été proposée en 1987. Depuis, plusieurs versions ont été proposées. Ces versions se différencient entre elles par les nombres de textes des ensembles d'apprentissage et de test, ainsi que par le nombre des catégories à apprendre. Le tableau 2.1 montre les cinq versions proposées, avec les statistiques concernant chacune d'elles.

Version	Préparée par	$\sharp Catégorie$	$\sharp Ens.Apprent$	$\sharp Test$	%Doc étiquetés
Vers.1	CGI [63]	182	21450	723	80%
Vers.2	Lewis [91]	113	14704	6746	42%
Vers.2.2	Yang [166]	113	7789	3309	100%
Vers.3	Apte [5]	93	7789	3309	100%
Vers.4	PARC [162]	93	9610	3662	100%

TABLE 2.1: *Différentes versions de la collection Reuters*

Ce corpus souffre d'une mal définition de ces catégories, et d'un grand déséquilibre dans la répartition des documents entre ces catégories. En effet, il existe des catégories qui sont favorisées par rapport aux autres en termes de nombre des documents présents dans les jeux de teste et d'apprentissage. En effet, seules les 20 premières catégories contiennent plus de 100 textes. Malgré la quasi-unanimité des chercheurs à reconnaître sa mauvaise qualité, les chercheurs continuent à utiliser ce corpus dans leurs expérimentations [136, 167, 77, 37]. Le tableau 2.2 montre la répartition des textes entre les ensembles d'apprentissages et de testes pour les 21 catégories les plus représentées dans le corpus Reuters.

2.8.2 Le corpus 20Newsgroups

C'est un corpus créé par Lang [83], contenant 20000 messages équirépartis dans vingt classes hiérarchisées. Le tableau 2.3 présente les différentes classes de ce corpus ainsi

Catégorie	$\sharp Apprentissage$	$\sharp Test$
Earn	2877	1087
Acquisition	1650	719
Money-fx	538	179
Grain	433	149
Crude	389	189
Trade	369	118
Interest	347	131
Wheat	212	71
Ship	197	89
Corn	182	56
Money-supply	140	34
Dlr	131	44
Sugar	126	36
Oilseed	124	47
Coffee	111	28
Gnp	101	35
Gold	94	30
Veg-oil	87	37
Soybean	78	33
Nat-gas	75	30
Bop	75	30

TABLE 2.2: *Répartition des documents sur les 21 catégories les plus représentées dans le corpus Reuters*

que la répartition des documents entre le corpus de test et le corpus d'apprentissage.

2.8.3 OHSUMED

C'est un corpus créé par Hersh [67], il contient prés de 350000 articles extrait du corpus MEDLINE. Ces articles sont répartis en plusieurs classes correspondant aux catégories du thésaurus MeSH (Medical Subject Headings).

Catégorie	$\sharp Apprentissage$	$\sharp Test$
alt.atheism	480	319
comp.graphics	584	389
comp.os.ms-windows.misc	572	394
comp.sys.ibm.pc.hardware	590	392
comp.sys.mac.hardware	578	385
comp.windows.x	593	392
misc.forsale	585	390
rec.autos	594	395
rec.motorcycles	598	398
rec.sport.baseball	597	397
rec.sport.hockey	600	399
sci.crypt	595	396
sci.electronics	591	393
sci.med	594	396
sci.space	593	394
soc.religion.christian	598	398
talk.politics.guns	545	364
talk.politics.mideast	564	376
talk.politics.misc	465	310
talk.religion.misc	377	251

TABLE 2.3: *Répartition des documents dans les catégories du corpus 20Newsgroups*

2.8.4 Le corpus WebKB

C'est un corpus créé par Graven et al[25], il contient des pages web extraites de quatre universités américaines importantes. Les catégories correspondent au type des pages : étudiant, université, personnel, cours, projet, département et autre.

2.8.5 Le corpus Yahoo Science

C'est un corpus qui a été proposé pour la première fois par McCallum [102], il contient les pages d'accueil de sites indexés dans l'annuaire Yahoo sous la rubrique Science. Les classes correspondent aux sous-catégories de sciences dans cet annuaire.

2.9 Conclusion

Dans ce chapitre, nous avons présenté l'ensemble des objectifs de la C.T ainsi que ses notions de base. Nous avons défini en premier lieu la notion de catégorisation de textes. Nous avons présenter par la suite le processus de C.T avec ces étapes : la représentation de textes, le choix de classifieurs et l'évaluation du modèle. Il étais aussi important de citer quelque domaines d'application de la C.T ainsi que les difficultés rencontrés dans ce domaines. A la fin, nous avons citer les différentes méthodes utilisées pour évaluer un système de C.T ainsi que les corpus standards les plus utilisés pour une telle évaluation.

La C.T s'est établie au cours de la dernière décennie comme un domaine majeur de recherche pour les technologies de l'information. Ce dynamisme est en partie dù à la demande importante des utilisateurs pour cette technologie. Elle devient de plus en plus indispensable dans de nombreuses situations où la quantité de documents textuels électroniques rend impossible tout traitement manuel.

Il reste néanmoins difficile de fournir des valeurs chiffrées sur les performances qu'un système de C.T peut actuellement atteindre. La tâche est souvent subjective. De même, la comparaisons entre les différents systèmes de C.T reste extrêmement difficile vu la variété des corpus d'évaluation ainsi que les mesures utilisées. Malgré l'apparition de certains standard d'évaluation, il reste toujours difficile de réunir tous les critères permettant une réelle évaluation.

Chapitre 3

Représentation des textes

3.1 Introduction

Techniquement, un texte est un ensemble de symboles enregistrés sous un format numérique. certain de ces formats sont destinés uniquement pour la distribution des documents tels que le format PDF(Portable Document format) ou le format PS(postScript Format), tandis que d'autres sont déterminés par des environnements ayant des règles bien définies tels que le format DOC de Microsoft Word,le format SXW d'Open Office ou le format LATEX. L'objectif est d'unifier la représentation des documents textuels quelque soit leurs formats en les transformant en un ensemble de termes (descripteurs) qui peuvent être utilisés facilement par les algorithmes d'apprentissage.

Afin de pouvoir utiliser les algorithmes d'apprentissage issus de la communauté d'apprentissage automatique, il est nécessaire d'affecter un poids à chaque descripteur dans chaque texte. Cette pondération et d'une extrême importance car elle influencera sur les résultats des algorithmes d'apprentissage. Ainsi, un deuxième objectif consiste à mieux pondérer les descripteurs.

Le nombre de descripteurs est un facteur très important dont dépend les performances d'une C.T. En effet, plusieurs algorithme d'apprentissage sont incapable de traiter un nombre élevé de descripteurs. A cet effet, il est nécessaire de réduire la dimensionnalité de l'espace de représentation en ne gardant que les meilleurs descripteurs. Les méthodes de réduction sont utilisées pour sélectionner ou extraire les descripteurs les plus importants. Dans ce chapitre, nous allons présenté les différentes méthodes de représentation de textes en précisant leurs avantages et inconvénients puis nous citons les techniques de pondération les plus utilisées dans la C.T ainsi que les techniques utilisées pour la réduction de la dimensionnalité.

3.2 Choix de descripteurs

Un descripteur peut être défini comme étant n'importe quel élément pouvant être utilisé comme attribut dans les algorithmes de classification.Vu que le choix de descripteurs est la première étape du processus de catégorisation, il est essentiellement important de choisir quels descripteurs utiliser pour représenter les documents. En effet, ce choix aura un impact majeur sur le reste du processus.

3.2.1 Représentation sac de mots

La représentation "sac de mots" et sans aucun doute la représentation la plus simple et la plus intuitive. Elle consiste à représenter chaque document par un vecteur dont chaque composante correspond au nombre d'occurrence d'un mot dans le document. Elle a été utilisé dans plusieurs travaux pour représenter les documents textuels comme [90, 85]. Les mots ont l'avantage de posséder un sens explicite. Néanmoins, l'implémentation de cette représentation soulève plusieurs difficultés. La première difficulté est celle de la délimitation des mots dans un textes. En effet, on n'arrive toujours pas à définir

ce qu'est un mot. R.Gilly considère dans [51] un mot comme étant une séquence de caractères appartenant à un dictionnaire, ou formellement, comme étant une suite de caractères séparés par des espaces ou des caractères de ponctuations. Cette définition n'est pas valable pour toute les langues. En effet, des langues tels que le Chinois ou le Japonais ne séparent pas leurs mots par des espaces. Ajoutons à cela que certains séparateurs peuvent faire partie de certains mots(par exemple : aujourd'hui, 127.0.0.1, pomme de terre, etc). Une autre difficulté concerne la gestion des mots composés (par exemple : Arc-en-ciel, pomme de terre, etc) et des sigles (comme : IBM, CAF,CAN, etc). La prise en considération de ces cas nécessite des traitements linguistiques assez complexes. Cette représentation des textes exclut toute analyse grammaticale et toute notion d'ordre entre les mots et par conséquent, des textes sémantiquement éloignés peuvent avoir la même représentation. Par exemple, les textes *le loup mange la chèvre* et *la chèvre mange le loup* se voit attribué la même représentation malgré qu'il sont sémantiquement différent.

3.2.2 Représentation par phrases

Étant donnée que la représentation "sac de mots" exclut toute notion d'ordre et de relation entre les mots d'un texte, plusieurs recherches ont tenté d'utiliser les phrases comme descripteurs à la place des mots [46, 152]. L'utilisation des phrases permet de solutionner le problème d'ambiguïté généré par l'utilisation de la représentation "sac de mots". Par exemple, le mot *souris* a plusieurs sens possibles tandis que *souris optique* et *souris domestique* ne présente aucune ambiguïté. Même si les phrases ont l'avantage de mieux conserver la sémantique par rapport aux mots, leurs utilisation comme descripteurs n'a pas aboutit aux résultats espérés. Selon Lewis[93], cette représentation est pénalisée par le grand nombre de combinaisons possibles qui entraîne des fréquences faibles et trop aléatoires. Une solution proposée dans [17] consistait à considérer une phrase comme étant un ensemble de mots contigus (mais pas nécessairement ordonnés) qui apparaissent ensembles mais qui ne respectent pas forcement les règles grammati-

cales.

3.2.3 Représentation par lemmes ou racines lexicales

La représentation par lemmes ou racines lexicales est une extension de la représentation "sac de mots" qui consiste à remplacer chaque mot par sa forme canonique. Ainsi, il s'agit de regrouper les différentes formes que peut avoir un mot (singulier, pluriel, masculin, féminin, présent, passé, future,etc.) en une seule forme appelée *forme canonique*. Le regroupement des différents formes d'un mot offre les deux avantages suivants :

- La réduction de la dimensionnalité de l'espace de représentation. En effet, dans la représentation "sac de mot", chaque forme d'un mot se voit attribué une dimension ; tandis qu'avec la représentation par lemme les différentes formes seront fusionnées en une seule dimension. Par exemple, des mots tels que *jouer, joueur, joueurs, jouable, jouerait, joue, joues,etc* seront remplacés par un seul descripteur à savoir la racine *jou* ou le lemme *jouer*
- L'augmentation des occurrences des descripteurs. En effet, il est préférable de considérer un seul descripteur *jouer* ayant sept occurrences que de considérer les sept descripteurs *jouer, joueur, joueurs, jouable, jouerait, joue, joues,etc* avec une occurrence de chacun.

La lemmatisation et la racinisation sont les deux techniques utilisées pour trouver la forme canonique d'un mot. La lemmatisation utilise une base de connaissance contenant les différentes formes fléchies correspondant aux différents lemmes possibles. Ainsi, les formes fléchies d'un nom seront remplacées par la forme singulier masculin tandis que les différents formes fléchies d'un verbe seront remplacées par la forme infinitif. La lemmatisation nécessite l'utilisation d'un dictionnaire de formes fléchies de la langue ainsi qu'un étiqueteur grammatical. Un algorithme efficace, nommé TreeTagger [137], a été développé pour treize langues différentes : l'allemand, l'anglais, le français, l'italien, le néerlandais, l'espagnol, le bulgare, le russe, le grec, le portugais, le chinois, le Swahili et le vieux français. La racinisation utilise une base de connaissances des règles syntaxiques

et grammaticales de la langues pour transformer les mots en leurs racines. L'un des algorithmes de racinisation les plus connus pour la langue anglaise est l'algorithme de Porter [115]. La lemmatisation est plus compliquée à mettre en œuvre puisqu'elle dépend des étiqueteurs grammaticaux. De plus, elle est plus sensible aux fautes d'orthographe que la racinisation.

3.2.4 Représentation par N-grammes

Contrairement aux méthodes de représentation précédemment décrites, la représentation par N-grammes exclu toutes notions relatives au langues quant à la représentation d'un document textuel. Elle consiste à effectuer un découpage du texte à représenter en plusieurs séquences de n caractères consécutifs. Ainsi, il s'agit de se déplacer sur le texte par étape d'un caractère et de sélectionner les n caractères à chaque déplacement. Par exemple, le découpage du texte *"la recherche d'information"* donnera comme résultats les 3-grammes suivant : *la-, a-r,-re,rec,ech,che,her,erc,rch,che,he-,e-d,-d',d'i,'in,inf,nfo,for,orm,rma,mat,ati,tio,ion*. La notion de n-grammes a été introduite par Shannon [139] en 1948 ; il s'intéressait à la prédiction d'apparition de certains caractères en fonction des autres caractères. Depuis cette date, les n-grammes sont utilisés dans plusieurs domaines comme l'identification de la parole, la recherche documentaire, etc. L'utilisation des N-grammes offre les avantages suivants [75] :

- Indépendance vis-à-vis la langue du document.
- Ne nécessite aucune segmentation préalable du document.
- Moins sensible aux fautes d'orthographes.
- Permet d'identifier la langue d'un document.

3.2.5 Représentation par concepts

Toutes les méthodes décrites auparavant sont des méthodes statistiques basées sur le nombre d'occurrences des descripteurs en excluant toute notion de relation sémantique entre ces différents descripteurs. Cette notion de sémantique étant négligée dans les méthodes statistiques, l'ambiguïté demeure un handicap majeur. En prenant en considération l'exemple de la figure 3.1 illustrant quatre documents du corpus Reuters-21578 appartenant à la même catégorie "corporate acquisition" ne partagent aucun mot commun (a part les mots vides), les méthodes statistiques considéreront les quatre documents totalement indépendants. Les concepts définis comme unités de connaissance, peuvent être utilisés comme descripteurs dans le but de résoudre le problème d'ambiguïté ainsi que le problème de synonymie. En effet, chaque concept représente un sens unique qui peut être exprimé par plusieurs mots synonymes. De même, un mot ayant plusieurs sens se retrouve mappé dans plusieurs concepts. La représentation conceptuelle améliore la représentation d'un document par le fait qu'elle permet de l'enrichir par des descripteurs sémantiquement proches au descripteurs d'origine. Ainsi, un document contenant le mot *véhicule* peut être indexé par d'autres mots tels que *voiture* ou *automobile*. Le passage d'une représentation par mot vers une représentation par concept nécessite l'utilisation de ressources sémantiques externes au contenu des documents tels que : les réseaux sémantiques, les thésaurus et les ontologies. De ce fait, les performances d'une telle représentation dépendent crucialement de la richesse sémantique des ressources utilisées en terme de nombre de concepts et de relations entre ces concepts.

3.3 Pondération des descripteurs

Une fois l'ensemble des descripteurs représentant nos documents textuels est fixé, cette étape consiste à pondérer les descripteurs en fonction de leur importance relative

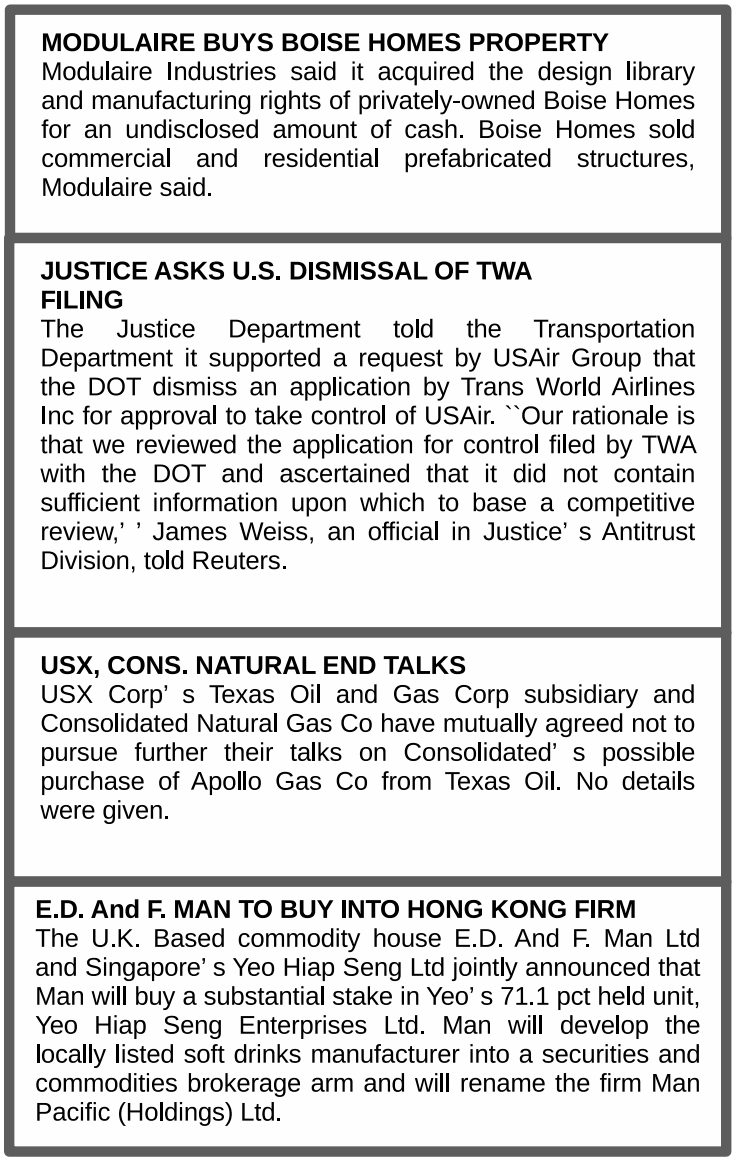


FIGURE 3.1 – *Exemple de 4 documents appartenant à la même catégorie corporate acquisitions ne partagent aucun mot commun[78]*

dans les documents. Une bonne pondération d'un descripteur dans un document doit prendre en considération l'aspect local,l'aspect global ainsi que l'aspect de normalisation. Généralement la pondération d'un descripteur i dans un document j implique le calcul de : $w_{ij} = L_{ij}G_iN_i$ où :

— L_{ij} est la pondération locale du descripteur i dans le document j. Cette pondération est généralement basée sur le nombre d'occurences du descripteur dans le

document.

— G_i est la pondération globale du descripteur i dans la collection des documents. Cette pondération défavorise les descripteurs les plus communs de la collection.

— N_j est le facteur de normalisation des poids du document j. Ce facteur a pour but d'éliminer l'influence de la taille des documents.

En fonction des choix sur ces trois aspects, plusieurs techniques ont été utilisé pour pondérer les descripteurs.

3.3.1 TF (Term Frequency)

Cette mesure est proportionnelle à la fréquence du terme dans le document (pondération locale). Ainsi, plus le terme est fréquent dans le document plus il est important. Elle peut être utilisée telle quelle ou selon plusieurs déclinaisons [126, 141].

$$tf_{ij} = f(t_i, d_j) \tag{3.1}$$

$$tf_{ij} = 1 + log(f(t_i, d_j)) \tag{3.2}$$

$$tf_{ij} = 0.5 + 0.5\frac{f(t_i, d_j)}{max_{t_i \in d_j} f(t_i, d_j)} \tag{3.3}$$

avec $f(t_i, d_j)$ la fréquence du terme t_i dans le document d_j

3.3.2 IDF (Invers Document Frequency)

Cette pondération mesure l'importance d'un terme dans toute la collection (pondération globale). Un terme qui apparaît souvent dans la base documentaire ne doit pas avoir le même impact qu'un terme moins fréquent. En effet, les termes qui apparaissent dans la majorité des documents n'ont pas de pouvoir discriminant pour distinguer les documents les uns des autres et doivent avoir par conséquence des pondérations faibles. La pondération IDF est donc inversement proportionnelle au nombre de documents contenant le terme à pondérer. Ainsi, plus le terme apparait dans plusieurs document moins il est discriminant et se voit attribuer une pondération faible. La pondération IDF est généralement exprimé comme suit :

$$idf(t_i) = log(\frac{N}{df(t_i)}) \tag{3.4}$$

, où :

- $df(t_i)$ est le nombre de documents contenant le terme t_i ;
- N est le nombre total de documents de la collection.

3.3.3 TFIDF

La pondération TFIDF combine les deux pondérations TF et IDF dans le but de fournir une meilleur approximation de l'importance d'un terme dans un document. Selon cette pondération, pour qu'un terme soit important dans un document, il doit apparaitre fréquemment dans le document et rarement dans les autres documents. Cette pondération est donnée par le produit de la pondération locale du terme dans le docu-

ment par sa pondération globale dans l'ensemble des documents du corpus.

$$tfidf(t_i, d_j) = tf_{ij} \times log(\frac{N}{df(t_i)}) \tag{3.5}$$

3.3.4 TFC

Cette mesure permet de pallier à l'inconvénient majeur de la mesure TFIDF à savoir la non prise en considération de la longueur des documents en lui ajoutant un facteur de normalisation. La pondération TFC d'un terme i dans un document j se calcule comme suit :

$$TFC_{ij} = \frac{TFIDF(t_i, d_j)}{\sqrt{\sum_{k=1}^{T} TFIDF(t_k, d_j)^2}} \tag{3.6}$$

3.4 Réduction de dimensionnalité

Comme tout autre domaine traitant du langage naturel, la catégorisation de textes à comme problème la grande dimension de l'espace de représentation. En effet, le nombre de descripteurs pour un corpus de taille raisonnable peut être de plusieurs dizaines de milliers. Cette grande dimensionnalité influence négativement sur la suite du processus de catégorisation en générant deux problèmes :

— un coût du traitement élevé. En effet, plus le nombre de dimensions est élevé, plus le volume de calcul est important ;

— Impossibilité de construire des règles fiables à partir de faibles fréquences de descripteurs. En effet, plus le nombre de dimensions est élevé, plus les occurrences des descripteurs deviennent faibles.

Les techniques utilisées pour la réduction de dimension sont issues de la théorie de l'information et de l'algèbre linéaire. Sebastaini [44] classe ces techniques de deux façons :

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

3.4.1 Réduction locale de dimension

Il s'agit de proposer, pour chaque catégorie c_i un nouvel ensemble de terme $\acute{T}_i$ avec $_i|Ti|$. Ainsi, chaque catégorie c_i possède son propre ensemble de termes et chaque document d_j sera représenté par un ensemble de vecteurs d_j différents selon la catégorie.

3.4.2 Réduction globale de dimension

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 quelque soit la catégorie.

3.4.3 Sélection de termes

Les techniques de réduction de dimensions par sélection consistent à filter l'ensemble de descripteurs en proposant un sous ensemble de descripteurs jugés pertinents par rapport aux autres descripteurs. Parmi ces techniques figurent :

1. **MI (Mutual Information) :** La technique MI mesure la dépendance mutuelle entre un mot t_k et une catégorie c_i. Sa formule est la suivantes[168] ;

$$MI(t_k, c_i) = log \frac{A(t_k, c_i)N}{(A(t_k, c_i) + C(t_k, c_i))(A(t_k, c_i) + B(t_k, c_i))} \tag{3.7}$$

   où :

— N est le nombre de document du corpus d'apprentissage.

— $A(t_k, c_i)$ est le nombre de documents contenant le terme t_k et appartenant à la catégorie c_i.

— $B(t_k, c_i)$ est le nombre de documents contenant le terme t_k et n'appartenant pas à la catégorie c_i.

— $C(t_k, c_i)$ est le nombre de documents ne contenant pas le terme t_k et appartenant à la catégorie c_i.

Cette technique favorise les catégories peu peuplées par rapport aux autres catégories ce qui mène à une dégradation des performances. Ce problème a été résolu par une autre alternative proposée dans[9] dont la formule est la suivante :

$$MI(t_k, c_i) = p(t_k|c_i) log \frac{A(t_k, c_i)N}{N(c_i)N(t_k)} \tag{3.8}$$

2. **La méthode CHI-2 :** La méthode CHI-2 mesure l'indépendance entre un terme t_k et une catégorie c_i en normalisant la méthode MI par la formule suivante [168] :

$$\chi^2(t_k, c_i) = \frac{N[A(t_k, c_i)D(t_k, c_i) - C(t_k, c_i)[B(t_k, c_i)]^2}{N(c_i)N(t_k)[D(t_k, c_i) + C(t_k, c_i)][D(t_k, c_i) + B(t_k, c_i)]} \tag{3.9}$$

où $D(t_k, c_i)$ est le nombre de documents ne contenant pas le terme t_k et n'appartenant pas à la catégorie c_i.

3. **Gain d'Information (IG) :** La méthode IG mesure la quantité d'information obtenue pour la prédiction de la catégorie sachant la présence ou l'absence d'un terme dans un document [168]. Sa formule est la suivante :

$$IG(t_k) = \sum_{c \in \{c_i, \overline{c_i}\}} \sum_{w \in \{w_k, \overline{w_k}\}} p(c|w) \times log \frac{p(c|w)}{p(w) \times p(c)} \tag{3.10}$$

Plusieurs travaux de recherche ont évalué les différentes techniques de sélection. Une comparaison entre les techniques IG,MI et χ^2 était effectuée dans [168] sur les corpus Ohsumed et Reuters-21578. Cette comparaison a montré que les meilleurs performances ont été obtenu avec les techniques IG et χ^2. D'autre travaux se sont focalisé sur une combinaison entre plusieurs techniques de sélection

afin de sélectionner les meilleurs termes. L'étude menée dans [128] a montré que la combinaison des termes sélectionnés de la technique χ^2 avec DF ou IG donne de meilleurs résultats.

3.4.4 Extraction de termes

Le regroupement d'attributs (Term Clustring) [142, 129, 7] est une autre alternative pour réduire la dimensionnalité. Elle consiste à représenter les documents dans un nouveau espace de représentation autre que celui d'origine. Chaque dimension du nouveau espace de représentation regroupe les termes partageant le même sens. Ainsi, les documents ne seront plus représentés par des termes mais plutôt par des regroupements de termes représentant des concepts sémantiques. Ce nouvel espace de représentation offre l'avantage de gérer la synonymie puisque les termes synonymes figureront dans les mêmes regroupements. De même, le fait qu'un terme peut figurer dans plusieurs regroupement permet aussi de gérer la polysémie.

Une autre méthode très intéressante pour l'extraction des termes et celle proposée par S. Deerwester et S. Dumais dans [30]. Cette méthode appelée LSA se base sur une décomposition en valeurs singulières de la matrice $documents \times termes$. Le but de cette décomposition est de changer la représentation en ne conservant que les k axes de plus fortes valeurs singulières.

Les méthodes de réduction basées sur l'extraction de termes sont très couteuses en terme de temps de calcul pendant l'apprentissage. De même, à chaque nouveaux documents, il est nécessaire de refaire tout le processus de regroupement.

3.5 Conclusion

Étant donnée que la représentation de textes est la première étape du processus de catégorisation de textes. Elle a suscité une grande importance de la part des chercheurs de ce domaine. En effet, une mauvaise représentation des documents influencera négativement sur le reste du processus. Le résultat de cette étape est la construction d'une matrice $descripteurs \times documents$ dont l'intersection entre la ligne i et la colonne j représente le poids du $i^{ième}$ descripteur dans le $j^{iéme}documents$. Dans ce chapitre, nous avons évoqué les étapes à suivre pour aboutir à une représentation de textes à savoir :

— Choix de descripteurs : La représentation "sac de mots" est sans doute la représentation la plus utilisé dans le domaine de la C.T. Cependant, les recherches s'accentuent de plus en plus vers les représentations sémantiques.

— Pondération de descripteurs : La majorité des pondérations utilisées dans la catégorisation de textes sont héritées du domaine de la RI, et plus particulièrement du modèle vectoriel.

— Réduction de dimensionnalité : Le nombre de dimension est un facteur très important pour les classifieurs. En effet, des classifieurs tels que les SVMs peuvent gérer un nombre élevés de dimensions tandis que d'autres classifieurs notamment les réseaux de neurones ne le sont pas.

Chapitre 4

Techniques de classification

4.1 Introduction

Après avoir représenter la collection de documents sous la forme d'une matrice $documents \times descripteurs$. Il est nécessaire de construire un modèle afin de prédire la catégorie de nouveaux documents. La construction d'un tel modèle implique l'utilisation des algorithmes d'apprentissage supervisée.

Un algorithme d'apprentissage produit une fonction (appelée aussi modèle) à partir d'un ensemble de documents $D = \{d_1, ..., d_N\}$ dont chaque document est étiqueté par une classe parmi l'ensemble des classes prédéfinies $C = \{c_1, ..., c_N\}$. Le modèle ainsi construit est utilisé pour prédire la classe d'un nouveau document non étiqueté.

Le domaine d'apprentissage automatique (AA) a donné naissance à une série d'algorithmes d'apprentissage dont il est impossible de donner une présentation exhaustive de ces algorithmes. Dans ce chapitre, nous allons présenté les algorithmes d'apprentissage les plus utilisées dans le domaine de la catégorisation de textes en citant les avantages

et les limites de chaque algorithme.

4.2 Types de classification

Dans la littérature, nous pouvons distinguer entre deux type de classification à savoir la classification "binaire" et la classification "multi-classe". La classification "binaire" consiste à définir une fonction $f : D \rightarrow [V, F]$ pour chaque classe c_i . Ainsi, pour chaque document à classer la fonction f retourne une valeur parmi deux valeurs possibles : V (vrai) si le document appartient à la classe c_i ou F (faux) sinon. Par contre, la classification "multi-classe" consiste à définir une fonction $f : D \rightarrow [0, 1]$ qui retourne une valeur comprise entre 0 et 1 pour chaque document à classer. Cette valeur est ensuite interprétée selon la méthode d'apprentissage utilisée. Une classification "multi-label" peut être résolue en combinant plusieurs classifications "binaires" selon deux principales approches :

— Dans la première approche, le classifieur binaire de chaque classe produira un score appelé CSV (Classification Status value). Le document sera assigné aux classes ayant fourni un score dépassant un seuil prédéfini. L'avantage d'une telle approche réside dans l'indépendance des classifieurs binaires. Ainsi, une fois le CSV d'une classe est calculé pour un certain document, il est possible de décider de son appartenance à la classe sans consulter les CSVs d'autres classes. Le seuil des CSVs peut être global pour toutes les classes comme il peut être local pour chaque classe. L'utilisation d'un seuil global n'est possible que si les CSVs fournis par les classifieurs sont comparables.

— La deuxième approche exclut toute notion de seuils, elle consiste à trier les CSVs fournis dans le but d'assigner le document aux N classes ayant les scores CSV les plus élevés. Cependant, il est nécessaire d'avoir des CSVs comparables.

4.3 Naïve Bayes

Largement utilisé pour sa simplicité, cet algorithme d'apprentissage a donné de bon résultats en le comparant avec d'autres algorithmes complexes [158]. En effet, il permet de réduire considérablement le temps nécessaire pour calculer la probabilité d'appartenance du document x_i à la classe c_j en utilisant l'équation :

$$P(c_j|x_i) = \frac{P(c_j)P(x_i|c_j)}{P(x_i)} \tag{4.1}$$

Étant donnée que les documents d'un corpus partagent la même probabilité, $P(x_i)$ est la même pour chaque document x_i et peut être éliminé de la formule précédente. De même, l'indépendance entre les descripteurs autorise le remplacement de $P(x_i|c_j)$ par le produit :

$$P(x_i|c_j) = P(c_j) \prod_{k=1}^{t} P(f_k|c_j) \tag{4.2}$$

La probabilité $P(c_j)$ peut être estimé en utilisant la fréquence relative d'assignement des documents de la collection à la classe c_j :

$$\hat{P}(c_j) = \frac{n_j}{n} \tag{4.3}$$

où n_j représente le nombre de documents assignés à la classe c_j. De même la probabilité $P(f_k|c_j)$ peut être approximée comme suit :

$$\hat{P}(f_k|c_j) = \frac{1 + n_{kj}}{l + \sum_{h=1}^{l} n_{hj}} \tag{4.4}$$

avec n_{hj} : le nombre de document assignés à la classe c_j contenant le descripteur f_h et l le nombre total de descripteurs. Finalement, la probabilité qu'une classe c_j affecte un certain document x_i peut être calculée sans avoir à pondérer les descripteurs par la

formule suivante :

$$\hat{P}(c_j|x_i) = \frac{n_j}{n} \prod_{k=1}^{l} \frac{1 + n_{kj}}{l + \sum_{h=1}^{l} n_{hj}} \tag{4.5}$$

L'algorithme de base considère un document comme un vecteur binaire. Ainsi, il se base sur les probabilités de présence/absence des descripteurs sans prendre en considération leurs nombres d'occurrences. Une extension du modèle de base appelée "modèle multinomial" permet de prendre en considération cette information supplémentaire pour estimer la probabilité d'appartenance d'un document à une classe via la formule suivante :

$$\hat{P}(c_j|x_i) = P(|x_i|)|x_i|! \prod_{h=1}^{l} \frac{P(f_h|c_j)^{freq(f_h,x_i)}}{freq(f_h, x_i)!} \tag{4.6}$$

L'avantage des classifieurs bayésiens réside dans leurs simplicité ainsi que le peu d'informations utilisées . En effet, ils sont basés sur un simple calcul de co-occurences sans aucune pondération de descripteurs. Néanmoins, quelques dégradations peuvent apparaitre en appliquant d'autre variations(Multi-variate Bernouli event model). Une étude comparative entre les deux modèles été menée dans [101] sur quatre différents corpus. Cette comparaison a montré que le modèle "multinomial" est plus performant que le modèle "multi-variate Bernoulli".

4.4 La méthode Rocchio

La méthode Rocchio [127] a été initialement proposée pour la recherche d'information dans le but de réaliser une reformulation de la requête. Le principe consiste à transformer la requête initiale de l'utilisateur en une requête optimale maximisant la distance entre les documents pertinents et les documents non pertinents [68]. La discrimination entre les documents pertinents et les documents non pertinents est calculée à

travers la formule suivante :

$$Q_1 = Q_0 + \frac{1}{n_r}\sum_{i=1}^{n_r} R_i - \frac{1}{n_s}\sum_{i=1}^{n_s} S_i \tag{4.7}$$

avec :

- Q_0 le vecteur de la requête initiale et Q_1 le vecteur de la nouvelle requête.
- n_r est le nombre de documents pertinents et n_s le nombre de documents non pertinents
- R_i est le vecteur du $i^{éme}$ document pertinent et S_i le vecteur du $i^{éme}$ document non pertinent.

Comme la majorité des techniques de classification, la méthode Rocchio a été réutilisée dans le domaine de la catégorisation de textes pour calculer un profil prototypique p_j pour chaque classe c_j. Ce profil est censé discriminer la classe par rapport aux autres classes via la formule suivantes [157] :

$$p_{jk} = max\frac{t}{|c_j|}\sum_{d_j \in c_j} d_{ik} - \frac{1-t}{|\bar{c_j}|}\sum_{d_j \notin c_j} d_{ik} \tag{4.8}$$

Lors de la classification d'un nouveau document, il faut calculer la mesure de similarité entre les profils des classes et le vecteur correspondant au nouveau document. Le document sera assigné finalement à la classe dont le profil est le plus proche au vecteur du document. La méthode Rocchio est caractérisée par sa simplicité et sa rapidité d'exécution dans les deux phases à savoir la phase d'apprentissage et la phase de classification. Cependant plusieurs études comparatives [77, 166] ont montré que le classifieur Rocchio est moins performant que d'autres classifieurs plus complexes. Cohen et Singer ont montré dans [22] qu'un réajustement des paramètres du classifieur Rocchio peut améliorer les performances. Une autre alternative (appelée *Query Zoning*) proposée par Schapire et al. dans [136] a permis d'apporter une amélioration avoisinant les 10%. L'idée consiste à réduire l'influence des exemples négatifs sur la construction des profils en ne prenant comme exemples négatifs que ceux les plus proches du bary-

centre des exemples positifs. Moschitti a étudié dans [108] les paramètres du classifieur Rocchio dans le but de construire un modèle permettant la sélection automatique et rapide des paramètres optimales. Le modèle construit a été testé sur trois corpus dans deux langues différentes et les résultats ont montré que les performances du classifieur Rocchio sont relativement proches à ceux obtenu par les meilleurs classifieurs (tels que les SVMs).

4.5 Les séparateurs à vaste marges (SVM)

Les SVMs ont été proposés pour la première fois dans [24, 156] pour le traitement de données numériques. L'idée principale des SVMs consiste à déterminer le meilleur séparateur entre classes dans l'espace de représentation. L'exemple de la figure4.1 illustre la représentation des éléments de deux classes notés 'x' et 'o'. Parmi Les trois hyperplans séparateurs notés A, B et C, l'hyperplan A est le meilleur séparateur car il est le plus distant de tout les éléments offrant ainsi la plus grande marge de séparation. La marge représente la plus petite distance entre les exemples de chaque classe et la surface séparatrice et se calcule par la formule suivante :

$$marge(S) = \sum_{c_j \in C} min_{x_i \in c_j}(d(x_i, S)) \tag{4.9}$$

Formellement, il s'agit de trouver l'hyperplan (w, b) minimisant la norme de w sous les contraintes :

$$\forall d_i, c_i(w.d_i - b) \leq 1 \tag{4.10}$$

avec d_i le $i^{ème}$ document, et b la distance à l'origine de l'hyperplan.

Dans certain cas, il est difficile de séparer efficacement n'importe quel jeu de données par un simple hyperplan. On dit alors que les classes ne sont pas linéairement séparables et que leurs exemples se chevauchent. La solution consiste à projeter les données dans

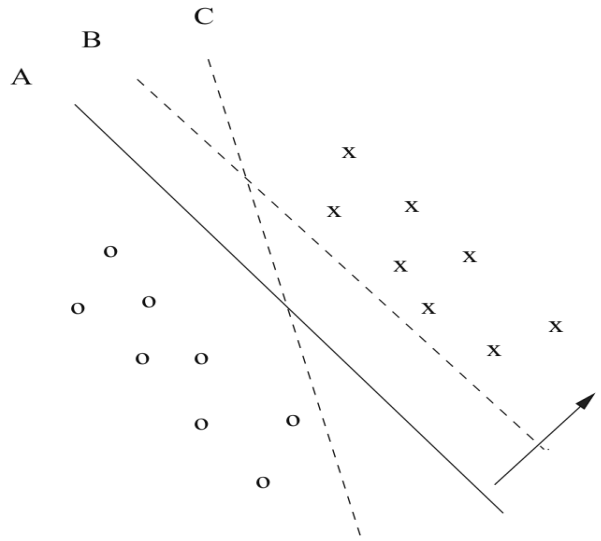


FIGURE 4.1 – *Exemples d'hyperplans séparateurs*

un espace de plus grande dimension où il est possible de séparer linéairement les classes. Cette transformation de l'espace de représentation est réalisée par une fonction k dite "fonction noyau". Dans le cas où la transformation de l'espace de représentation ne suffit pas pour séparer linéairement les classes, une deuxième solution consiste à autoriser un nombre limité d'exemples à être mal classés.

L'avantage des SVMs réside dans leurs capacité à gérer des vecteurs de grande dimension. Selon Joachims [76], les SVMs sont bien adaptés au traitement de données textuelles du fait que les textes se caractérisent par la grande dimensionnalité qui génère des descripteurs peu fréquents mais qui peuvent être utiles en leurs corrélant avec d'autres descripteurs pour former des catégories linéairement séparables. les SVMs ont été largement utilisés dans différents domaines traitant des données textuelles. H. Drucker et al. ont utilisé dans [35] les SVMs pour le filtrage des spams et ont confirmé que les SVMs produisent les meilleurs performances en les comparant avec d'autres méthodes de classification tels que la méthode Rocchio et les arbres de décision. Dumais et al. ont montré aussi dans [36] l'utilisation bénéfique des SVMs pour le cas de la classification hiérarchique du web. Les SVMs présentent néanmoins quelques limites. Ils ont du mal

à fournir de bon résultats dans le cas des espaces de très petite dimension. Ajoutons à cela, le fait qu'il est difficile à l'utilisateur de prédire les paramètres fournissant les meilleurs performances.

4.6 K plus proches voisins

la méthode des K plus proches voisins (K Nearest neighbors) est une méthode de classification se basant sur le rapprochement entre le vecteur associé au document à classer et les vecteurs associés aux documents d'apprentissage. Le classifieur KNN fait partie des méthodes dite "*lazy*" marquée par l'absence de la phase d'apprentissage. Pour classer un nouveau document, il s'agit d'utiliser une mesure de similarité pour sélectionner les k documents les plus proches au document à classer. Le document sera assigné à la classe la plus représentée dans l'ensemble des k voisins. La méthode KNN se caractérise par sa simplicité et sa facilité d'implémentation. Cependant, elle est très couteuse en terme d'espace mémoire car elle exige de stocker tout les vecteurs de documents dans la mémoire. De plus, elle nécessite un temps de calcul considérable qui devient de plus en plus pénalisant dans le cas de larges corpus d'apprentissage. Un autre problème réside dans le choix de la valeur du paramètre k. En effet, une grande valeur de k permet de réduire l'effet du bruit ainsi que le risque du sur-apprentissage mais rend les frontières entre classes moins distinctes. Plusieurs travaux se sont focaliser sur l'amélioration du classifieur KNN. Bailey et al. ont proposé dans [6] une variante appelée WKNN qui consiste à associer une pondération aux k plus proches voisins en fonction de leurs similarité avec le document à classer. Ainsi, les documents les plus proches doivent avoir un poids plus élevé dans la décision que les documents voisins qui sont plus éloignés.

4.7 les algorithmes de Boosting

Le" Boosting" est une technique d'apprentissage automatique qui consiste à combiner plusieurs classifieurs d'une façon itérative dans le but d'améliorer les performances. Ainsi, il s'agit de construire un classifieur composé de plusieurs classifieurs peu performant mais très rapide. L'originalité des algorithmes de boosting réside dans le fait que les différents classifieurs combinés ne sont pas construit indépendamment les uns aux autres mais d'une façon séquentielle. Ainsi, le $(i+1)^{iéme}$ classifieur est construit en prenant comme ensemble d'apprentissage les documents dont les i classifieurs précédemment construit n'ont pas pu classifier correctement. Finalement, ces différents classifieurs sont combinés en affectant un poids à chaque classifieur selon son taux d'erreurs. AdaBoost [43] est un algorithme de type Boosting largement utilisé. Il est basé sur une pondération des documents selon leurs difficulté de classification. Au fil des itérations, les documents mal classifiés voient leur poids augmenter et les documents bien classifiés, leur poids diminuer. Cela revient donc à forcer les classifieurs à ce concentrer sur les documents mal classés. Les auteurs de cet algorithme ont montré dans [134] l'absence du sur-apprentissage.

Kim et al. ont utilisé dans [80] l'algorithme AdaBoost pour faire de la catégorisation de textes. Ils ont combiné des classifieurs Naive bayes afin de permettre à l'algorithme AdaBoost d'utiliser la fréquence de termes (TF) comme pondération au lieu des pondérations binaires. Shapire et Singer ont proposé dans [135] une approche de boosting nommée BoosTexter combinant plusieurs techniques de classification pour la catégorisation de textes . Les résultats ont montré que l'approche proposée a donnée les meilleurs performances par rapport aux classifieurs Rocchio et Naive Bayes.

4.8 Les arbres de décisions

Les arbres de décision sont des méthodes supervisées parues depuis les années 60, leurs principe se base sur une décomposition hiérarchique des exemples selon un certain nombre de prédicats. Dans le cas de données textuelles, les prédicats correspondent à des conditions sur la présence/absence des termes. L'objectif des arbres de décision est la construction d'un arbre dont chaque nœud correspond à un sous ensemble d'exemples et chaque arrête correspond à un prédicat. La construction de tels arbres se base sur un partitionnement récursive de l'ensemble des exemples selon un certain critère. Le partitionnement prend en entrée l'ensemble total des exemples constituant la racine de l'arbre pour évaluer les différents partitionnements possibles à savoir un partitionnement pour chaque variable ou attribut de l'espace de représentation. Après avoir évaluer les partitionnements possibles, il faut choisir le partitionnement maximisant un certain critère. Chaque sous ensemble de la partition représente un nœud qui sera par la suite partitionné. Le processus se réitère jusqu'à ce qu'aucune amélioration du critère ne soit plus possible. Il existe plusieurs variantes des arbres de décision qui se différencient essentiellement par le critère de partitionnement. Parmi les plus célèbre figure l'algorithme CHAID[79] dont le partitionnement est basé sur la statistique χ^2, L'algorithme ID3 [117] et l'algorithme C4.5 [118] dont le partitionnement est basé sur l'entropie de Shannon. L'exemple illustré dans la figure 4.2 représente un arbre de décision correspondant à l'exemple de golf décrit dans [117] dont l'objectif et de prédire s'il est possible de jouer du golf en fonction de quatre variables décrivant les conditions climatiques. L'arbre de décision est construit à partir du jeu de données décrit dans le tableau 4.1. Le premier partitionnement est établi selon la variable "Extérieur" et a généré trois nœuds, le premier nœuds a généré aussi deux nœuds par partitionnement selon la variable "Humidité" tandis que le troisième nœuds a généré deux nœuds par partitionnement selon la variable "Vent". Une fois l'arbre construit, il est possible d'extraire les règles de décision permettant de classer un nouveau exemple. Les arbres de décision ont l'avantage de posséder une structure hiérarchique les rendant compréhensibles pour

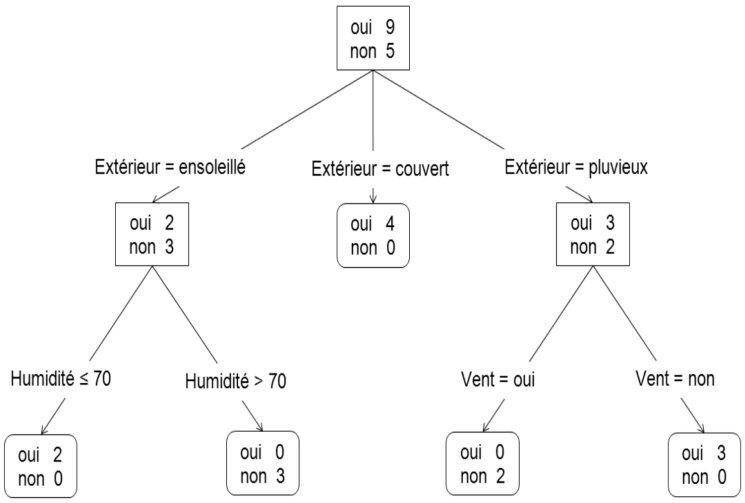


FIGURE 4.2 – *Exemples d'arbre de décisions*

Extérieur	Vent	Température	Humidité	Golf
ensoleillé	non	29	85	non
ensoleillé	oui	27	90	non
couvert	non	28	78	oui
pluvieux	non	21	96	oui
pluvieux	non	20	80	oui
pluvieux	oui	18	70	non
couvert	oui	18	65	oui
ensoleillé	non	22	95	non
ensoleillé	non	21	70	oui
pluvieux	non	24	80	oui
ensoleillé	oui	24	70	oui
couvert	oui	22	90	oui
couvert	non	27	75	oui
pluvieux	oui	22	80	non

TABLE 4.1: *L'ensemble des exemples du problème de golf*

l'utilisateur ainsi que leurs capacité de transformer un problème complexe en plusieurs problèmes simples. Cependant, ils souffre du problème du sur-apprentissage [98].

4.9 Réseaux de neurones

Les réseaux de Neurones sont des classifieurs modélisant la manière dont le cerveau humain fonctionne. En effet, le cerveau humain contient un nombre important de cellules nerveuses, chaque cellule est connectée avec d'autres cellules similaires formant ainsi un réseau complexe de signaux de transmissions. Chaque cellule collecte ces entrées à partir d'autres cellules et génère des sorties vers d'autres cellules. Les réseaux de neurones artificiels tentent de réaliser cette modélisation. Stricker définit formellement un neurone dans [146] comme une fonction algébrique paramétrée, à valeurs bornées, de variables réelles x_i (ayant des poids w_i) appelées entrées dont leurs combinaison linéaire représente le potentiel du neurone et est calculé comme suit :

$$v = w_0 + \sum_{i=1}^{n} w_i \times x_i \tag{4.11}$$

le résultat de la fonction représente la sortie du neurone.

les réseaux de neurone ont été utilisé dans de nombreux travaux de classification de textes. Wiener et al. ont proposé dans [162] deux approches utilisant l'un des plus célèbre réseaux de neurones à savoir le *perceptron multi-couche*. La première approche consiste à construire un réseaux de neurones pour chaque classe tandis que la deuxième consiste a regroupé les classes en cinq ensembles et de construire un réseaux de neurones pour chaque ensemble de classes. Les résultats ont montré que les meilleurs performances ont été obtenu avec la deuxième approche.

4.10 Conclusion

Le choix de technique d'apprentissage est considéré comme le cœur du processus de catégorisation de textes. La majorité des algorithmes d'apprentissage sont originaire de

deux domaines à savoir le domaine de l'apprentissage automatique et le domaine du Datamining. Dans le domaine du textmining, les recherches se sont orientées beaucoup plus sur les algorithmes linéaires tels que les réseaux de neurones et les SVMs du fait qu'ils s'adaptent mieux aux particularités des données textuelles à savoir le nombre élevé de dimensions. Plusieurs travaux de recherche se sont focalisés sur la comparaison des différents algorithmes dans la catégorisation de textes. Cependant, il est actuellement impossible de confirmer qu'un algorithme est meilleur qu'un autre. En effet, deux critères peuvent influencer le choix du meilleur classifieur à savoir l'exactitude et la rapidité. Ainsi, si l'exactitude des résultats et la minimisation des erreurs sont plus important que l'aspect temps de classification, on peut pencher vers des méthodes plus complexes et plus performantes, par contre si on veut avoir une classification rapide on peut choisir des classifieurs simples. Face à ce dilemme, plusieurs recherches s'intéressent à la combinaison de plusieurs classifieurs selon plusieurs stratégies de combinaison.

Chapitre 5

Les ontologies

5.1 Introduction

Durant ces dernières années, les ontologies deviennent de plus en plus utilisées pour la représentation des connaissances dans différents domaines manipulant des connaissances incomplètes, complexes ou évolutives. Les ontologies sont l'un des concepts de base du "Web Sémantique" dont l'objectif principale est d'assurer une meilleur exploitation des données et services du web non seulement par les humains mais aussi par les programmes. En effet, le web est structuré par des formats (tel que HTML) compréhensibles par l'humain. Cependant, il est quasiment impossible qu'un programme puisse l'interpréter. En d'autres termes, un programme agissant sur le web peut construire, transporter, traiter et produire de l'information à un utilisateur sans considérer le contenu sémantique. Comme son non l'indique, le domaine du "web sémantique" consiste à proposer une vision sémantique du Web en étiquetant sémantiquement ces ressources par des métadonnées afin que les machines puissent les manipuler d'une manière similaire à leur manipulation par l'utilisateur.

5.2 Qu'est ce qu'une ontologie ?

Le terme "Ontologie" est apparu la première fois en $17^{éme}$ siècle dans le lexique philosophique de Rudolf Gockel. Étymologiquement, le terme Ontologie est composé de *"ont"* le participe présent du verbe Grecque *einai* et de *"logia"* i.e l'étude , ce qui se traduit par "*l'étude de ce qui existe*". Selon Barry Smith [143], la première apparition du terme "ontologie" dans le monde informatique remonte à 1967 dans les travaux de modélisation de données de S.H. Mealy [103]. Dans ces travaux, Mealy affirme que l'existence des objets dans le monde selon leurs multiple représentations peut être modélisée par les ontologies. L'utilisation des ontologies dans le monde informatique remonte aux années 90 dans le but de créer une représentation des connaissances d'un domaine. Cela a motivé la création de plusieurs forums tels que les séries de la conférence FOIS (Formal Ontology and Information Systems). La proposition de Clancy dans [20] est la plus importante dans ce sens. En effet, les approches classiques dans le domaine d'Intelligence Artificielle consistait à répliquer le comportement d'un expert dans la base de connaissances. Ainsi, il ne s'agissait pas de modéliser les données mais de modéliser la démarche de l'expert. Selon Clancy, l'objectif de l'ingénierie des connaissances est de modéliser réellement les connaissances des systèmes. En d'autre termes, il s'agit d'établir une séparation entre la modélisation des connaissances du domaine modélisé et les connaissances de raisonnement décrivant les règles heuristiques d'utilisation de ces connaissances du domaine.

Les définitions les plus populaires et originales de l'ontologie sont celles énoncé par Gruber [56] qui la définie comme une *"spécification explicite d'une conceptualisation"* et celle de Borst [16] qui la défini comme *"une spécification formelle d'une conceptualisation partagée"*. En 1998, Studer et al. ont fusionné dans [147] les deux définitions pour définir l'ontologie comme une *"spécification formelle et explicite d'une conceptualisation partagée"*. Ces définitions se basent sur les quatre notions suivantes[2] :

— Formelle : Description de l'ontologie par un langage compréhensible par la ma-

chine en excluant le langage naturel.

— Explicite : les concepts avec leurs contraintes d'utilisation doivent être explicitement définis.

— Conceptualisation : le modèle abstrait d'un phénomène du monde réel par identification des concepts clefs de ce phénomène.

— Partagée : l'ontologie n'est pas la propriété d'un individu, mais elle représente un consensus accepté par une communauté d'utilisateurs.

Di jorjio et al. ont proposé dans [32], une définition formelle qui considère l'ontologie comme un tuple $O = \{C, T, R_c, R_t, L, <_c, f_{tc}, f_{rc}\}$ où :

— C est l'ensemble de concepts.

— T est l'ensemble de termes.

— R_c est l'ensemble de relations entre concepts.

— R_t est l'ensemble de relations entre termes.

— L est l'ensemble de labels de relations (étiquette sémantique permettant de nommer une relation.

— $<_c$: $C * C$ est la relation d'ordre partiel sur C définissant la hiérarchie entre les concepts, $<_c$ (c_1, c_2) signifie que c_1 est plus général que c_2.

— f_{tc} est la fonction d'association d'un terme préféré à un concept.

— f_{rc} est la signature d'une fonction associative entre concepts.

Pour illustrer cette définition, les auteurs ont donnée un exemple d'ontologie concernant les perturbations atmosphériques. Comme montré dans la figure 5.1, Les concepts sont représentés par des rectangles, les termes par des diamants et les relations par des ellipses. L'ensemble des concepts C regroupe $\{c_1, c_2, c_3, c_4\}$, l'ensemble des termes est $T = \{Perturbationatmosphérique, Orage, Averse, Pluie, Bruine\}$, et l'ensemble des relations R_c est constitué d'une seule relation, de label *"Entraîne"*. Le terme "Perturbation atmosphérique" est le terme préféré du concept c_1 : lorsque nous désignons le concept c_1, nous désignons tous les phénomènes de perturbations atmosphériques. L'existence d'une relation $f_{rc}(Entraine) = (c_2, c_4)$ signifie que l'orage entraîne la pluie. La hiérarchie des concepts $<_c$ est indiquée par les flèches simples et spécifie par exemple que le concept *Orage* est un sous-concept de *Perturbation atmosphérique*, qui sera qua-

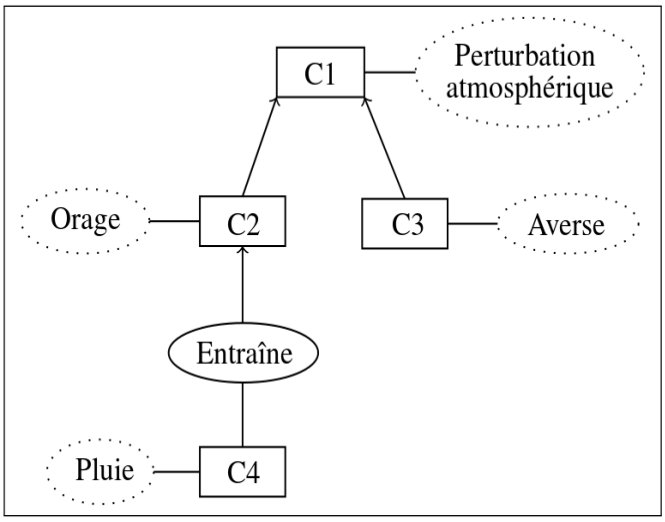


FIGURE 5.1 – *Exemples d'ontologie concernant les perturbations atmosphériques [32]*

lifié de père du concept *Orage*. Le terme *pluie* désigne un concept de l'ontologie et *entraîne* un label de relation de l'ontologie, alors que *provoquer* ou *innondation* sont des items du motif séquentiel <(pluie)(provoquer innondation)>.

Hotho et Staab ont proposé dans [69] une autre définition formelle orientée vers le domaine de la catégorisation de textes qui consiste à définir l'ontologie comme un quintuple $\{L, F, C, H, ROOT\}$ avec :

- L : un lexique contenant un ensemble de termes.
- C : l'ensemble des concepts.
- F : la fonction de référencement qui relie un ensemble de termes à un ensemble de concepts.
- H : l'hétérarchie de concepts dont les concepts sont reliés par une relation directe, acyclique, transitive et réflexives. Ainsi, $H(c_1, c_2)$ signifie que c_1 est un sous concept de c_2.
- ROOT : la racine de l'hétérarchie des concepts tels que : $\forall c \in C : H(c, ROOT)$

FIGURE 5.2 – *Exemple illustratif de la définition formelle d'une ontologie[69]*

5.3 Constituants d'ontologie

Selon Gruber [56], le formalisme d'ontologie se base sur cinq composants à savoir les concepts, les relations, les fonctions, les instances et les axiomes :

— Concepts : les concepts sont des entités cohérentes dotées d'une existence indépendante. Chaque concept représente ainsi un groupe d'individus partageant les mêmes caractéristiques. Comme montré dans la figure5.3, Un concept est composé de trois éléments constituant les sommets d'un triangle sémantique. Il s'agit de [153] : (1) un ou plusieurs termes exprimant le concept en langage naturel, (2) la signification sémantique du concept sous forme d'attributs et propriétés, appelée également *notion* ou *intension* du concept, (3) les objets exprimés par le concept, appelés également *extension* du concept.

— Relations : les relations permettent de structurer les connaissances en décrivant les différentes interactions qui existent entre les concepts de l'ontologie. Il existe deux types de relations à savoirs les relations taxonomiques et les relations associatives. Les relations taxonomiques organisent les concepts hiérarchiquement en reliant un concept spécifique à un concept générique. les relations associatives

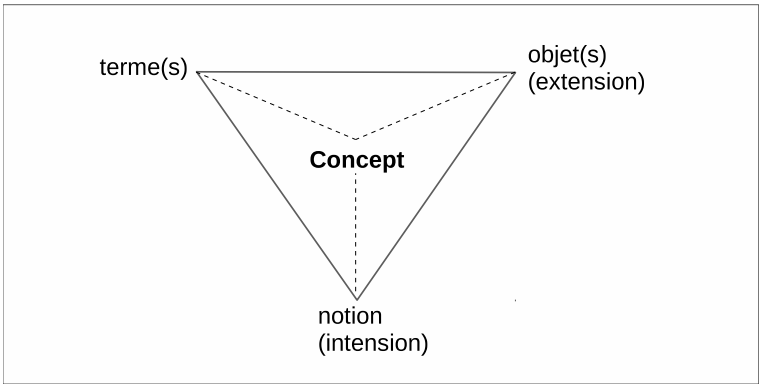


FIGURE 5.3 – *Les éléments d'un concept*

désignent toutes relations non-taxonomique entre les concepts.

— Instances : les instances sont les objets appartenant au concept. Par exemple *Opel* est une instance du concept *voiture*. Vu que les ontologies sont supposées représenter conceptuelement un domaine, la majorités des ontologies ne contiennent pas d'instances. Une ontologie contenant des instances devient alors une base de connaissances.

— Fonctions : les fonctions sont des cas particuliers des relations où le $n^{ème}$ élément de la relation est unique pour les $n-1^{éme}$ éléments précédents.

— Axiomes : les axiomes explicitent des énoncés conceptuels toujours vrais dans le contexte de l'ontologie. Ils sont utilisés pour contrôler la signification des concepts et relations, de restreindre la valeur des attributs ou encore de vérifier la validité des informations spécifiées [66].

5.4 Construction d'ontologie

Selon Subhashini [119], la construction d'ontologies suit un des trois schémas suivants (voir figure5.4) : Schéma simple, schéma multiple et schéma hybride. Ces différents schémas se différencient selon la manière de gérer les sources d'informations. Dans le schéma simple, il s'agit de construire une seule ontologie globale pour toutes

les sources d'informations en s'appuyant sur le vocabulaire partagé entre ces différentes sources d'informations. Cependant, il est nécessaire d'avoir des sources d'informations partageant la même vue sur le domaine pour assurer l'intégration d'information. De plus, tout changement dans les sources d'informations risque d'affecter la conceptualisation. Dans le schéma multiple, il s'agit de construire une ontologie locale pour chaque

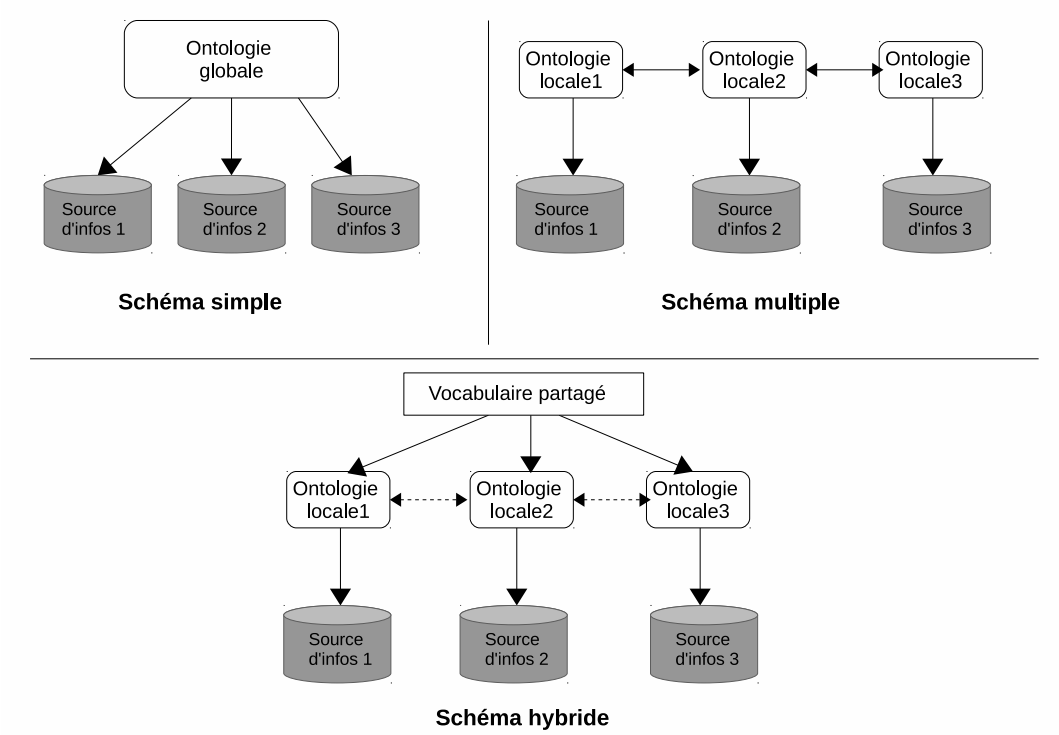


FIGURE 5.4 – *les trois schémas de construction d'ontologie [159]*

source d'informations. L'avantage est l'indépendance des sources d'informations. Ainsi, les changements d'une source d'informations n'affecteront pas les autres ontologies. Cependant, le manque d'un vocabulaire commun rend difficile la comparaison entre les différentes ontologies locales. La solution consiste à définir un mapping inter-ontologie reliant les termes des différentes sources d'informations. Selon [159], il est très difficile de définir ce mapping inter-ontologie à cause de l'hétérogénéité sémantique. Avec un schéma hybride, les ontologies locales sont construites à partir d'un vocabulaire partagé global contenant les termes de base du domaine. L'utilisation d'un vocabulaire partagé garantira la construction d'ontologies locales comparables.

Plusieurs travaux se sont focalisés sur la proposition d'une méthode pour la construction d'ontologie(TOVE[57], Entreprise Model Approach [154], Methontology[60, 39], IDEF5[104]). Ces méthodes partagent les étapes principales suivantes [34] :

— Identification du domaine de l'ontologie ainsi que son champ d'application ;

— Définition des objectifs attendus de l'ontologie ;

— Spécification informelle des concepts ;

— Codage de l'ontologie en représentant formellement les concepts et les axiomes ;

— Évaluation de l'ontologie.

Après avoir analyser les travaux de Gandon [49] et Leclerc [87] sur les ontologies, Mhiri el al. ont dégagé dans [105] un consensus sur le processus de conception d'ontologie. Comme illustré dans la figure 5.5, ce consensus nécessite trois étapes pour transformer des données brutes à une ontologie opérationnelle. La première étape intitulée *"concpetualisation"* consiste à identifier les connaissances à partir d'un corpus représentatif du domaine dans le but de générer un modèle conceptuel formé d'un ensembles de concepts et de relations entre ces concepts. La deuxième étape est celle de *l'ontologisation*, et consiste à formaliser le modèle conceptuel en utilisant un langage générique. La dernière étape intitulée "opérationnalisation" consiste a doté l'ontologie construite de mécanismes d'inférence en utilisant un langage formel.

5.5 Types d'ontologies

Les ontologies sont généralement classées selon la quantité et le type des structures utilisées dans leur conceptualisation. Une des premières classification d'ontologies est celle établi par Sowa dans [144]. La classification de Sowa permet de distinguer entre trois types d'ontologies à savoir les ontologies formelles, les ontologies terminologiques et les ontologies basées sur les prototypes. Comme montré dans l'exemple illustatif de la figure 5.6, dans les ontologies terminologiques, les concepts et les relations ne sont pas totalement spécifiés par des axiomes et des définitions déterminant les conditions

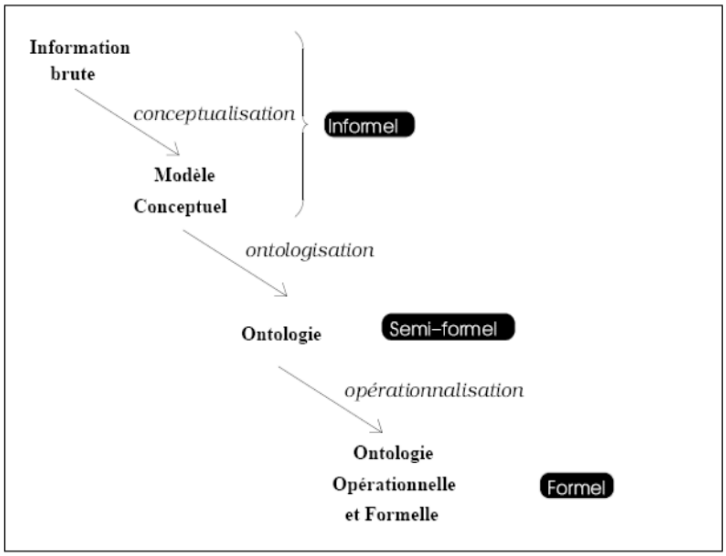


FIGURE 5.5 – *Processus de construction d'ontologie [105]*

suffisantes et nécessaires de leurs utilisations. Ainsi, un concept est simplement spécifié à travers ces relations avec les autres concepts. Les ontologies formelles se caractérisent par la spécification de leurs concepts et relations par des axiomes et des définitions exprimés en Logique. Les ontologies basées sur les prototypes sont construit par regroupement hiérarchique des données en plusieurs prototypes. Ainsi, chaque prototype correspond à un concept de l'ontologie.

Deux classification proposées dans [155] permettent de classer les ontologies selon la structure utilisée dans la conceptualisation ou le sujet de conceptualisation. En se basant sur la structure de conceptualisation, une ontologie peut être :

— ontologie terminologique spécifiant les termes utilisés pour spécifier les connaissances(lexique, glossaires,..) ;
— ontologie d'informations structurant les termes (schéma d'une base de données) ;
— ontologies de connaissances spécifiant la conceptualisation des connaissances.

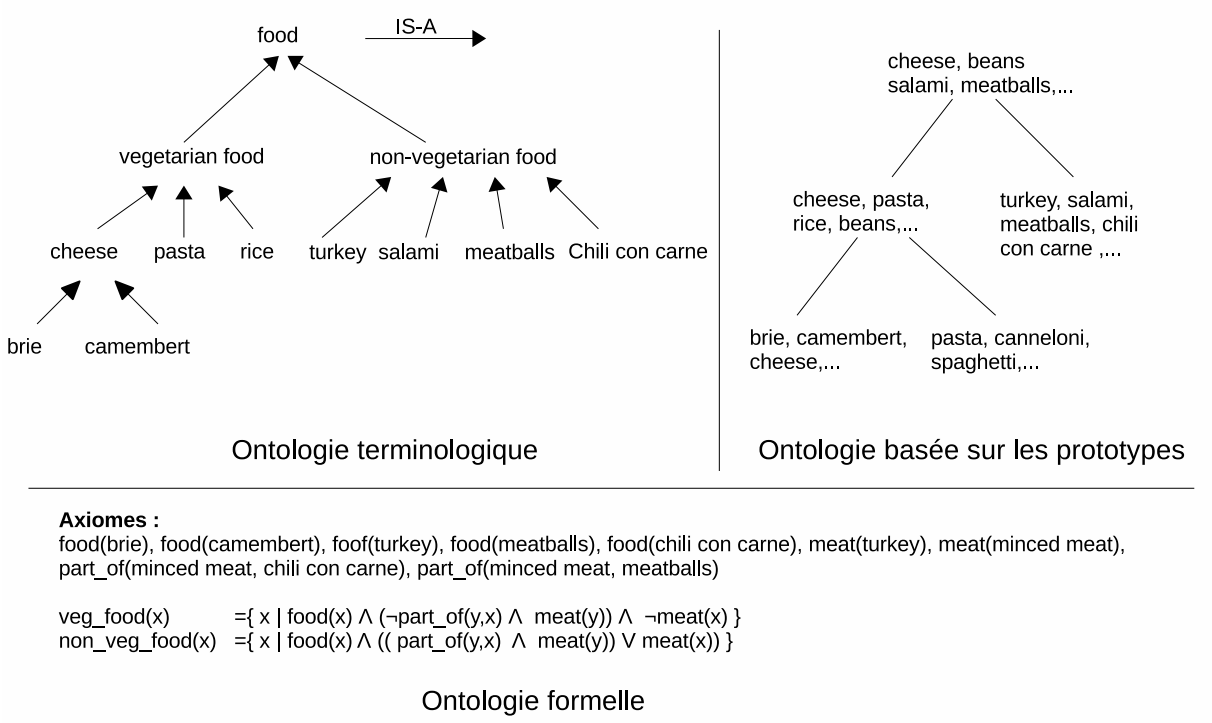


FIGURE 5.6 – *Exemple illustratif de le classification de SOWA[13]*

Selon [82], en comparant cette classification avec celle de SOWA, les ontologies terminologiques ont les mêmes définitions, les ontologies de connaissances font partie des ontologies formelles. Cependant, les ontologies d'informations se situent au milieu des deux types d'ontologie (terminologique et formelle).

En se basant sur le sujet de conceptualisation, une ontologie peut être (voir figure 5.7) :

— ontologie d'application : fournissant le nécessaire pour modéliser les connaissances pour une application particulière. L'objectif est d'associer un langage à l'application permettant de faciliter l'intégration de l'application dans un environnement.

— ontologie de domaine [153] : décrivant les connaissances d'un domaine spécifique. Ces ontologies se basent sur une description du vocabulaire utilisé dans le domaine.

— ontologie générique[89, 144] : décrivant des concepts généraux qui ne dépendent

pas d'un domaine particulier tels que les concepts de temps et d'espace.

— ontologie de représentation [100, 56] : fournissant des structures de représentation utilisées par les ontologies de domaine et les ontologies génériques pour la description de leurs concepts.

— ontologie de taches : fournissant les concepts et relations utilisées pour la spécification d'un processus de raisonnement dans le cadre de réalisation d'une tache particulière.

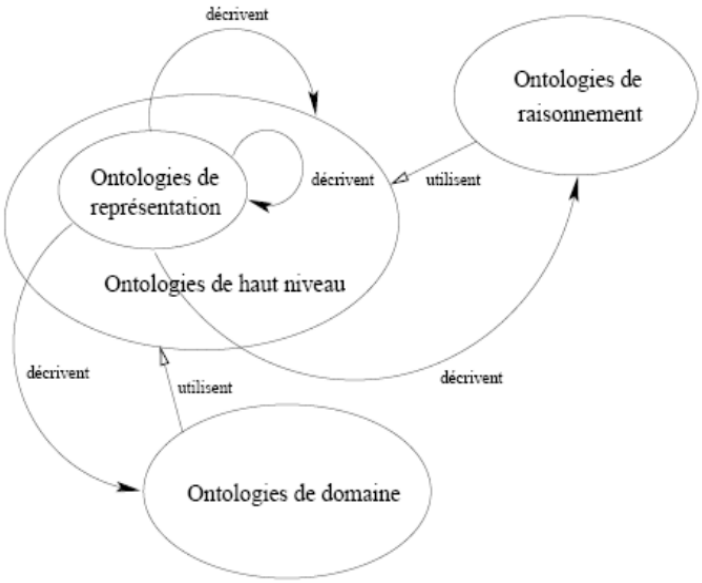


FIGURE 5.7 – *Types d'ontologies*

Uschold et Gruninger proposent dans [153] une autre classification des ontologies qui se basent sur leurs niveaux de formalisation. Ainsi, une ontologie peut être :

— informelle : si elle est exprimée en langage naturel. L'inconvénient majeur de ces ontologies réside dans l'absence de formalisation qui rend difficile leurs opérationnalisation.

— semi-informelles : si elle est exprimée par un langage structuré et dérivé du langage

naturel offrant un formalisme restreint qui permet de réduire l'ambiguïté. La version de teste de l'ontologie *Entreprise*[42] est un exemple de ce type d'ontologie.

— semi-formelle : si elle est exprimée par un langage formel artificiel. C'est le cas de la version Ontolingua de l'ontologie *Entreprise*.

— formelle : si elle est exprimée par un langage formel artificiel dont les termes sont méticuleusement définit par la sémantique formelle. L'ontologie TOVE [41] est un exemple.

5.6 Langages de représentation d'ontologie

Différents langages de spécification d'ontologies ont été développé durant cette dernière décennie. Ces langages ont pour objectif d'assurer une formalisation de connaissances garantissant un meilleur partage. Nous citons dans ce qui suit, quelques langages orientés Web Sémantique recommandés par le World Wide Web Consortium (W3C) :

— **XML (eXtended Markup Language) :** est un langage permettant de fournir deux structures pour un document : une structure logique ainsi qu'une structure physique. Ainsi, XML structure un document sous forme d'un ensemble d'éléments à l'aide de balises. XML-schéma permet de définir la structure, les contraintes, et la sémantique de documents XML. Les primitives qu'il met en place sont plutôt orientées application que concept. En effet, la sémantique définie dans le document est interprétable dans le contexte de l'opération faite sur le document mais ne permet pas d'établir des inférences en dehors de ce contexte. XML et XML-schéma sont considérés comme des langages définissant le format de «message» alors qu'un langage d'ontologies a pour but de représenter la connaissance.

— **RDF (Resource Description Framework)** permet d'encoder, d'échanger et de réutiliser des méta-données structurées. Il consiste à décrire les données ainsi que les méta-données à l'aide d'un ensemble de triplets appelés "graphe RDF". Chaque triplet se compose d'un sujet, un prédicat et un objet afin de représenter

les relations entre les choses. Il permet de définir des ressources avec des propriétés et des états. RDF-Schéma définit les relations entre ces ressources. Le pouvoir sémantique de ces deux langages est limité car les axiomes ne peuvent pas être directement décrits. Le type des relations (symétrique, transitive, ...) ne peut être spécifié.

— **OIL (Ontology Inference Layer)** est un langage basé sur trois éléments essentiels : les frames, la logique de description et les standards du web (RDF(s) et XML). La description de l'ontologie est divisée en trois couches : la couche objet (instances concrètes), la couche de premier méta-niveau (définition de l'ontologie) et la couche de second meta-niveau (définition des caractéristiques de l'ontologie). OIL permet de définir des classes et des relations et un nombre limité d'axiomes. Les relations sont considérées comme des classes et peuvent être organisées hiérarchiquement.

— **OWL (Ontology Web Language)** est un langage XML offrant un degré d'interprétation plus élevé que RDF et RDFS. En effet, l'OWL offre des mécanismes de comparaison des propriétés et des classes (identité, équivalence, contraire, cardinalité, symétrie, transitivité, disjonction, etc). Une ontologie OWL est composée d'un en-tète (méta-données), d'axiomes et de faits. Les axiomes concernent la définition complète ou partielle de concepts et de relations, la spécification de propriétés sur les relations et la définition d'axiomes sur les classes et les relations. OWL est fournit avec les trois sous langages suivants :

1. Le langage OWL-LITE permettant la modélisation d'ontologies ayant une complexité formelle peu élevée. Son expressivité est limitée à des hiérarchies de classification et de fonctionnalités de contraintes simples de cardinalité 0 ou 1.

2. Le langage OWL-DL offrant une expressivité élevée avec des raisonnements plus complexes en DL (Logique de description) ainsi qu'une décidabilité assurant la terminaison des calculs dans un temps fini. Néanmoins, il ne peut être utilisé qu'avec certaines restrictions. Par exemple, une classe ne peut pas être une instance d'une autre classes.

3. Le langage OWL-FULL permettant le plus haut niveau d'expressivité ainsi que la possibilité d'étendre le vocabulaire d'OWL. Néanmoins, il ne permet pas de garantir la complétude et la décidabilité des calculs.

— **DAML+OIL** est un langage désigné à décrire la structure d'un domaine à travers une approche orientée objet. Ainsi la structure est décrite en terme de *classes* et *propriétés*. Une ontologie décrite par DAML+OIL est constituée d'un ensemble d'axiomes fournissant des relations de subsomption entre *classes* ou *propriétés*. Le langage DAML+OIL est une extension du langage RDF offrant plus d'expressivité relativement aux exigences du Web sémantique. Plus précisément, DAML+OIL est la combinaison de deux langages construit séparément à savoir le langage DAML[1] et le langage OIL. Il permet de modéliser les aspects suivants [73] :
  — définition de classes de propriétés ;
  — définition de classes de ressources ;
  — relations logiques entre classes (disjonction, union, équivalence, etc.) ;
  — relations d'héritage entre classes ;
  — restriction de propriétés (cardinalité, etc.) et typage ;
  — prise en charge des collections (listes) ;
  — instanciation de classes de propriétés et de ressources.

5.7 Apport des ontologies dans la catégorisation de textes

Plusieurs recherches sur la catégorisation de textes se sont focalisées sur l'utilisation des ontologies dans le but de proposer de nouvelles approches sémantiques. En effet, avant l'apparition des ontologies, la quasi-totalité des approches existantes se basaient sur des modèles probabilistes s'appuyant sur les occurrences des mots (ou lemme). Plus précisément, il y a plus d'une vingtaine d'années que W. Croft a abordé dans [26] la

1. Darpa Agent Markup Language

nécessité du passage vers les approches sémantiques : *"The statistical approach has many advantages and can achieve a reasonable level of effectiveness with tech- niques that are very efficient. However, it appears that to achieve significant improvements in retrieval effective- ness compared to current techniques, systems must be designed to acquire and use explicit domain knowledge."*

L'utilisation des ontologies été sollicitée dans de nombreux travaux en phase de représentation afin de proposer de nouvelles approches de représentation basées sur les concepts [15, 69, 107, 138, 122]. Bloehdorn et Hotho ont proposé dans [15] une nouvelle approche de représentation basée sur l'extraction des concepts à partir d'ontologie. Les auteurs ont expérimenté leur approche en utilisant les ontologies WordNet et Mesh (Medical Subject Headings) et affirment avoir obtenu une amélioration de 6.8%. Mladenic et Globelnik ont proposé dans [107] une approche de mapping automatique des pages Web en utilisant l'ontologie Yahoo ![81] dont l'objectif est d'associer un classifieur à chaque catégorie de la hiérarchie yahoo !. Wu et al. ont proposé dans [165] une approche de catégorisation de textes basée sur la construction automatique d'ontologies de domaines à partir de règles morphologiques et méthodes statistiques. Plus précisément, il s'agit d'extraire les concepts à partir des documents étiquetés afin de construire une ontologie représentative du domaine. Les documents à catégoriser seront représentés par les concepts de l'ontologie construite. Les auteurs affirment avoir obtenu de bons résultats malgré la construction automatique de l'ontologie du domaine sans aucune intervention humaine.

5.8 Conclusion

Dans ce chapitre, nous nous somme intéressés aux ontologies comme moyen indispensable de représentation des connaissances permettant ainsi la facilité de partage et de réutilisation de ces connaissances. Après avoir abordé la notion d'ontologie du point

de vue de la communauté Ingénierie de connaissances en présentant les définitions les plus populaires, nous avons mis l'accent sur la définition formelle orientée vers le domaine de la catégorisation de texte proposée par Hotho et Staab dans [69]. Nous avons constaté que la manière de gérer les sources d'informations est un facteur majeur dans le processus de construction d'ontologies. De ce fait, il étais important de discuter les différents schémas de construction d'ontologies en citant les avantages et les inconvénients de chaque schéma. Le processus de développement des ontologie dépend crucialement des langages utilisés pour leurs implémentation. Ces langages se différencient selon leurs niveau d'expressivité ainsi que les paradigmes de représentation dont ils se basent. Les langages de représentation d'ontologies sont présentés dans la section 5.6. Dans la dernière section, nous avons présenté les travaux utilisant les ontologies dans le domaine de la catégorisation de textes.

Chapitre 6

Catégorisation de textes multilingues

6.1 Introduction

Le traitement de données multilingues sur le web a connu cette dernière décennie une extrême importance de la part des chercheurs. En effet, les utilisateurs du web ne se contente plus de retrouver l'information désirée dans leur langue maternelle, ils recherchent plutôt l'information pertinente quelle que soient la langue et la forme de stockage. Ce besoin s'est justifié par le recul de la domination de l'Anglais sur le web qui devient de plus en plus multilingue. De même, le nombre d'utilisateurs non-anglophone du web connait une progression constante. Avec ce phénomène de multilinguisme, on se retrouve avec des documents de différentes langues partageant les mêmes catégories. Une solution de les catégoriser consiste à appliquer plusieurs catégorisations monolingues à savoir une catégorisation monolingue pour chaque langue. Néanmoins, cette solution nécessite la présence d'un corpus d'apprentissage pour chaque langue. De plus, les documents d'une langue ne seront utiles que pour catégoriser les documents en provenance

de cette même langue. Ces raisons ont donné naissance à un nouveau domaine qui est bien la *catégorisation de textes multilingues*. Dans ce chapitre, nous allons présenter ce nouveau domaine en définissant tout d'abord le processus de catégorisation multilingue avec ces différents scénarios puis en citant les travaux effectués dans ce domaine.

6.2 Définition

La catégorisation des textes multilingue (C.T.M) consiste à catégoriser un texte rédigé dans une langue donnée, à partir d'un modèle de prédiction construit sur une base d'apprentissage dans une ou plusieurs langues cibles. En effet il s'agit de savoir comment catégoriser un document en utilisant des documents d'autres langues. La catégorisation des textes multilingues se rapporte à l'attribution des documents basés sur leurs contenus, à une ou plusieurs catégories prédéfinies. L'objectif de la C.T.M est de construire des systèmes capables de catégoriser des documents en provenance de différentes langues en se basant sur une base documentaire contenant des documents d'autres langues. Par exemple, catégoriser un document en Espagnol en utilisant une base documentaire formée des documents en Anglais et Français. Ce phénomène de multilinguisme nous amènent à se poser les questions suivantes :

— Le multilinguisme concernera la base d'apprentissage à savoir les documents sur lesquels sera effectué l'apprentissage et/ou la base de test à savoir les documents à catégoriser.

— S'agit-il de construire un seul modèle prenant en considération toutes les langues prises en considération ou de construire plusieurs modèles à savoir un modèle pour chaque langue.

Ces questions permettent de distinguer trois types de C.T.M [124] :

6.2.1 Catégorisation des textes par multiples langues

Dans ce cas, Il s'agit de construire un seul modèle de prédiction à partir d'un corpus d'apprentissage contenant des documents étiquetés dans différentes langues. Vu la disponibilité des documents étiquetés dans les différentes langues, Ce scénario exclu l'utilisation des techniques de traduction ce qui permettra d'éviter toute perte d'information. Cependant, plusieurs inconvénients se présentent :

— La nécessité de disposer d'un ensemble de documents étiquetés dans chaque langue ; ce qui est extrêmement difficile pour les langues peu présentes sur le web.

— La grande dimensionnalité engendrée par l'union des vocabulaires des différentes langues.

— La difficulté de la sélection des descripteurs causée par la coexistence de plusieurs langues.

6.2.2 Catégorisation des textes par croisement de langues

Dans La catégorisation des textes par croisement de langues, dite en anglais Cross-Language Text Categorization (CLTC), un ensemble de documents étiquetés est disponible dans une seule langue L_1. Cet ensemble est utilisé pour construire un modèle de prédiction afin de catégoriser des documents non étiquetés exprimés dans une autre langue L_2. Pour cela, il est nécessaire d'utiliser la traduction automatique. Deux manières différentes de traduction peuvent être employées.

— Traduction des documents étiquetés : les documents étiquetés sont traduits dans la langue des documents non étiquetés afin de construire le modèle de prédiction.

— Traduction des documents à classer : Dans ce cas, c'est les documents non étiquetés qui sont traduit vers la langue des documents étiquetés. Le modèle de prédiction est donc construit en utilisant des documents non traduit.

6.2.3 Catégorisation des textes avec la langue universelle

Ce scénario utilise une langue de référence universelle à laquelle tous les documents sont traduits. Cette langue devrait contenir toutes les propriétés des langues et doit être organisée d'une façon sémantique : les mots indiquant les mêmes concepts dans les langues devraient être traduits aux mêmes termes dans la langue universelle.

6.3 Pourquoi la C.T.M ?

Les recherches dans le domaine de la catégorisation multilingues ont connu ces dernières années une grande importance. Plusieurs raisons ont motivé la recherche dans ce domaine. Nous pouvons énumérés les raisons suivantes :

— **La Disponibilité des collections multilingues :** Grâce à la révolution qu'a connu les réseaux en terme de vitesse de transmission et l'arrivée du WEB. L'utilisateur se retrouve face à une gigantesque masse d'information dans différentes langues. l'intérêt de l'utilisateur n'est plus d'accéder à l'information dans sa langue maternelle mais de retrouver l'information pertinente quelle que soient la langue et la forme de stockage.

— **Le recul de la domination de l'Anglais :** A l'arrivée du WEB, la quasi totalité des documents accessibles sur le net ont été rédigé en Anglais. En effet,vu que le WEB est une invention américaine. il était tout à fait normal qu'il soit largement et rapidement utilisé par les pays anglophones. Depuis, cette domination de l'Anglais à connu un recul pour ouvrir la voie vers un réseau mondial multilingue. Des statistiques menées par G.Numberg sur 2.5 millions de pages ont montré que 85% de leurs contenu est rédigé en Anglais. Des statistiques menées par ExciteHome sur la langue de plusieurs centaines de millions de pages ont révélé que seulement 72% des pages sont en Anglais [50].

— **Augmentation de la population non-anglophone du Web :** Même si la

population anglophone du web reste la plus majoritaire, son pourcentage par rapport à la population non-anglophone se réduit de plus en plus. Les statistiques sur le nombre d'utilisateurs d'Internet en décembre 2013 (voir figure 6.1) montre que la population anglophone de l'Internet ne représente désormais que 28.6% alors qu'elle représenté 36.5% en 2003.

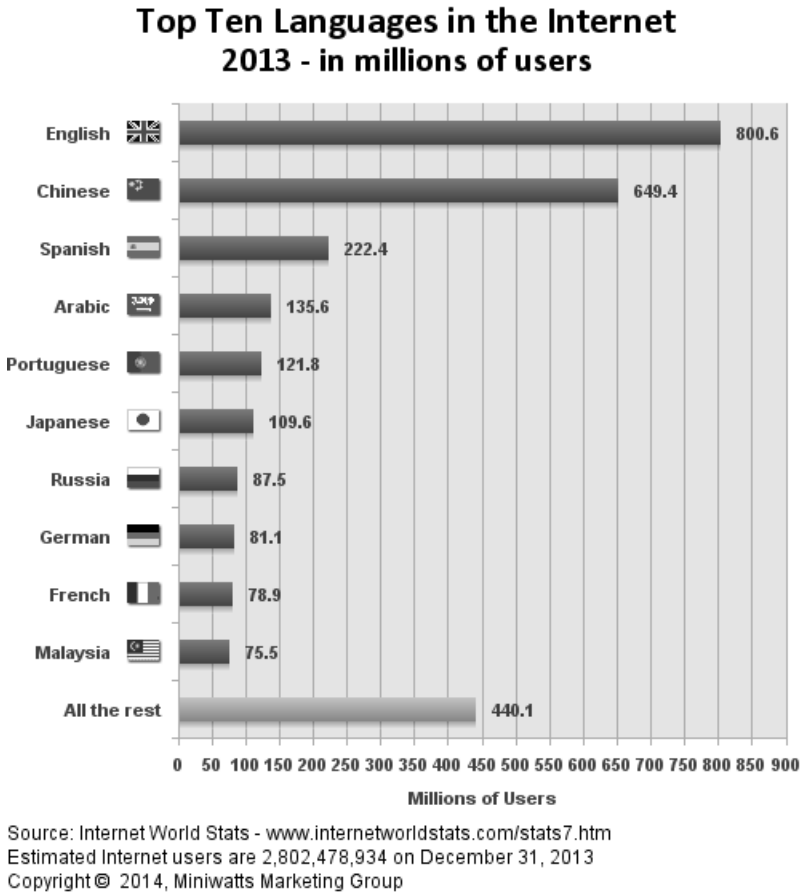


FIGURE 6.1 – *La répartition de la population du Web par langue*

6.4 Travaux connexes

Étant donnée que le domaine de la catégorisation multilingue est relativement récent, peu de travaux peuvent être recensés. En effet, la majorité des travaux proviennent essentiellement du domaine de la recherche d'information multilingue. Nous pouvons

distingué quatre familles d'approches à savoir les approches basées sur la traduction automatique, les approches basées sur les dictionnaires et corpus, les approches basées sur l'adaptation de domaine et les approches basées sur les ressources sémantiques .

6.4.1 Approches basées sur la traduction automatique

La majorité des approches proposées pour la catégorisation multilingue se basent sur la traduction automatique du corpus d'apprentissage et/ou le corpus de test. Ces approche nécessitent l'utilisation des traducteurs automatiques dans le système. Il est important ici de signaler la différence qui existe entre l'objectif des systèmes multilingues et l'objectif des traducteurs automatiques. En effet, si les systèmes multilingues s'intéressent à la notion de similarité entre documents, les traducteurs automatiques sont orientés vers la lisibilité et fiabilité de la traduction fournie. Dans le but de garantir une bonne traduction,il est nécessaire d'identifier le sens du mot avant de pouvoir le traduire. Cette identification de sens exige d'éliminer certains types d'ambiguïté dont les plus importants sont :

- **Polysémie :**Un même terme peut avoir différents sens. Le terme anglais "plant" illustre bien ce problème car il possède au moins 3 sens différents (la plante, l'installation technique, le coup monté). Par contre, "power plant" n'a qu'un seul sens. Donc, si l'on tient compte des termes voisins au terme ambigu, c'est-à-dire son contexte, on arrive à déterminer son sens exact.
- **L'homographie :** Deux mots différents s'écrivent de la même façon. Par exemple, "*livre*" est soit la conjugaison du verbe "*livrer*", soit le nom synonyme d'"*ouvrage*". La catégorie lexicale du terme (nom, verbe, adjectif) permet de lever cette ambiguïté de sens.
- **Le sens large :** un terme qui a un sens très large, (exemple : "*air*") peut prendre un sens particulier dans certain domaine ("*air bag* "). Il se peut que dans une autre langue, un concept particulier résultant de l'association de termes généraux, dans un syntagme nominal (SN), soit décrit par une toute autre combinaison de termes.

"*Air bag*" se traduit par "*sac gonflable de protection*". Pour résoudre ce problème, il faut identifier la totalité de l'expression, parmi les composantes du syntagme nominal pour déterminer le concept spécifique qui se cache derrière la combinaison des termes.

Jalam propose dans [74] trois schémas pour l'utilisation de la traduction automatique dans la catégorisation multilingue. Le premier schéma appelé "trivial" est une extension directe de la catégorisation monolingue. Ce schéma consiste à construire un modèle de prédiction sur chacun des L corpus (langues) pris en considération par le système. Ainsi, il faut identifier la langue de chaque document afin de pouvoir lui appliquer le modèle correspondant à cette langue. Si le texte est identifié comme « intéressant » il sera traduit vers la langue cible. L'avantage avec ce schéma est l'intervention de la traduction à la fin de processus : le traducteur n'intervient pas dans l'apprentissage et, par conséquent, aucune distorsion d'information ni perte n'est commise à cette étape. L'inconvénient majeur de ce schéma est qu'il exige de faire $|L|$ apprentissages, un par langue, et ceci suppose d'avoir des quantités suffisantes de textes étiquetés dans chaque langue et dans chaque classe. Ceci est difficile surtout pour les langues peu présentes sur le Web. Dans le deuxième schéma, le modèle de prédiction est construit à partir d'un corpus d'apprentissage d'une seule langue. Ainsi, le document à catégoriser doit être traduit dans la langue d'apprentissage (la langue du modèle de prédiction) afin de pouvoir lui appliquer le modèle de prédiction. Ce schéma a comme avantage la construction d'un seul modèle de prédiction mais son inconvénient majeur est le rôle primordial dont joue le traducteur dans la phase de classement. Le troisième schéma utilise la traduction automatique dans les deux phases , apprentissage et classification. Ainsi, les documents d'apprentissage sont traduit vers une langue cible et par conséquent un seul modèle de prédiction est construit. Chaque document à catégoriser sera traduit vers la langue cible afin de pouvoir lui appliquer le modèle de prédiction.

L'utilisation des traducteurs automatiques dans la catégorisation multilingue a été exploré dans [40, 110]. Les expérimentations ont montré que les performances dépendent crucialement de la qualité des traducteurs utilisés. Shi et al. proposent dans [140] une

approche pour palier au problème d'ambiguïté dans la traduction automatique en sélectionnant parmi les traductions possibles d'un mot, celle qui correspond au contexte des documents de l'autre langue (la langue cible). Pour cela, les auteurs utilisent l'algorithme d'espérance-maximisation(Expectation-maximisation)[31] pour estimer les probabilités associés aux différentes traductions possibles du mot. Les résultats des expérimentations ont montré l'efficacité de l'approche proposée par rapport aux approches de traduction classiques.

Les techniques de traduction automatique ont été aussi utilisées dans La recherche d'information multilingue pour traduire les requêtes des utilisateurs vers la langue des documents. Les performances n'ont pas été assez satisfaisant. En effet, vu que les requêtes sont souvent exprimée sous forme d'une liste de mots dépourvue de sémantique, les traducteurs automatiques ne produiront pas forcement de bonnes traductions.

6.4.2 Approches basées sur les dictionnaires et les corpus

Les dictionnaires et les corpus alignés se différentient par rapport aux traducteurs automatique par le fait qu'ils se basent sur une traduction mot à mot sans prendre en considération leurs catégories syntaxiques. Les dictionnaires se basent sur l'alignement entre deux listes de mots. Ainsi, La traduction basée sur ces dictionnaires fournit en sortie les traductions d'un terme donné en entrée. L'utilisation des dictionnaires dans la recherche d'information multilingue à montrer leurs insuffisances à fournir de bonnes traductions. Cela revient essentiellement à l'incapacité de couvrir la totalité des mots de la langue ainsi que l'absence des mots techniques. Plusieurs travaux [71, 28, 8] ont montré que l'utilisation des dictionnaires comme seul moyen de traduction ne produisaient que la moitié des performances du cas monolingue. A la différence des dictionnaires qui se basent sur l'alignement entre deux listes de mots, Les corpus permettent de réaliser la traduction en se basant sur l'alignement entre deux ensembles de documents pour deux langues différentes. On distingue deux types de corpus qui se différentient

selon le type d'alignement, à savoir les corpus parallèles et les corpus comparables. La construction des corpus parallèles se base sur la mise en correspondance de chaque document d'une langue source L_1 avec le document résultant de sa traduction dans la langue cible L_2. Les corpus comparables quant a eux, se composent de deux ensembles

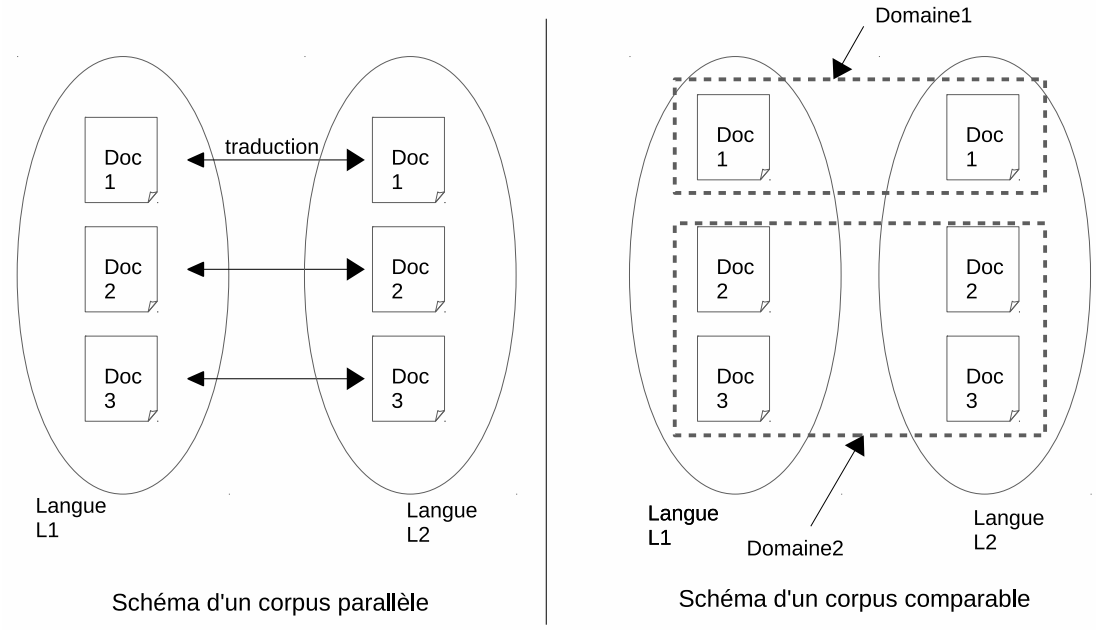


FIGURE 6.2 – *La différences entre corpus parallèle et corpus comparable*

de documents pour deux langues différentes, qui ne sont pas parallèles au sens strict du terme, mais qui contiennent des informations similaires selon un certain critère (comme par exemple les actualités journalières d'une même période, d'un même évènement , d'un même domaine, etc.). Les corpus parallèles ont été utilisé dans la recherche d'information multilingue pour relier les termes d'une langue à ceux d'autres langues d'une manière automatique. Litmann et al. proposent dans [95] une méthode basée sur la technique LSI afin de représenter les documents dans un espace vectoriel réduit prenant en considération les termes des deux langues (ceux du corpus parallèle). Les résultats ont montré l'utilité de la méthode proposée pour la traduction des requètes. Wu et al. ont proposé dans [164] une nouvelle approche pour la catégorisation par croisement de langues se basant sur l'utilisant des corpus parallèles pour la génération automatique des dictionnaires bilingues. L'approche intitulée CLNBC (Cross Language Naive Bayes

Classifier) consiste à :

— Utiliser le modèle probabiliste pour générer un dictionnaire bilingue probabiliste à partir d'un corpus parallèle en calculant la probabilité conditionnelle d'occurrence d'un mot w_e d'une langue e sachant un document d d'une langue f.

$$P(w_e|d) = \sum_{w_f \in d} P(w_e|w_f)P(w_f|d) \tag{6.1}$$

le dictionnaire bilingue ainsi construit est censé fournir la meilleur traduction d'un mot w_e dans la langue f.

— Appliquer un classifieur bayésien afin de prédire la catégorie d'un document.

Les auteurs ont expérimenté l'approche en générant un dictionnaire bilingue Anglais-Chinois à partir d'un corpus parallèle Anglais-Chinois contenant 276,889 paires de traduction et affirment avoir obtenu des résultats proches de la catégorisation monolingue. Gliozzo et Strapparava proposent dans [52] une approche qui consiste à utiliser les corpus comparables pour la construction d'un modèle multilingue de domaine, afin de définir par la suite une fonction de similarité générale entre les documents de différentes langues dont la formule est la suivantes :

$$f(x) = \sum_{i=1}^{n} \lambda_i K(x_i, x) + \lambda_0 \tag{6.2}$$

où :

— λ_i représente le poids associé au document x_i

— $K(x_i, x)$ représente la fonction noyau utilisée.

Cette fonction est utilisée par un classifieur SVM afin d'associer le document à la classe ayant le plus proche vecteur. La construction du modèle multilingue est le résultat de la décomposition en valeurs singulières de la matrice $termes * documents$ fusionnant les deux matrices $termes * documents$ des corpus d'apprentissage des langues prises en considération. L'approche a été évalué en utilisant un corpus comparable formé de 32354 documents en Italien et 27821 documents en Anglais, répartis en quatre catégories. Les auteurs ont constaté que les performances avoisinent les performances monolingues en

fonction du nombre de documents dédiés pour l'apprentissage. Ainsi, plus le corpus d'apprentissage est consistant et plus les performances s'améliorent.

6.4.3 Approches basées sur l'adaptation de domaine

La majorité des approches basées sur l'utilisation des traducteurs automatiques se basent sur la traduction des documents d'une langue source vers une langue cible dans le but de transformer le problème multilingue en un problème monolingue. Les performances de ces approches se heurtent aux problème de différences entre les langues et les cultures du fait qu'ils utilisent les mots comme moyen pour transférer la connaissance d'une langue à une autre. En effet, même si un mot se répète fréquemment dans une langue, sa traduction peut être rarement utilisée dans d'autres langues. Ainsi, l'utilisation de la traduction automatique comme seul moyen risque de générer une distribution divergente des données entre les documents traduits de la langue source et les documents originaux de la langue cible. Une solution à ce problème consiste à utiliser l'adaptation de domaine pour assister les traducteurs automatiques [116, 14, 53, 160]. L'adaptation de domaine (Domaine Adaptation) se réfère à l'adaptation d'un classifieur construit à partir d'une source de données relative à un domaine (appelée domaine source) aux données en provenance d'une source relative à un autre domaine (appelée domaine cible). En d'autre termes, il s'agit de classer des documents d'un domaine cible en utilisant un classifieur construit préalablement sur des documents étiquetés d'un autre domaine source. Les techniques d'adaptation peuvent être utilisé dans le cas de la catégorisation multilingue en considérant chaque langue comme un domaine à part. Prettenhoffer et Stein ont proposé dans [116] une approche d'adaptation intitulé CL-SCL (Cross-Language Structural Correspondence Learning) qui consiste à introduire des descripteurs indépendant de la langue. En premier lieu, il s'agit de sélectionner les mots les plus discriminants à partir des documents étiquetés. Ces mots sont traduits vers la langue cible (la langue des documents à étiqueter) formant ainsi des paires de mots appelés *pivot*. Chaque paire-pivot associe un mot parmi les mots les plus discriminant à

sa traduction dans la langue cible. En deuxième lieu, il s'agit de modéliser la corrélation entre les paires-pivots et les mots les moins discriminant (ceux qui ne figurent pas dans les paires-pivots). L'avantage de cette méthode est de réduire les risques de mauvaises traductions en ne traduisant que les mots les plus discriminant. Les auteurs ont expérimenté l'approche sur un corpus multilingue (Anglais, Allemand, Français, Japonais) en prenant l'Anglais comme langue source et affirment la compétitivité de leur approche avec les approches classique de traduction.

6.4.4 Approches basées sur les ressources sémantiques

A la différences des familles d'approches décrites précédemment qui tentent de résoudre le problème du multilinguisme par des méthodes statistiques, d'autres approches se basent sur la mise en correspondance des langues par le biais des ressources sémantiques associées à ces langues. Parmi les ressources sémantiques les plus sollicitées figurent les thésaurus et ontologies multilingues. Ces ressources mettent en correspondance les termes de différentes langues en les regroupant dans des classes. Ces regroupement forment les entrées avec lesquelles seront représentés les documents. Les thésaurus multilingues ont été utilisés en premier lieu par G. Salton dans [132] pour représenter les documents et les requêtes dans le domaine de la recherche d'information par croisement de langues (CLIR). Les auteurs ont utilisé un thésaurus Anglais-Allemand pour représenter les documents exprimés en Anglais et les requètes exprimées en Allemand. Les expérimentations ont montré que les performances du cas multilingue (Anglais-Allemand) avoisinait les performances du cas monolingue. Yang et al. ont proposé dans [161] une approche pour le cas de la catégorisation multilingue par multiple langues qui consiste en trois principales phases :

— Construction de thésaurus bilingue en utilisant la technique d'analyse de cooccurrence généralement utilisée dans la recherche d'information par croisement de langue (dite en anglais Cross Language Information Retrieval) et CLTC.

— Apprentissage de la catégorisation en tenant compte non seulement des documents

pré classifiés en une langue L_1 mais également des documents pré classifiés en une autre langue L_2 et en utilisant aussi le thésaurus bilingue construit.

— Assignation de la catégorie pour chaque document non classifié dans L_1 ou L_2 en utilisant le modèle correspondant de catégorisation des textes induit précédemment.

Selon la langue utilisée dans le document non classifié, il nécessite d'employer la méthode respective d'extraction d'attributs pour extraire des attributs à partir du document non classifié. En conclusion, le vecteur de document est employé pour déterminer une catégorie appropriée sur la base du modèle correspondant de catégorisation des textes. Siersdorfer et de Melo ont proposé dans [29] une nouvelle approche qui consiste à utiliser les ontologies afin de générer une représentation conceptuelle. Les auteurs utilisent une nouvelle technique de mapping appelée OMR (Ontology region mapping) qui consiste à associer des termes , non pas à des concepts isolés, mais plutôt à des régions entières de l'ontologie. Les expérimentations ont montré que les performance de l'approche proposée surpasse les résultats de la représentation *sac de mots*. Litvak et al. ont proposé dans [99] une nouvelle méthodologie basée sur l'utilisation des ontologies de domaine pour la représentation conceptuelle des documents du web. Les auteurs ont évalué la méthodologie proposée en utilisant une ontologie OWL relative au domaine des produits chimiques et affirment avoir obtenu d'assez bon résultats.

6.5 Tableau comparatif

Le tableau 6.1 présente une étude comparative entre les différentes approches décrites dans ce chapitre.

	Type de CTM	Resources utilisées	type de descipteurs	langues supportées	Classifieurs utilisé
Jalam[74]	PLTC	Traducteur Babelfish	N-grammes et Mots	Anglais, Allemnad et Français	K-NN, C4.5
Wu el al[164]	CLTC	Corpus paralèlle et Traducteur SystranPremium 5.0	Mots	Anglais et Chinois	SVM, NAIVE BAYES et LSI
Gliozo et Straparrava [52]	CLTC	Corpus comparable	Mots	Anglais et Italien	SVM et LSA
Prettenhoffer et Stein [116]	PLTC	Traducteur Google	Mots	Anglais, Allemand, Français et Japonais	CL-SCL
Yang et al. [161]	CLTC	Thésaurus bilingue	Mots	Anglais et Chinois	NAIVE BAYES et SVM
Siersdorfer et de Melo [29]	CLTC	Ontologies (WordNet2.1+Spanish WordNet) et traducteur Babelfish	Concept	Anglais et Espagnol	SVM
Litvak et al. [99]	CLTC	Ontologie de domaine	Concept et Mots	Anglais, Arabe, Russe et Hébreu	C4.5, BAYES NETWORK et NAIVE BAYES

TABLE 6.1: Tableau comparative d'un ensemble de méthode pour la catégorisation de textes multilingues

6.6 Conclusion

Ce chapitre a porté essentiellement sur le domaine de la catégorisation de textes multilingues. L'objectif de ce domaine est d'adapter le processus de catégorisation de textes au phénomène du multilinguisme qui envahi de plus en plus le Web, créant ainsi chez l'utilisateur le besoin d'explorer les informations disponibles dans d'autres langues. Une solution plus simple à ce problème consiste à transformer le problème en plusieurs catégorisations monolingues par la construction d'un classifieur pour chaque langue. Néanmoins, cette solution n'est plus envisageable pour les langues diminuées ou peu présentes dans le Web. La majorité des recherches se sont orientées vers des solutions basées sur le croisement de langues. Ainsi, La question principale est la suivante :"*Comment utiliser des documents étiquetés d'une certaine langue pour catégoriser des documents exprimés dans d'autres langues ?*".

Une fois avoir introduit la notion de multilinguisme, nous avons défini le problème de la catégorisation multilingue en précisant les différents types, puis nous avons enchainé avec une classification des différentes approches proposées dans ce domaine tout en précisant les avantages et les inconvénients de chaque famille d'approches. Suite à une synthèse de ces différentes approches, nous avons remarqué que la majorité de ces approches se basent directement ou indirectement sur la traduction comme moyen pour résoudre le problème. Plusieurs approches se définissent comme des approches n'utilisant pas la traduction mais on les analysant, on s'aperçoit qu'ils se basent sur des ressources construites en utilisant la traduction tels que les corpus parallèles. D'autres approches excluent réellement la traduction du processus de la catégorisation en utilisant les corpus comparables comme moyen pour la mise en correspondance entre les langues. Néanmoins, ces approches se heurtent au problème de l'indisponibilité des corpus pour les langues diminuées.

Partant de ce principe, nous avons proposé deux approches pour la catégorisation

multilingue par croisement de langue. Similairement aux approches basées sur les ressources sémantiques [29, 99], ces deux approches se basent sur l'utilisation des ontologies. Dans La première approche, nous utilisons une seule ontologie monolingue comme moyen pour réduire le bruit que peut générer une mauvaise traduction. Contrairement à la première approche, la deuxième approche se base sur l'alignement entre les ontologies dans le but d'écarter la traduction du processus de la catégorisation. Le chapitre suivant illustre les deux approches proposées.

Chapitre 7

Approches proposées et expérimentations

7.1 Introduction

Durant cette dernière décennie, plusieurs approches ont été proposées dans le cadre de la Catégorisation de Textes. La majorité de ces approches se basent sur la représentation "Sac de Mots" qui consiste à représenter chaque texte sous forme d'un vecteur dont chaque composante représente un mot. C'est vrai que le mot possède l'avantage d'avoir un sens explicite, néanmoins cette représentation souffre de plusieurs inconvénients :

— Elle ne prend pas en considération les unités multi-mots en les considérant indépendant les uns aux autres, ce qui n'est pas le cas en réalité car personne ne peut contredire le fait que les mots «union» et "européenne" ensemble ont une sémantique différente que lorsqu'ils sont pris séparément.

— Elle ne tient pas compte de la relation de synonymie entre les mots. De ce fait, des mots tels que «aimer» et «adorer » seront considérés comment différents alors qu'ils sont synonymes.

— Elle souffre d'un manque de généralisation. En effet, il n'y a aucune possibilité de généraliser les mots spécifique «gold» et «silver» sous le mot général «precious metal».

Afin d'y remédier aux différents problèmes de la représentation "sac de mots", la majorité des recherches se sont orientées vers l'utilisation des ontologies dans le but de proposer de nouvelles méthodes de représentations en remplaçant les mots par leurs sens[54, 59, 112, 106].

Le besoin de traitement de documents multilingues se fait de plus en plus ressentir. Cela est dû à :

— La disponibilité des collections multilingues sur le Web a créé chez l'utilisateur, le besoin de retrouver ou traiter l'information quelque soit la langue utilisée.

— Le recul de la domination de l'Anglais comme langue maternelle des utilisateurs du Web.

— L'apparition de plusieurs zones d'union dans le monde comme conséquence au phénomène de la globalisation a motivé le lancement de plusieurs projets multilingues tels que le projet EMIR (European Multilingual Information Retrieval) utilisant le système SPIRIT [120]et le projet MEDLIB (Mediterranean Digital Library) [121].

Notre travail consiste à étendre l'utilisation des ontologies dans la catégorisation de textes monolingues pour catégoriser des textes multilingues. Nous avons proposé deux approches de type CLTC (Catégorisation de textes par croisement de langues). Ainsi, il s'agit de catégoriser un texte d'une langue L_2 en utilisant un modèle de prédiction construit à partir d'un ensemble de textes étiquetés dans une autre langue L_1. Sachant que chaque langue possède son propre vocabulaire, il devient nécessaire d'unifier le vocabulaire utilisé pour la représentation des textes des deux langues. La première approche se base sur une hybridation entre l'utilisation des ontologies et les techniques de traduction. Cette hybridation offre les avantages suivants :

— Sans l'utilisation des techniques de traduction, il devient nécessaire d'avoir une ontologie spécifique à chaque langue.

— L'utilisation d'une ontologie riche permet de minimiser les erreurs de traduction à travers l'usage des relations sémantiques.

La deuxième approche exclut l'utilisation directe des techniques de traduction en incorporant une ontologie pour chaque langue. Cette deuxième approche permet d'éviter toute perte ou distorsion d'information causée par les techniques de traduction. Néanmoins, cela nécessite un mapping entre les différentes ontologies utilisées. Dans ce chapitre, nous allons détailler les approches proposées puis nous présentons nos expérimentations ainsi qu'une comparaison entre les différentes approches.

7.2 Première Approche

Comme illustré dans la figure 7.1, notre première approche se décompose en trois phases :

— phase de représentation qui consiste à traduire les textes à étiqueter vers la langue des documents étiquetés afin de les représenter par la même méthode de représentation.
— Phase d'apprentissage qui consiste à créer les profils conceptuels des catégories.
— Phase de classification qui consiste à pondérer les vecteur construits afin de pouvoir calculer par la suite la distance entre le vecteur du texte à étiqueter et les profils conceptuels des catégories.

7.2.1 Phase de représentation

Le premier problème à résoudre dans une catégorisation de textes est : Comment représenter les textes afin de faciliter les traitements mais surtout de ne retenir que l'information utiles pour la catégorisation ?. La méthode de représentation la plus largement utilisée dans la C.T est la méthode " Sac de mots " qui utilise l'ensemble des mots avec leurs fréquences pour représenter les documents et catégories. Néanmoins,

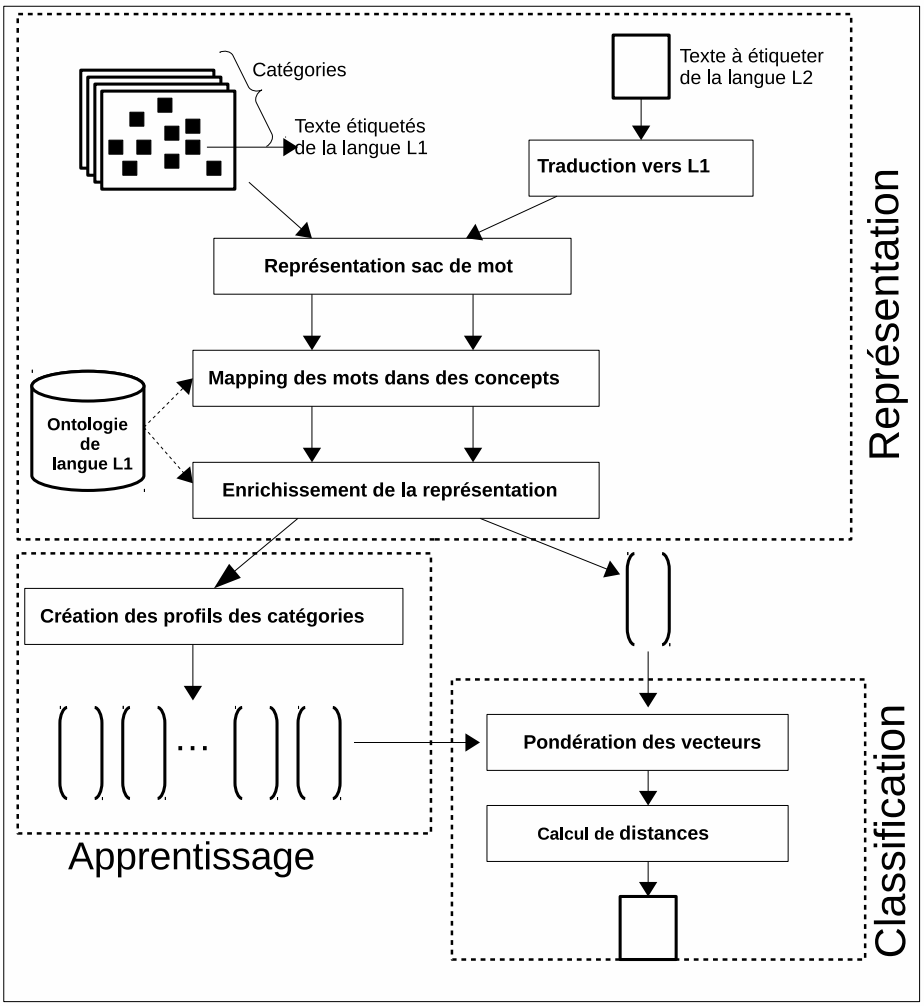


FIGURE 7.1 – *L'architecture de la première approche*

cette représentation ne peut être utilisée pour représenter des textes en provenance de différentes langues.

Dans notre approche, nous représentons les textes par une représentation conceptuelle enrichie nécessitant les étapes suivantes pour chaque texte à représenter :

1. **Tokenisation :** Cette première étape consiste à convertir le texte en un ensemble de mots. Cette étape permet de reconnaitre les espaces de séparation des mots, les chiffres et les ponctuations.
2. **Élimination des mots vides :** Après avoir convertit le texte en un vecteur de

mots, cette étape consiste à ne garder que les mots significatifs en éliminant les mots vides (pronoms personnels, prépositions,....).

3. **Mapping des mots en concepts :** cette étape consiste à remplacer chaque mot par le concept approprié a son sens à partir de l'ontologie de la langue du texte a représenter. Ce mapping permettra de réduire la dimensionnalité de l'espace de représentation. En effet, tout les mots dénotant un sens seront remplacés par le même concept. Comme indiqué dans l'exemple illustartif de la figure 7.2, le mapping a permis de réduire la dimensionnalité en passant de 8 mots à seulement 4 concepts. les trois mots "Government", "Authorities" et "regime" sont mappés dans le même concept.

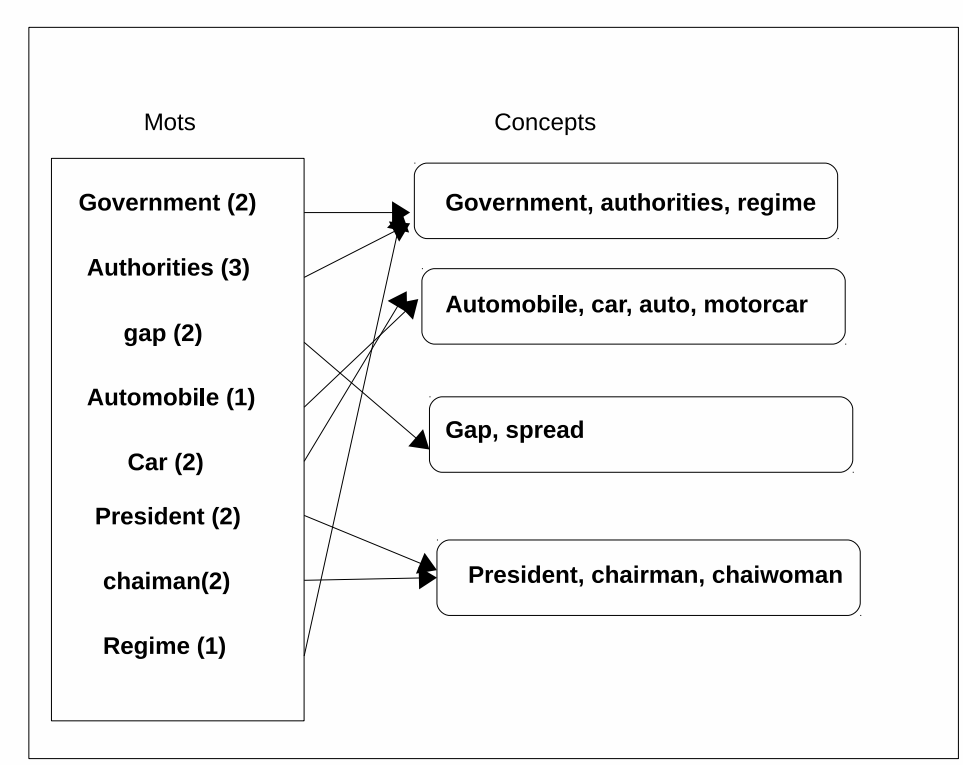


FIGURE 7.2 – *Exemple d'un mapping de mots en concepts*

4. **Désambiguïsation du mapping :** Il est tout à fait claire que le processus de mapping des mots dans leurs concepts relatives est ambiguë, car un mot peut avoir plusieurs sens. C'est pour cette raison là qu'il est nécessaire d'utiliser une

stratégie de désambiguïsation afin de trouver le concept le plus approprié à un mot. Dans notre travail, nous utilisons les deux stratégies de désambiguïsation suivantes :

- **Tout les Concepts :** Cette stratégie est la plus simple car elle consiste tout simplement à ignorer le problème en prenant tous les concepts proposés par l'ontologie comme concepts appropriés dont le but d'augmenter l'espace de représentation. Les fréquences de concepts seront calculées comme suit :

  $$cf(d, c) = tf\{d, \{t \in T | c \in ref_c(t)\}\}.$$

  où $ref_c(t)$ représente l'ensemble des concepts associés au terme t.
- **Premier Concept :** Dans le cas ou l'ontologie utilisée donne pour chaque mot une liste ordonnée de concepts selon un certain critère. Cette stratégie de désambiguation consiste à prendre seulement le premier concept de la liste comme le concept le plus approprié. Les fréquences de concepts seront donc calculés comme suit :

  $$cf(d, c) = tf\{d, \{t \in T | first(ref_c(t)) = c\}\}.$$

  où $first(ref_c(t))$ est le premier élément de l'ensemble ordonné des concepts associés au terme t.

5. **Enrichissement de la représentation :** Après avoir effectuer la stratégie de mapping ainsi que celle de la désambiguïsation, cette étape consiste à augmenter ou enrichir l'espace de représentation en prenant en considération la relation de subsomption entre les concepts de l'ontologie. Les fréquences des concepts seront mise à jour comme suit : $cf(d, c) = \sum_{b \in H(c)} cf(d, b)$, où $H(c)$ représente l'ensemble des concepts subsumant le concept c.

L'application de la phase de représentation sur le corpus d'apprentissage donnera comme résultat une matrice $documents * concepts$ représentant le nombre d'occurrences de chaque concept dans chaque document. Pour le corpus de test, il faut tout d'abord traduire chaque document vers la langue du corpus d'apprentissage afin de pouvoir lui appliquer la phase de représentation. Les étapes de la phase de représentation sont décrites dans l'algorithme 1.

Algorithme 1 : Phase de représentation

Entrées :
Un ensemble T de documents
Ontologie O monolingue utilisée
Sorties :
Matrice Cf de contingence (documents * concepts)
Dictionnaire de concepts
Debut
Pour chaque document $d \in T$ **faire**
    Transformer le document d en vecteur de mots $d = (x_1, x_2, \ldots, x_p)$
    Réduire le vecteur d en éliminant les mots vides
    Construire le vecteur d'occurrence $tf = (tf_1, tf_2, \ldots, tf_m)$
    dans le document d.
    Pour chaque mot x du vecteur d **faire**
        Construire l'ensemble S des concepts associés au mot x dans l'ontologie O
        Si stratégie de désambigüisation est *premier concept* **alors**
            Ajouter le concept **first(S)** au dictionnaire de concepts
            $Cf(d, first(S)) \longleftarrow Cf(d, first(S)) + tf(d, x)$
        Sinon
            Pour chaque concept $s \in S$ **faire**
                Ajouter le concept s au dictionnaire de concept
                $Cf(d, s) \longleftarrow Cf(d, s) + tf(d, x)$
            Finpour
        Finsi
    Finpour
Pour chaque concept s du dictionnaire de concept **faire**
    Construire l'ensemble H de concepts subsumant le concept s
    Pour chaque concept $h \in H$ **faire**
        Ajouter le concept h au dictionnaire de concepts
        $Cf(d, h) \longleftarrow cf(d, h) + cf(d, s)$
    Finpour
Finpour
Fin

7.2.2 Phase d'apprentissage

Cette phase consiste à créer un profil pour chaque catégorie, ce profil contiendra les concepts qui caractérisent ou discriminent bien la catégorie par rapport au autres catégories. Dans notre approche, nous avons choisi d'utiliser la méthode χ_2 multivariée pour la sélection des concepts caractéristiques. C'est une méthode supervisée permettant la sélection de termes en prenant en compte non seulement leurs fréquences dans chaque catégorie mais aussi l'interaction des termes entre eux et les interactions entre les termes et les catégories. Son principe consiste à extraire les K meilleurs termes caractérisant le mieux une catégorie par rapport aux autres, ceci pour chaque catégorie. Pour ce faire, le tableau croisé global (termes-catégories) qui représente le nombre total d'occurrences des termes, de dimension $p \times m$ (voir tableau 7.1) est calculé. La somme totale des occurrences est notée $\mathcal{N}$. Les valeurs N_{jk} des cellules (X_j, e_k) représentent le nombre de fois où le terme X_j est présent dans les documents étiquetés e_k. Puis, les contributions de ces cellules (X_j, e_k) au χ_2 associé à ce tableau sont calculées comme indiqué dans l'équation 7.1, puis triées par ordre décroissant pour chacune des catégories.

$$\mathcal{C}_{jk}^{\chi^2} = N\frac{(f_{jk} - f_{j.}f_{.k})^2}{f_{j.}f_{.k}} \times signe(f_{jk} - f_{j.}f_{.k}) \tag{7.1}$$

Le signe dans l'équation 7.1 permet de déterminer le sens de la contribution du terme à la discrimination de la catégorie par rapport aux autres catégories. Ainsi, un signe positif indique que c'est la présence du terme qui participe à la discrimination tandis qu'un signe négatif révèle que c'est son absence qui y participe.

Les principales caractéristiques de cette méthode sont les suivantes :

— Elle est supervisée car elle s'appuie sur l'information apportée par l'ensemble des documents étiquetés.

— Elle est multivariée car elle évalue globalement le role d'un terme par rapport aux

	e_1		e_k		e_m	
X^1	N_{11}	$\cdots$	N_{1k}	$\cdots$	N_{1m}	$N_{1.}$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
X^j	N_{j1}	$\cdots$	N_{jk}	$\cdots$	N_{jm}	$N_{j.}$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
X^n	N_{n1}	$\cdots$	N_{nk}	$\cdots$	N_{nm}	$N_{n.}$
	$N_{.1}$		$N_{.k}$		$N_{.m}$	$N = N_{..}$

TABLE 7.1: Tableau croisé global du nombre total d'occurrences

autres.

— Elle tient compte de l'intéraction termes-catégories car elle permet de choisir, pour chaque catégorie, les termes qui contribuent le plus à sa discrimination .

— Malgré sa sophistication, elle est de complexité linéaire en nombre de termes.

L'algorithme 2 illustre les étapes nécessaires pour la construction des profils conceptuels.

Algorithme 2 : Phase d'apprentissage

Entrées :
Matrice Cf de contingence (documents * concepts)
Dictionnaire *Dict* de concepts
L'ensemble de classes $C = \{c_1, c_2, \ldots, c_m\}$
La taille des profils : K
Sorties :
Les profils conceptuels des catégories
Debut
Construire le tableau croisé N d'occurrence de chaque concept dans chaque catégorie
Pour chaque classe $c_k \in C$ **faire**
Pour chaque concept $s_i \in Dict$ **faire**
Calculer la fréquence $f_{ki} = \frac{N_{ki}}{N}$
Calculer la contribution $\mathcal{C}_{ki}^{\chi^2} = N\frac{(f_{ki} - f_{k.}f_{.i})^2}{f_{k.}f_{.i}} \times signe(f_{ki} - f_{k.}f_{.i})$
Finpour
Trier les $\mathcal{C}_{ki}^{\chi^2}$ dans l'ordre décroisant
Construire le profil conceptuel de la catégorie c_k à partir des k premiers $\mathcal{C}_{ki}^{\chi^2}$
Finpour
Fin

7.2.3 phase de classification

Dans cette phase, il s'agit de comparer le profil du document à catégoriser avec les profils conceptuels des catégories déjà calculés dans la phase d'apprentissage. Pour cela, deux étapes sont nécessaires :

1. **Pondération des profils :** La pondération permet de représenter (numériquement) l'importance d'un concept dans une catégorie. Dans notre approche, nous avons choisi d'utiliser la mesure $TF - IDF$ car c'est la plus largement utilisée et la plus performante dans ce domaine. Un poid est attribué à chaque concept dans chaque catégorie selon la formule suivante :

$$tfidf(t_k, c_i) = tf(t_k, c_i) \times \log(\frac{|C|}{df(t_k)}) \tag{7.2}$$

   Avec :

   t_k :le concept à pondérer.

   c_i :la catégorie dont laquelle le concept est pondéré.

   $tf(t_k, c_i)$:la fréquence d'apparition du concept t_k dans les textes étiquetés c_i.

   $df(t_k)$:le nombre de catégories où le concept t_k figure dans leurs profils.

   $|C|$:le nombre total de catégories.

2. **Calcul de distance :** Une fois les profils sont pondérés, cette étape consiste à calculer la distance entre le profil du document et les profils des catégories. Finalement, le document sera assigné à la catégorie dont son profil est le plus proche à celui du document. Dans notre approche, nous avons choisi d'utiliser la mesure de similarité Cosinus, une des mesures de similarité les plus fréquemment utilisées car elle donne de bons résultats sur des corpus variés. Elle consiste à calculer les valeurs des cosinus des angles séparant les vecteurs des objets (les profils dans notre cas) qu'on veut comparer [131]. Par rapport à un simple produit scalaire, cette mesure présente l'avantage de normaliser les scores de chaque objet en fonction de sa taille, elle-même pondérée par le poids des termes. Le résultat

renvoyé est facilement exploitable ensuite car c'est une valeur située entre 0 et 1. La valeur 1 indiquant une similarité maximum (les deux objets sont identiques) et 0 une similarité nulle (les deux objets n'ont absolument rien en commun). Cette mesure se définie comme suit :

$$\mathcal{S}_{i,j} = \frac{\sum_{w \in i \frown j} tfidf(w,i) \times tfidf(w,j)}{\sqrt{\sum_{w \in i} tfidf^2(w,i) \times \sum_{w \in j} tfidf^2(w,j)}} \tag{7.3}$$

Avec :

w :un concept.

i et j : les deux objets(profils) a comparer.

$tfidf(w,i)$:le poids du concept w dans i.

$tfidf(w,j)$:le poids du concept w dans j. Ce qui peut ce traduire de la façon suivante : " **Plus on a de concepts communs et plus ces concepts communs ont des pondérations fortes, plus la similarité sera proche de 1, donc forte et vice versa.** "

Algorithme 3 : Phase de classification

Entrées :
Le vecteur conceptuel du document d à étiqueter
Les Profils conceptuels des catégories
Dictionnaire *Dict* de concepts
L'ensemble de classes $C = \{c_1, c_2, \ldots, c_m\}$
Sorties :
La catégorie à assigner au document d
Debut
Pondérer les profils conceptuels ainsi que le vecteur conceptuel du document d
Pour chaque classe $c_i \in C$ **faire**
    Calculer la mesure de similarité *Cosinus* entre le profil conceptuel de la classe c_i et le vecteur du document d
Finpour
Trier les similarités calculées par ordre décroissant
Assigner le document à la classe c ayant la plus grande valeur se similarité
Fin

7.3 Deuxième Approche

La deuxième approche se différencient par rapport à la première approche au niveau de la phase de représentation (voir figure 7.3). Contrairement à la première approche, la deuxième approche exclu l'utilisation des techniques de traduction en incorporant une ontologie pour chaque langue. Ainsi, les documents à catégoriser seront représentés directement par les concepts de l'ontologie de la langue dont il sont exprimés sans avoir besoin de les traduire vers la langue des documents d'apprentissage. Vu que le document à catégoriser et les documents d'apprentissage seront représentés par des ontologies différentes, il est nécessaire d'effectuer un alignement entre les deux ontologies afin de réduire les hétérogénéités terminologiques existantes entre les deux ontologies. Le processus d'alignement consiste à trouver des correspondances entre les connaissances spécifiées dans les deux ontologies utilisées dans le but de pouvoir les utiliser conjointement. Euzenat et al. classe dans [38] les méthodes d'appariement entre les concepts et/ou relations des deux ontologies en trois classes :

- les méthodes terminologiques qui se basent sur la comparaison des labels désignant deux concepts ou deux relations.
- les méthodes qui comparent les propriétés internes des concepts et relations (attributs des concepts, portée d'une relation,...,etc).
- les méthodes qui comparent les propriétés externes des concepts et relations (subsomptions, relations entre concepts,...,etc.).
- les méthodes qui comparent les extensions des concepts et relations.
- les méthodes qui comparent la sémantique des concepts et relations.

Dans le cas des ontologies multilingues, deux cas de figures se présentent [114] :

- Différentes ontologies monolingues développées séparément : Dans ce cas, chaque ontologie doit être alignée aux autres ontologies via des mapping 1 à 1. L'avantage de ce cas est la modélisation des concepts selon la culture de la nation pratiquant la langue. De ce fait, la conceptualisation dépasse le niveau terminologique. Néanmoins, il est difficile d'automatiser ce cas de figure car il nécessite l'intervention

humaine [114].

— Existence d'une ontologie monolingue servant de modèle pour le développement d'autres ontologies monolingues dans différentes langues : Dans ce cas, chaque ontologie doit être alignée au modèle pivot servant de passerelle entre les différentes ontologies monolingues. Ce cas de figure facilite le processus de mapping. En effet, pour incorporer une nouvelle ontologie monolingue, il suffit d'assurer l'alignement avec le modèle pivot. Cependant, la conceptualisation est limitée aux concepts et relations du modèle pivot.

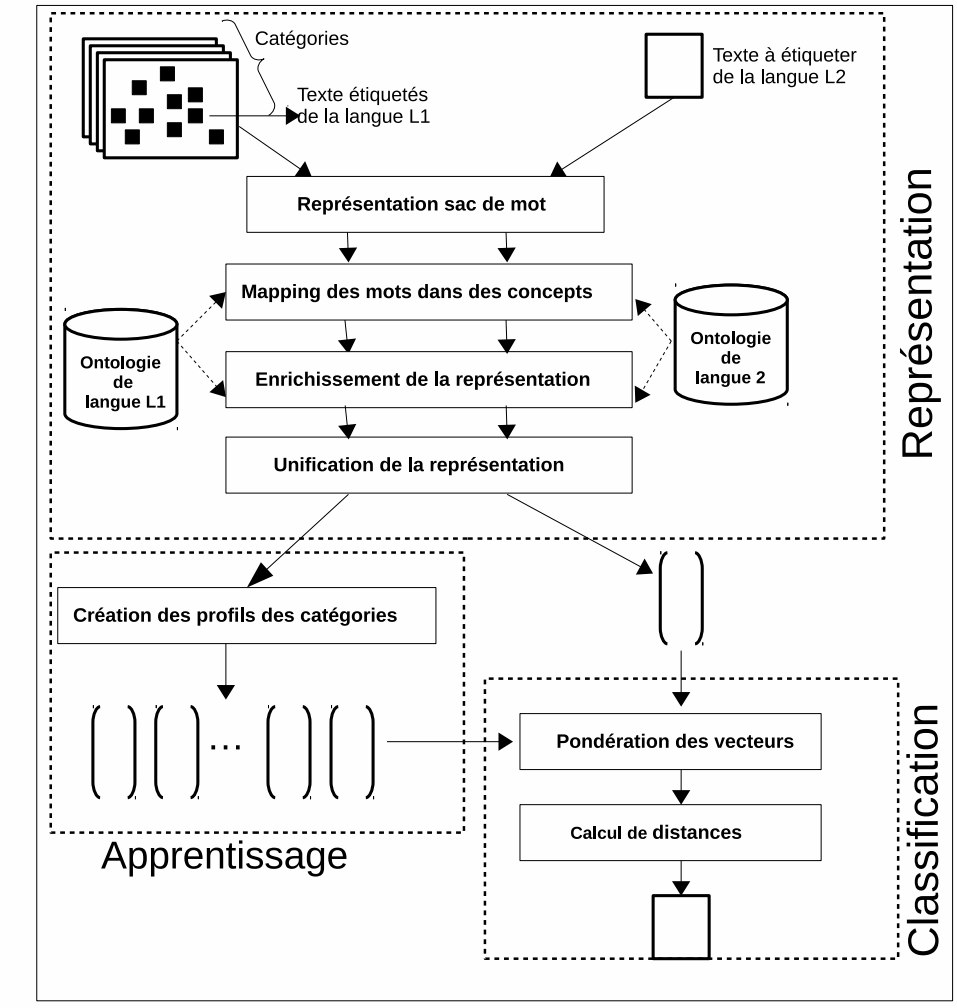


FIGURE 7.3 – *Architecture de la deuxième approche*

Algorithme 4 : Phase de représentation de la deuxième approche

Entrées :
Un ensemble T de documents
Ontologie O_1 pour la langue L_1
Ontologie O_2 pour la langue L_2
Fonction d'alignement Map des concepts des deux ontologies vers le modèle pivot
Sorties :
Matrice Cf de contingence (documents * concepts)
Dictionnaire de concepts
Debut
Pour chaque document $d \in T$ **faire**
Transformer le document d en vecteur de mots $d = (x_1, x_2, \ldots, x_p)$
Réduire le vecteur d en éliminant les mots vides
Construire le vecteur d'occurrence $tf = (tf_1, tf_2, \ldots, tf_m)$
dans le document d.
Pour chaque mot x du vecteur d **faire**
Construire l'ensemble S des concepts associés au mot x dans l'ontologie de la langue du document d ($O_1 ou O_2$)
Si stratégie de désambigüisation est *premier concept* **alors**
Ajouter le concept **Map(first(S))** au dictionnaire de concepts
$Cf(d, Map(first(S))) \longleftarrow Cf(d, Map(first(S))) + tf(d, x)$
Sinon
Pour chaque concept $s \in S$ **faire**
Ajouter le concept $Map(s)$ au dictionnaire de concept
$Cf(d, Map(s)) \longleftarrow Cf(d, Map(s)) + tf(d, x)$
Finpour
Finsi
Finpour
Pour chaque concept s du dictionnaire de concept **faire**
Construire l'ensemble H de concepts subsumant le concept s
Pour chaque concept $h \in H$ **faire**
Ajouter le concept h au dictionnaire de concepts
$Cf(d, h) \longleftarrow cf(d, h) + cf(d, s)$
Finpour
Finpour
Fin

7.4 Expérimentations et évaluation

Afin d'évaluer et de valider nos contributions, une phase d'expérimentation s'avère indispensable. Cette phase a pour objectif d'étudier les performances de nos approches implémentées. En outre, ceci nous permet aussi d'identifier les contraintes et les insuffisances de nos approches. Une présentation de l'environnement de développement qui va supporter notre application ainsi que les différentes ressources utilisées sont décrites dans un premier lieu dans cette section. La suite est consacrée à l'évaluation des résultats.

7.4.1 Ontologies utilisées

Dans le cadre de nos expérimentations, nous avons utilisé l'ontologie terminologique "Princeton WordNet"(PWN) développée par des linguistes du laboratoire des sciences cognitives de l'université de Princeton pour le traitement des documents exprimés en Anglais. C'est une ressource qui couvre des catégories lexico-sémantiques appelées *Synsets*. Ces synsets sont des ensembles de synonymes qui regroupent des items lexicaux ayant des significations similaires comme par exemple les mots "a board" (un panneau) et "a plank" (une planche) groupés dans le même synset. La définition des synsets varie du trés spécifique au trés général. Les synsets les plus spécifiques ne regroupent qu'un nombre restreint de significations lexicales alors que les synsets les plus généraux couvrent un nombre trés large de significations. La différence que présente WordNet par rapport aux dictionnaires traditionnels se traduit par la séparation des données en quatre bases de données associées aux catégories de verbes , des noms, d'adjectifs et d'adverbes. Chaque base de données est organisée différemment des autres. Ainsi, les noms sont organisés en hiérarchie, les verbes par des relations, les adjectifs et les adverbes par des hyper-espaces N-dimension. Les synsets sont ensuite reliés par différentes relations sémantiques tels que synonymie, antonymie, hyperonymie, méronymie,

métonymie, implication, causalité...etc. Dans nos expérimentations, nous avons utilisé la version la plus récente (3.0) qui couvre la majorité des noms, verbes, adjectifs et adverbes de la langue anglaise. le tableau 7.2 montre la distribution des synsets sur les quatre bases de données (nom, verbe, adjectif, adverbe). En plus de l'ontologie Word-

Base	Nombre de mots	nombre de synsets
Nom	117798	82115
Verbe	11529	13767
Adjective	21479	18156
Adverbe	4481	3621
Total	155287	117659

TABLE 7.2: Distribution des mots et synsets dans Wordnet3.0

Net, nous avons utilisé le WordNet espagnol développé au seins du laboratoire TALP de l'université polytechnique de Catalogne afin de pouvoir traiter les documents exprimés en langue espagnol. Cette ressource fait partie de l'ontologie multilingue EuroWordNet offrant ainsi l'avantage d'être structurée de la même manière que le Princeton WordNet. Le tableau 7.3 montre la distribution des synsets dans le WordNet espagnol utilisé.

Base	Nombre de mots	nombre de synsets
Nom	47732	43367
Verbe	12398	9043
Adjective	17999	14941
Total	78129	67351

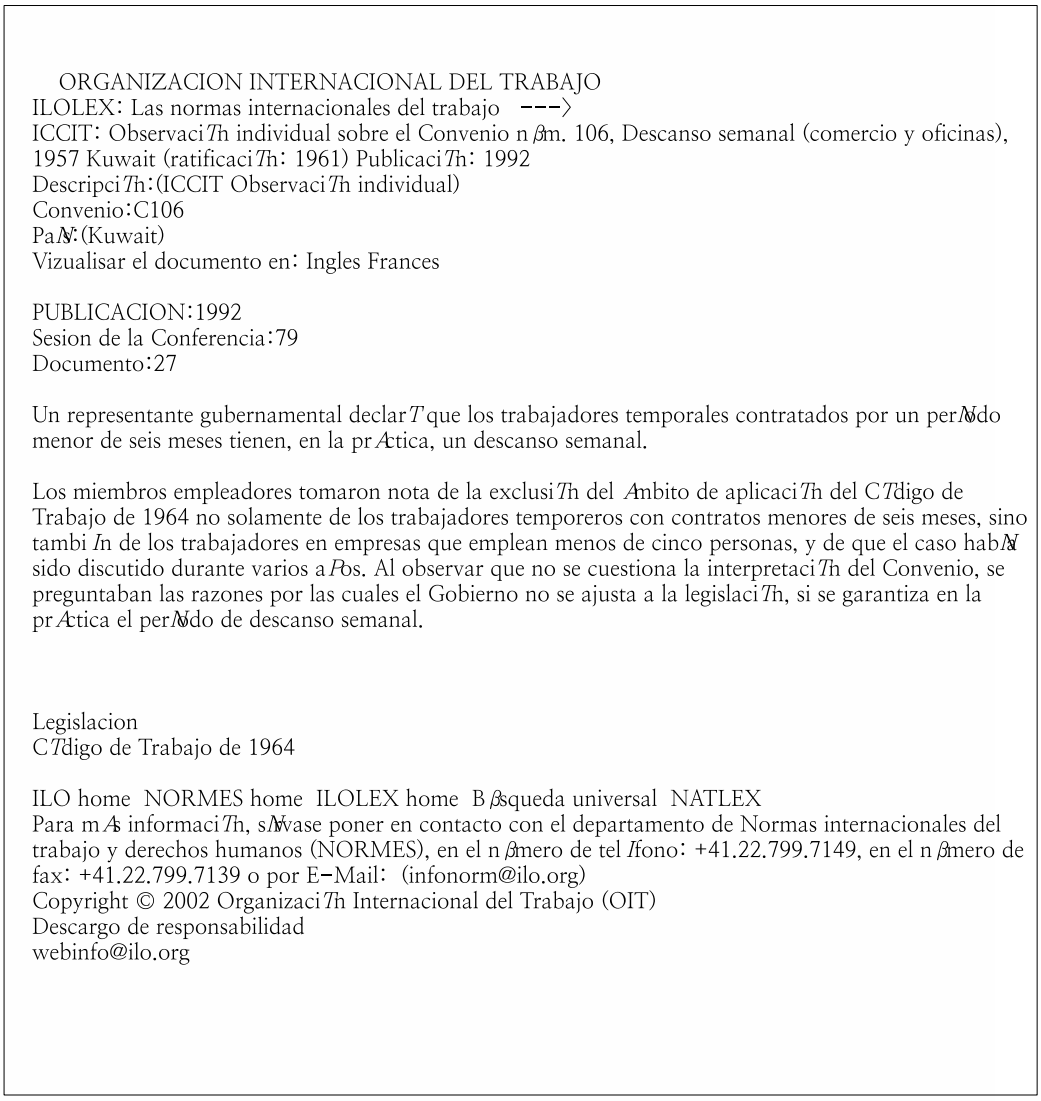
TABLE 7.3: Distribution des mots et synsets dans le Wordnet Espagnol

7.4.2 Corpus utilisés

Nous avons entrepris nos expérimentations en utilisant les deux corpus suivant :

1. **Corpus ILO :** Le corpus ILO est une collection trilingue de documents dont chaque document est étiqueté avec une seule catégorie (mono-classification) ac-

cessible gratuitement via le site web de l'organisation ILOLEX [1] (International Labour Organisation). Les documents de ce corpus traitent des sujets relatives aux domaine du travail sous forme de conventions, recommandation, ratifications et commentaires de comités. Les langages concernés sont l'anglais, l'espagnol et le français. La figure 7.4 montre un exemple d'un document du corpus ILO.



ORGANIZACION INTERNACIONAL DEL TRABAJO
ILOLEX: Las normas internacionales del trabajo ---〉
ICCIT: Observaci*T*h individual sobre el Convenio n*β*m. 106, Descanso semanal (comercio y oficinas), 1957 Kuwait (ratificaci*T*h: 1961) Publicaci*T*h: 1992
Descripci*T*h:(ICCIT Observaci*T*h individual)
Convenio:C106
Pa*N*:(Kuwait)
Vizualisar el documento en: Ingles Frances

PUBLICACION:1992
Sesion de la Conferencia:79
Documento:27

Un representante gubernamental declar*T* que los trabajadores temporales contratados por un per*N*do menor de seis meses tienen, en la pr*A*ctica, un descanso semanal.

Los miembros empleadores tomaron nota de la exclusi*T*h del *A*mbito de aplicaci*T*h del C*T*digo de Trabajo de 1964 no solamente de los trabajadores temporeros con contratos menores de seis meses, sino tambi*I*n de los trabajadores en empresas que emplean menos de cinco personas, y de que el caso hab*N* sido discutido durante varios a*P*os. Al observar que no se cuestiona la interpretaci*T*h del Convenio, se preguntaban las razones por las cuales el Gobierno no se ajusta a la legislaci*T*h, si se garantiza en la pr*A*ctica el per*N*do de descanso semanal.

Legislacion
C*T*digo de Trabajo de 1964

ILO home NORMES home ILOLEX home B*β*squeda universal NATLEX
Para m*A*s informaci*T*h, s*N*rvase poner en contacto con el departamento de Normas internacionales del trabajo y derechos humanos (NORMES), en el n*β*mero de tel*I*fono: +41.22.799.7149, en el n*β*mero de fax: +41.22.799.7139 o por E-Mail: (infonorm@ilo.org)
Copyright © 2002 Organizaci*T*h Internacional del Trabajo (OIT)
Descargo de responsabilidad
webinfo@ilo.org

FIGURE 7.4 – *Structure d'un documents ILO*

Dans nos expérimentations, nous avons utilisé une version bilingue de documents en anglais et espagnol répartis en 10 catégories. Le tableau 7.4 montre la réparti-

1. http ://www.ilo.org

tion des documents sur l'ensemble des catégories.

Catégorie	Anglais	Espagnol
Special prov. by Sector ef Econ. Act.	108	121
Conditions of employment	397	86
Conditions of work	299	71
Economic and Social development	22	23
Employement	410	448
Labour relations	276	278
Labour administration	85	81
Health and labour	98	86
Social security	150	148
Training	79	20
Total	1924	1362

TABLE 7.4: Répartition des documents sur les catégories du corpus ILO

2. **Corpus Reuters**[92] **:** En 2000, L'agence Reuters a mis en disposition une large collection de dépêches contenant deux volume RCV1 et RCV2. Le volume RCV1 contenant 810000 documents anglais, tandis que le volume RCV2 contient 487000 documents dans 13 langues différentes. Ces documents ne sont pas parallèle mais écrit par des journalistes locaux pour chaque langue. Les documents des deux volumes sont étiquetés par une liste de 126 catégories dont seulement 103 catégories sont réellement exploitées. La figure 7.5 montre la structure d'un document de cette collection.

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<newsitem itemid="2330" id="root" date="1996-08-20" xml:lang="en">
<title>USA: Tylan stock jumps; weighs sale of company.</title>
<headline>Tylan stock jumps; weighs sale of company.</headline>
<dateline>SAN DIEGO</dateline>
<text>
<p>The stock of Tylan General Inc. jumped Tuesday after the maker of
process-management equipment said it is exploring the sale of the
company and added that it has already received some inquiries from
potential buyers.</p>
<p>Tylan was up $2.50 to $12.75 in early trading on the Nasdaq market.</p>
<p>The company said it has set up a committee of directors to oversee
the sale and that Goldman, Sachs &amp;amp; Co. has been retained as its
financial adviser.</p>
</text>
<copyright>(c) Reuters Limited 1996</copyright>
<metadata>
<codes class="bip:countries:1.0">
  <code code="USA"> </code>
</codes>
<codes class="bip:industries:1.0">
  <code code="I34420"> </code>
</codes>
<codes class="bip:topics:1.0">
  <code code="C15">  </code>
  <code code="C152"> </code>
  <code code="C18">  </code>
  <code code="C181"> </code>
  <code code="CCAT"> </code>
</codes>
<dc element="dc.publisher" value="Reuters Holdings Plc"/>
<dc element="dc.date.published" value="1996-08-20"/>
<dc element="dc.source" value="Reuters"/>
<dc element="dc.creator.location" value="SAN DIEGO"/>
<dc element="dc.creator.location.country.name" value="USA"/>
<dc element="dc.source" value="Reuters"/>
</metadata>
</newsitem>
```

FIGURE 7.5 – *Structure d'un documents Reuters*

A partir de ce corpus, nous avons construit un corpus bilingue Anglais-Espagnol pour l'évaluation de nos approches. Le tableau 7.5 illustre la répartition des documents sur les 8 catégories utilisées.

Code Catégorie	Description catégorie	Anglais	Espagnol
C183	Privatisations	200	205
GSPO	Sport	401	84
GDIS	Disasters	278	116
GJOB	Labour issues	401	197
GDEF	Defence	227	83
GCRIM	Crim, Law enforcement	401	157
GDIP	International relations	401	237
GVIO	War, Civil war	401	306
Total		2710	1385

TABLE 7.5: Répartition des documents sur les catégories du corpus Reuters

7.4.3 Bibliothèques Utilisées

Dans le cadre de nos expérimentations, nous avons utilisé les bibliothèques suivantes :

1. **JWNL API :**[2] JWNL (Java WordNet Library) est une API Java permettant l'accès à l'ontologie WordNet à partir d'un programme java. Elle est compatible avec les versions WordNet 2.0 à 3.0.
2. **GTA API :**[3] GTA (Google Translate API) est une bibliothèque java permettant la traduction des documents d'une langue source vers une langue cible pour un nombre important de langue.

2. téléchargeable à partir de :http ://sourceforge.net/projects/jwordnet/
3. téléchargeable à partir de : https ://code.google.com/p/google-api-translate-java/

7.4.4 Résultats et discussion

Notre première expérimentation consiste à évaluer les performances de la première approche sur les deux corpus utilisés. Les résultats de cette expérimentation sont récapitulés dans les tableaux 7.6 et 7.7 (voir aussi les figures 7.6 et 7.7).

	Avec enrichissement		Sans enrichissement	
Désambiguïsation	First Synset	All Synset	First Synset	All Synset
k=100	0.645	0.643	0.602	0.595
k=200	0.649	0.661	0.625	0.635
k=300	0.653	0.656	0.631	0.633
k=400	0.661	0.661	0.641	0.620
k=500	0.663	0.659	0.646	0.637
k=600	0.668	0.660	0.649	0.638
k=700	0.669	0.661	0.650	0.644
k=800	0.673	0.669	0.654	0.646
k=900	0.671	**0.680**	0.656	0.655
k=1000	0.664	0.676	0.656	0.653

TABLE 7.6: Comparaison des résultats (macroaveraged F_1) de la première approche sur le corpus Reuters

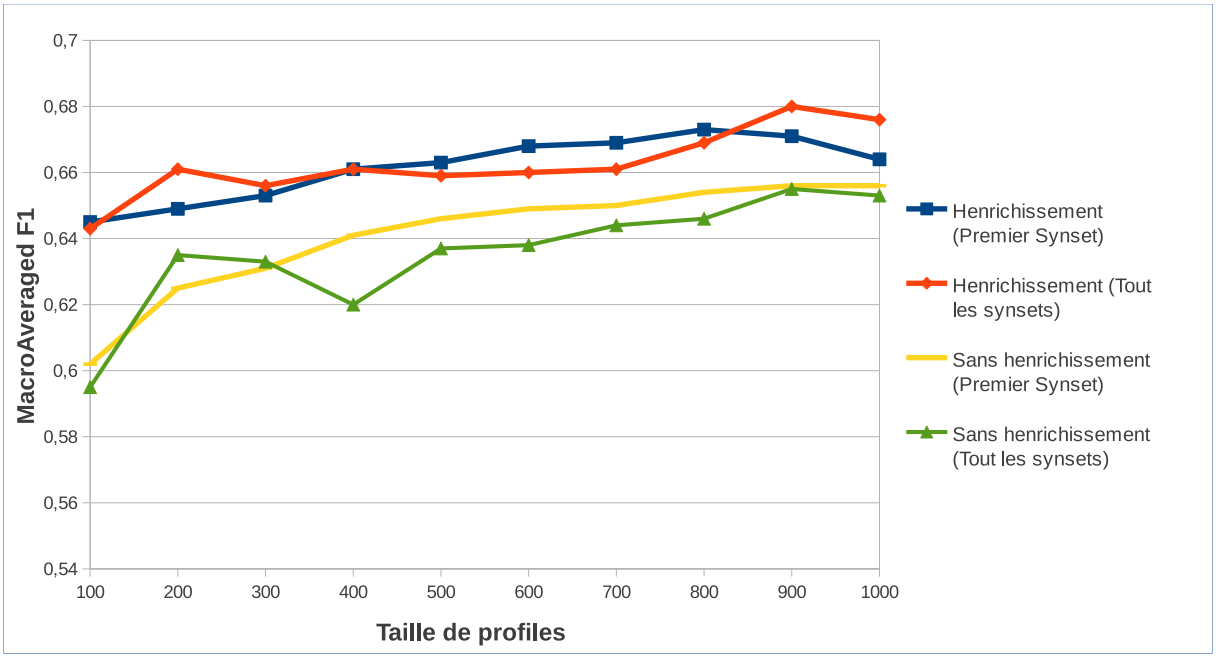


FIGURE 7.6 – *Représentation des résultats fournis dans le tableau 7.6*

Les résultats présentés montrent l'utilité de l'étape d'enrichissement dans la phase de représentation. En effet, pour les deux corpus, les meilleurs résultats ont été obtenu dans le cas "Avec Enrichissement". Pour le corpus Reuters, on constate que l'enrichissement a apporté une nette amélioration de 2.4%. Les mêmes constatations peuvent être affirmé sur le corpus ILO avec une nette amélioration de 3.3%.

	Avec enrichissement		Sans enrichissement	
	First Synset	All Synset	First Synset	All Synset
k=100	0.651	0.482	0.606	0.470
k=200	0.677	0.540	0.636	0.476
k=300	0.691	0.579	0.654	0.513
k=400	0.694	0.614	0.656	0.535
k=500	0.709	0.617	0.669	0.562
k=600	0.711	0.630	0.671	0.562
k=700	0.712	0.630	0.679	0.562
k=800	**0.716**	0.633	0.680	0.565
k=900	0.713	0.640	0.683	0.580
k=1000	0.713	0.640	0.683	0.577

TABLE 7.7: Comparaison des résultats (macroaveraged F_1 de la première approche sur le corpus ILO

Concernant la taille de profils, on constate qu'en augmentant la valeur de la taille de profil (K), les performances s'améliorent successivement puis ce stabilisent pour une certaine valeur qui varient selon le corpus à savoir $k = 800$ pour le corpus ILO et $k = 900$ pour le corpus Reuters. En ce qui concerne la désambiguïsation, on remarque que la stratégie "First Synset" est généralement plus performante que la stratégie "All Synset" pour les deux corpus.

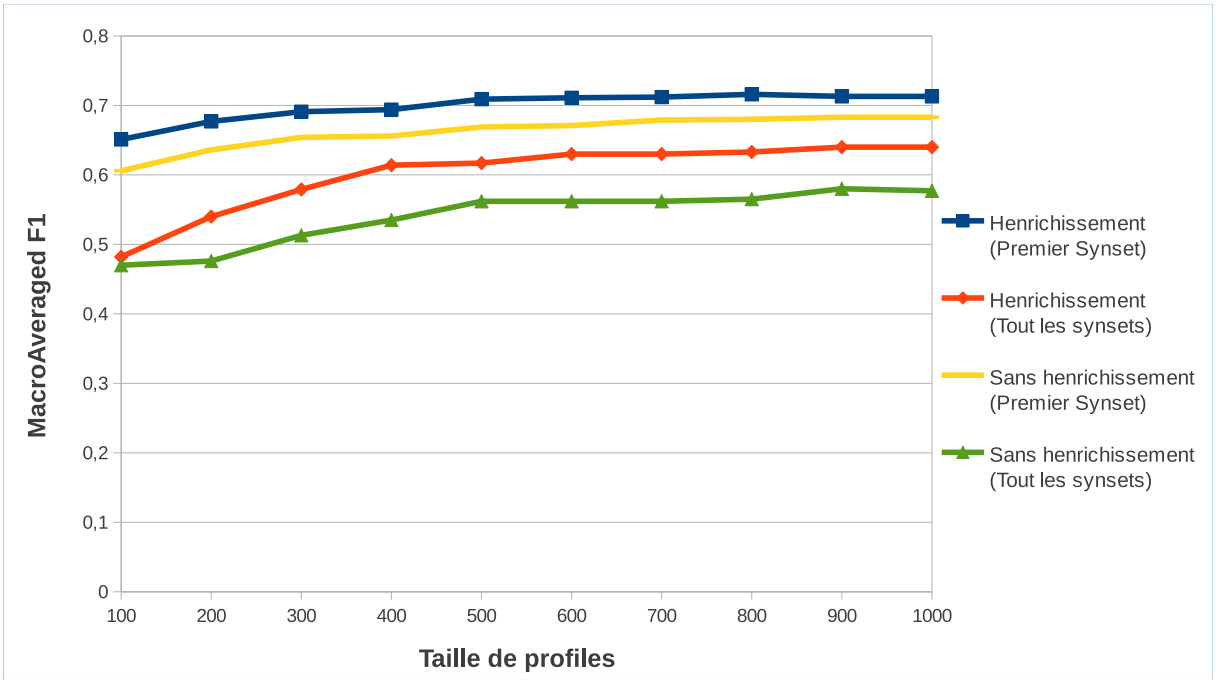


FIGURE 7.7 – *Représentation des résultats fournis dans le tableau 7.7*

La deuxième expérimentation est dédiée à l'évaluation des résultats de la deuxième approche sur les deux corpus utilisés. Dans cette expérimentation, il s'agit d'utiliser une ontologie pour chaque langue. Vu que les documents en anglais seront indexés par les synsets du Princeton WordNet tandis que les documents en espagnol seront indexés par les synsets du WordNet espagnol, il est nécessaire d'effectuer un alignement entre les deux WordNets. Dans cette expérimentation, nous avons utilisé l'alignement proposé dans [27]. Les résultats de cette expérimentation sont récapitulés dans les tableaux 7.8 et 7.9 (voir aussi les figures 7.8 et 7.9).

	Avec enrichissement		Sans enrichissement	
Désambiguïsation	First Synset	All Synset	First Synset	All Synset
k=100	0.475	0.431	0.418	0.361
k=200	0.495	0.433	0.424	0.381
k=300	0.496	0.423	0.435	0.405
k=400	0.518	0.433	0.437	0.401
k=500	0.520	0.435	0.442	0.408
k=600	0.528	0.453	0.441	0.411
k=700	0.520	0.463	0.454	0.419
k=800	0.523	0.468	0.460	0.410
k=900	0.520	0.472	0.454	0.423
k=1000	0.521	0.478	0.464	0.424

TABLE 7.8: Comparaison des résultats (macroaveraged F_1) de la deuxième approche sur le corpus Reuters

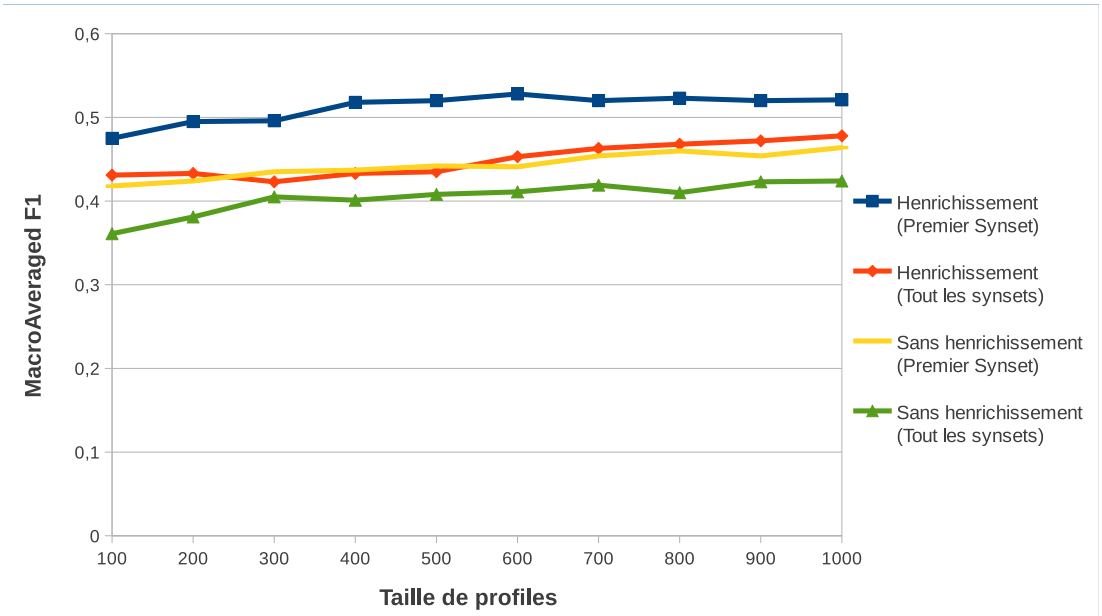


FIGURE 7.8 – *Représentation des résultats fournis dans le tableau 7.8*

	Avec enrichissement		Sans enrichissement	
Désambiguïsation	First Synset	All Synset	First Synset	All Synset
k=100	0.554	0.385	0.518	0.365
k=200	0.577	0.449	0.545	0.380
k=300	0.593	0.490	0.557	0.416
k=400	0.602	0.516	0.565	0.450
k=500	0.612	0.520	0.572	0.471
k=600	0.615	0.524	0.576	0.467
k=700	0.612	0.521	0.585	0.462
k=800	0.615	0.528	0.586	0.471
k=900	0.611	0.540	0.587	0.472
k=1000	0.615	0.541	0.589	0.476

TABLE 7.9: Comparaison des résultats (macroaveraged F_1) de la deuxième approche sur le corpus ILO

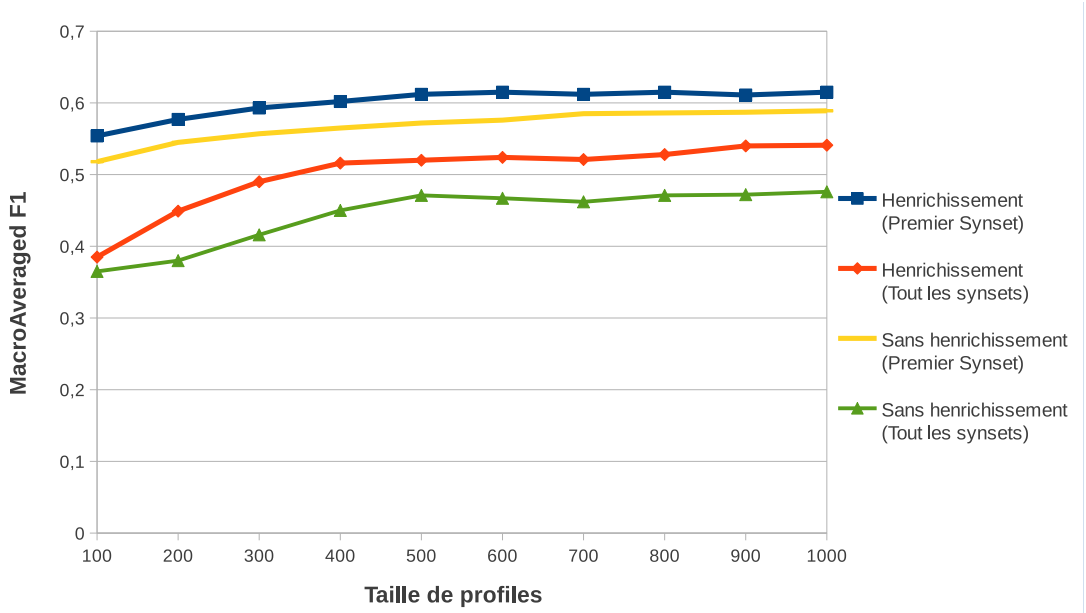


FIGURE 7.9 – *Représentation des résultats fournis dans le tableau 7.9*

En comparant les résultats des deux approches sur les deux corpus, on s'aperçoit que la première approche est plus performante que la deuxième approche. C'est vrai que la deuxième approche permet d'éviter la traduction des documents à étiqueter. Néanmoins, les résultats retrouvés ont été moins performant que ceux de la première approche. Cela s'explique par la non richesse du Wordnet espagnol utilisé par rapport au Princeton Wordnet. Les statistiques relatives aux nombre de synsets dans les deux WordNets utilisés montrent que le Princeton WordNet est largement riche avec ces 117659 synsets par rapport au Wordnet espagnol avec seulement 67351 synsets. Afin de pouvoir valider nos résultats, nous avons effectué une troisième expérimentation qui consiste à comparer les résultats des deux approches avec ceux de l'approche basée sur la traduction. Ainsi, il s'agit d'exclure l'utilisation des ontologies en utilisant la traduction comme seul moyen. Les résultats sont présentés dans le tableau 7.10 (voir aussi les figures 7.10 et 7.11).

	corpus ILO			Corpus Reuters		
Approche	Traduction	Approche1	Approche2	Traduction	Approche1	Approche2
k=100	0.641	0.651	0.554	0.624	0.645	0.475
k=200	0.653	0.677	0.577	0.644	0.649	0.495
k=300	0.670	0.691	0.593	0.644	0.653	0.496
k=400	0.670	0.694	0.602	0.654	0.661	0.518
k=500	0.670	0.709	0.612	0.658	0.663	0.520
k=600	0.670	0.711	0.615	0.660	0.668	0.528
k=700	0.670	0.712	0.612	0.657	0.669	0.520
k=800	0.670	**0.716**	0.658	0.658	0.673	0.523
k=900	0.670	0.713	0.611	0.657	0.671	0.520
k=1000	0.670	0.713	0.615	0.658	0.664	0.521

TABLE 7.10: Comparaison des résultats des deux approches avec l'approche basée sur la traduction

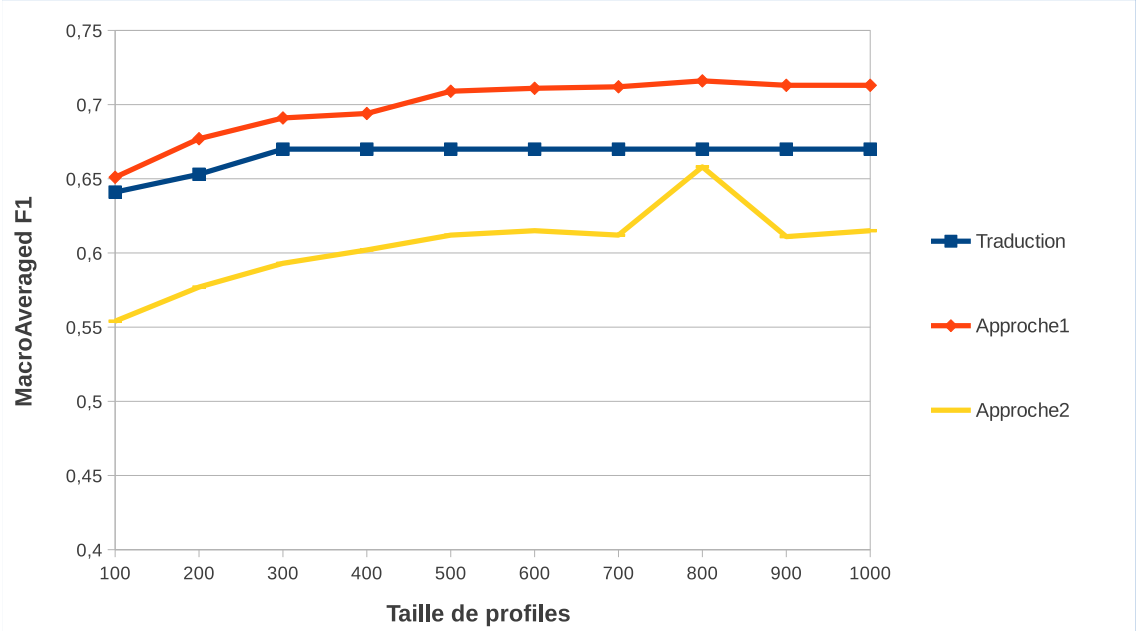


FIGURE 7.10 – *Représentation des résultats fournis dans le tableau 7.10 pour le corpus ILO*

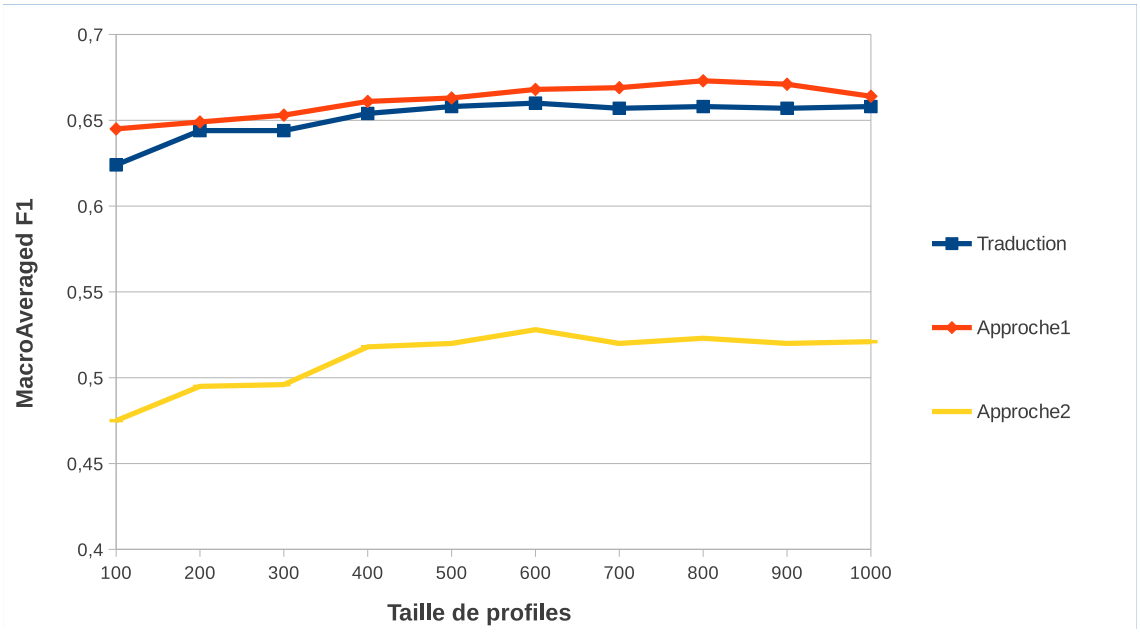


FIGURE 7.11 – *Représentation des résultats fournis dans le tableau 7.10 pour le corpus Reuters*

En analysant les résultats obtenus, on remarque que les résultats de notre première approche sur les deux corpus utilisés sont plus performant que l'approche basée sur la traduction. En effet, on constate une amélioration de 4.8% pour le corpus ILO et une amélioration de 1.2% pour le corpus Reuters. Cela nous permet d'affirmer que l'utilisation d'une ontologie assez riche comme le Princteon WordNet permet de réduire les distorsions causées par l'utilisation des techniques de traduction.

7.5 Conclusion

Dans ce chapitre, nous avons présentés nos contributions pour le problème de la catégorisation de textes multilingues. Plus précisément, nous avons introduit les ontologies comme moyen pour assister les techniques de traduction automatique. Nous avons proposé en premier lieu une approche qui consiste à combiner l'utilisation des traducteurs automatiques avec l'utilisation d'une ontologie monolingue. Cette première approche est beaucoup plus bénéfique dans le cas des langues les moins utilisées. La deuxième approche proposée se base sue l'exclusion des techniques de traduction en incorporant une ontologie pour chaque langue. Néanmoins, il devient nécessaire d'effectuer un alignement entre ces différentes ontologies. Les expérimentations entreprises dans ce chapitre ont montré l'utilité d'utilisation des ontologies pour la catégorisation de textes multilingues.

Chapitre 8

Conclusion

Nous nous somme intéressé au cours de cette thèse à l'utilisation des ontologies pour la catégorisation de textes multilingues. En effet, notre objectif consiste à étendre l'utilisation des ontologies dans la catégorisation monolingue pour catégoriser des documents provenant de différentes langues. Comme n'importe quel problème multilingue, la majorité des recherches dans le domaine de la catégorisation de textes multilingue utilisent les techniques de traduction afin de convertir le cas multilingue en un cas monolingue. Néanmoins, ces recherche sont pénalisées par l'inconvénient majeur de la traduction à savoir la perte d'information que peut induire une mauvaise traduction. Nos contributions consistent à proposer une extension de l'utilisation des ontologies dans le cas monolingue vers le cas multilingue dans le but d'assister l'utilisation de la traduction dans la catégorisation multilingue.

La première approche proposée consiste à utiliser le Princeton wordNet pour assister les techniques de traduction. En effet, l'utilisation d'une ressource assez riche peut être utile pour enrichir l'espace de la représentation ainsi que pour limiter les erreurs de traduction. Les résultats ont montré l'utilisation bénéfique du WordNet pour la catégorisation multilingue par rapport aux résultats obtenus en utilisant la traduction

comme seul moyen.

La popularité du Princeton WordNet a motivé les recherches pour la construction de ressources similaires dans d'autres langues. Ainsi, nous pouvons recenser plus de 50 WordNets d'autres langues. La deuxième approche consistait à éliminer l'utilisation des techniques de traduction. En effet, avec l'apparition de ces ressources, il devient possible de représenter directement un document dans la taxonomie du Wordnet associé à sa langue. Dans nos expérimentations, nous avons utilisé le WordNet Espagnol pour réaliser une catégorisation par croisement de langue (Anglais, Espagnol). Les expérimentations ont montré que les résultats de la première approche ont été plus performants que la deuxième approche. Cela est due au manque de richesse des différents WordNet par rapport au Princeton WordNet.

A l'issue de ce travail, de nombreuses pistes restent à explorer :

— Évaluation de nos approches sur d'autres langues afin de pouvoir confirmer nos constatations.

— Prendre en considération le cas des langues qui ne sont pas issues de la même famille : Dans nos expérimentations , nous avons pris en considération deux langues issues de la même famille à savoir la famille "indo-européenne". Par conséquent, la mise en correspondance entre les ressources sémantiques de ces deux langues est beaucoup plus facile par rapport au cas des langues issues de différentes familles. Notre deuxième perspectives rentre dans ce cadre. Elle concerne l'expérimentation de nos approches en prenant le cas des langues de différentes familles.

— Assister les WordNets d'autres langues par le biais de l'utilisation de ressources linguistiques tels que les corpus parallèles et corpus comparables : Les expérimentations menées sur la deuxième approche ont révélé l'importance de la richesse des ontologies utilisées sur les performances de la catégorisation. Une autre perspective consiste à incorporer une étape d'enrichissement des ontologies utilisés dans le but de garantir un alignement de meilleur qualité.

— Expérimenter nos approches sur un corpus focalisé d'un domaine bien spécifique

en utilisant une ontologie de domaine dans le but de minimiser les problèmes d'ambiguïté.

— Utilisation des mesures de similarité sémantiques à la place des mesures de similarité statistiques. En effet, nos approches n'utilisent les ontologies que dans la phase de représentation. Il serait important d'utiliser les ontologies dans la phase de classification et plus précisément dans le calcul de distance entre le vecteur du document à catégoriser et les profils des catégories. Cela permettra d'éliminer un des problème majeur du modèle vectoriel à savoir l'indépendance mutuelle des descripteurs.

Bibliographie

[1] K. Aas and L. Eikvil. Text categorisation : A survey. technical report, norwegian computing cente, 1999.

[2] Florence Amardeilh. *Web Sémantique et Informatique Linguistique : propositions méthodologiques et réalisation d'une plateforme logicielle*. PhD thesis, Université de Nanterre, Paris, France, 2004.

[3] Ion Androutsopoulos, John Koutsias, Konstantinos V. Chandrinos, and Constantine D. Spyropoulos. An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, pages 160–167, New York, NY, USA, 2000.

[4] Chidanand Apté, Fred Damerau, and Sholom M. Weiss. Automated learning of decision rules for text categorization. *ACM Trans. Inf. Syst.*, 12(3) :233–251, 1994.

[5] Chidanand Apté, Fred Damerau, and Sholom M. Weiss. Towards language independent automated learning of text categorization models. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, pages 23–30, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[6] T. Bailey and A. K. Jain. A note on distance-weighted k-nearest neighbor rules. *IEEE Transactions on Systems, Man and Cybernetics,*, 8(4) :311–313, April 1978.

[7] L. Douglas Baker and Andrew Kachites McCallum. Distributional clustering of words for text classification. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 96–103, New York, NY, USA, 1998.

[8] Lisa Ballesteros and Bruce Croft. Dictionary methods for cross-lingual information retrieval. In RolandR. Wagner and Helmut Thoma, editors, *Database and Expert Systems Applications*, volume 1134 of *Lecture Notes in Computer Science*, pages 791–801. Springer Berlin Heidelberg, 1996.

[9] Ron Bekkerman, Ran El-Yaniv, Naftali Tishby, and Yoad Winter. Distributional word clusters vs. words for text categorization. *Journal of Machine Learning Research.*, 3 :1183–1208, March 2003.

[10] Mohamed Amine Bentaallah and Mimoun Malki. Utilisation de wordnet dans la catégorisation de textes multilingues. In *Extraction et gestion des connaissances (EGC'2007), Actes des cinquièmes journées Extraction et Gestion des Connaissances, Namur, Belgique, 23-26 janvier 2007, 2 Volumes*, pages 195–196, 2007.

[11] Mohamed Amine Bentaallah and Mimoun malki. Wordnet based multilingual text categorization. *INFOCOMP Journal of Computer Science*, 6(4) :52–59, 2007.

[12] Mohamed Amine Bentaallah and Mimoun Malki. The use of wordnets for multilingual text categorization : A comparative study. In *Proceedings of the 4th International conference on Web and Information Technologies, ICWIT 2012, Sidi Bel Abbes, Algeria, April 29-30, 2012*, pages 121–128, 2012.

[13] Chris Biemann. Ontology learning from text : A survey of methods. *LDV Forum*, 20(2) :75–93, 2005.

[14] John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, bollywood, boomboxes and blenders : Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*,

pages 440–447, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[15] S. Bloehdorn and A. Hotho. Text classification by boosting weak learners based on terms and concepts. In *Data Mining, 2004. ICDM '04. Fourth IEEE International Conference on Data Mining*, pages 331–334. IEEE Computer Society Press, 2004.

[16] W. Borst. *Construction of Engineering Ontologies*. PhD thesis, niversity of Twente, Enschede, The Netherlands, 1997.

[17] Maria Fernanda Caropreso, Stan Matwin, and Fabrizio Sebastiani. Text databases and document management. chapter A Learner-independent Evaluation of the Usefulness of Statistical Phrases for Automated Text Categorization, pages 78–102. IGI Global, Hershey, PA, USA, 2001.

[18] William B. Cavnar and John M. Trenkle. N-gram-based text categorization. In *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, 1994.

[19] Kian Ming Adam Chai, Hai Leong Chieu, and Hwee Tou Ng. Bayesian online classifiers for text classification and filtering. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '02, pages 97–104, New York, NY, USA, 2002.

[20] William J. Clancey. Knowledge acquisition as modeling. chapter The Knowledge Level Reinterpreted : Modeling Socio-technical Systems, pages 33–49. John Wiley & Sons, Inc., New York, NY, USA, 1993.

[21] Cyril Cleverdon. Optimizing convenient online access to bibliographic databases. *Inf. Serv. Use*, 4(1-2) :37–47, 1984.

[22] William W. Cohen and Yoram Singer. Context-sensitive learning methods for text categorization. *ACM Trans. Inf. Syst.*, 17(2) :141–173, 1999.

[23] David Cohn, Rich Caruana, and Andrew Mccallum. Semi-supervised clustering with user feedback. Technical report, 2003.

[24] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Mach. Learn.*, 20(3) :273–297, September 1995.

[25] Mark Craven, Dan DiPasquo, Dayne Freitag, Andrew McCallum, Tom Mitchell, Kamal Nigam, and Seán Slattery. Learning to extract symbolic knowledge from the world wide web. In *Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence*, AAAI '98/IAAI '98, pages 509–516, Menlo Park, CA, USA, 1998. American Association for Artificial Intelligence.

[26] W. B. Croft. User-specified domain knowledge for document retrieval. In *Proceedings of the 9th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '86, pages 201–206, New York, NY, USA, 1986. ACM.

[27] J. Daudé, L. Padró, and G. Rigau. Mapping wordnets using structural information. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, ACL '00, pages 504–511, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[28] D. Davis and T. Dunning. A trec evaluation of query translation methods for multi-lingual text retrieval. In *Proceedings of the Fourth Text Retrieval Evaluation Conference, NIST*, 1995.

[29] Gerard De Melo and Stefan Siersdorfer. Multilingual text classification using ontologies. In *Proceedings of the 29th European Conference on IR Research*, ECIR'07, pages 541–548, Berlin, Heidelberg, 2007. Springer-Verlag.

[30] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6) :391–407, 1990.

[31] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incom-

plete data via the EM algorithm. *Journal of the Royal Statistical Society : Series B*, 39 :1–38, 1977.

[32] Lisa Di Jorio, Céline Fiot, Lylia Abrouk, Danièle Hérin, and Maguelonne Teisseire. Enrichissement d'ontologie : Quand les motifs séquentiels labellisent des relations. In *BDA'07 : Journées "Bases de Données Avancées"*, page 19, France, October 2007.

[33] Nakache Didier. *Extraction automatique des diagnostics à partir des comptes rendus médicaux textuels*. PhD thesis, CEDRIC Laboratory, Paris, France, 2007.

[34] Razika Driouche. *Proposition d'unearchitecture d'intégration des applications d'entreprise basée sur l'interopérabilité sémantique de l'EBXML et la mobilité des agents*. PhD thesis, Université Mentouri, Constantine, Algérie, 2007.

[35] H. Drucker, S. Wu, and V.N. Vapnik. Support vector machines for spam categorization. *Neural Networks, IEEE Transactions on*, 10(5) :1048–1054, Sep 1999.

[36] Susan Dumais and Hao Chen. Hierarchical classification of web content. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, pages 256–263, New York, NY, USA, 2000. ACM.

[37] Susan Dumais, John Platt, David Heckerman, and Mehran Sahami. Inductive learning algorithms and representations for text categorization. In *Proceedings of the Seventh International Conference on Information and Knowledge Management*, CIKM '98, pages 148–155, New York, NY, USA, 1998.

[38] Bouquet P. Dieng R. Ehrig M. Hauswirth M. Jarrar M. Lara R. Maynard D. Napoli A. Stamou G. Stuckenschmidt H. Shvaiko P. Tessaris S. van Acker S. Zaihrayeu I. Bach T. L. Euzenat J., Barrasa J. D2.2.3 : State of the art on ontology alignment. Technical report, 2004.

[39] M. Fernández-López, A. Gómez-Pérez, and N. Juristo. Methontology : from ontological art towards ontological engineering. In *Proc. Symposium on Ontological Engineering of AAAI*, 1997.

[40] B. Fortuna and J. Shawe-Taylor. The use of machine translation tools for cross-lingual text mining. In *Workshop on Learning with Multiple Views, ICML, 2005*, 2005.

[41] M.S. Fox, M. Barbuceanu, and M. Gruninger. An organisation ontology for enterprise modelling : preliminary concepts for linking structure and behaviour. In *Proceedings of the Fourth Workshop on Enabling Technologies : Infrastructure for Collaborative Enterprises,*, pages 71–81, Apr 1995.

[42] John Fraser and Austin Tate. The enterprise tool set - an open enterprise architecture. In *Proceedings of the Workshop on Intelligent Manufacturing Systems, International Joint Conference on Artificial Intelligence (IJCAI-95*, 1995.

[43] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1) :119 – 139, 1997.

[44] F.Sebastianni. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1) :1–47, 2002.

[45] N. Fuhr and G. E. Knorz. Retrieval Test Evaluation of a Rule Based Automatic Indexing (AIR/PHYS). In *Proceedings of the 7th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 391–408, Swinton, UK, UK, 1984. British Computer Society.

[46] Norbert Fuhr and Chris Buckley. A probabilistic learning approach for document indexing. *ACM Trans. Inf. Syst.*, 9(3) :223–248, 1991.

[47] Luigi Galavotti, Fabrizio Sebastiani, and Maria Simi. Experiments on the use of feature selection and negative evidence in automated text categorization. In José Borbinha and Thomas Baker, editors, *Research and Advanced Technology*

for Digital Libraries, volume 1923 of *Lecture Notes in Computer Science*, pages 59–68. Springer Berlin Heidelberg, 2000.

[48] WilliamA. Gale, KennethW. Church, and David Yarowsky. A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26(5-6) :415–439, 1992.

[49] Fabien Gandon, Fabien G, and Rose Dieng-Kuntz. Ontologie pour un systeme multi-agents dedie a une memoire d'entreprise. In *Actes des journées francophones d'Ingénierie des Connaissances IC'2001*, pages 1–20, 2001.

[50] Nunberg Geoffrey. Will the internet always speak english ? Technical report, 2001.

[51] Y. Gilli. *Texte et fréquence*. Number v. 360 in Annales littéraires de l'Universite de Franche-Comté : Université de Franche-Comté. 1988.

[52] Alfio Gliozzo and Carlo Strapparava. Cross language text categorization by acquiring multilingual domain models from comparable corpora. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, ParaText '05, pages 9–16, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[53] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain adaptation for large-scale sentiment classification : A deep learning approach. In *Proceedings of theTwenty-eight International Conference on Machine Learning (ICML'11)*, volume 27, pages 97–110, June 2011.

[54] J. Gonzalo, F. Verdejo, I. Chugur, and J. Cigarran. Indexing with WordNet synsets can improve text retrieval. *Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems*, 1998.

[55] J. Greenberg. Metadata extraction and harvesting : a comparison of two automatic metadata generation applications. *Internet Cataloging*, 6 :59–82, 2004.

[56] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowl. Acquis.*, 5(2) :199–220, June 1993.

[57] Michael Gruninger and Mark S. Fox. The design and evaluation of ontologies for enterprise engineering. Workshop on Implemented Ontologies, European Conference on Artificial Intelligence, 1994.

[58] Dziczkowski Grzegorz. *Analyse des sentiments : système autonome d ?exploration des opinions exprimées dans les critiques cinématographiques*. PhD thesis, Ecole Nationale Supérieure des Mines de Paris, Paris, FRANCE, 2008.

[59] JoséMaría Gómez, JoséCarlos Cortizo, Enrique Puertas, and Miguel Ruiz. Concept indexing for automated text categorization. In Farid Meziane and Elisabeth Métais, editors, *Natural Language Processing and Information Systems*, volume 3136 of *Lecture Notes in Computer Science*, pages 195–206. Springer Berlin Heidelberg, 2004.

[60] Asunción Gómez-Pérez, Mariano Fernández-López, and A.J DE VINCENTE. Towards a method to conceptualize domain ontologies. In *ECAI-96 Workshop on Ontological Engineering*, ECAI-96 Workshop Proceedings, 1996.

[61] Hele-Mai Haav and Tanel-Lauri Lubi. A survey of concept-based information retrieval tools on the web. In *Proc. 5th East-European Conference ADBIS*, volume 2, pages 29–41, 2001.

[62] P. J. Hayes, P. M. Andersen, I. B. Nirenburg, and L. M. Schmandt. TCS : a shell for content-based text categorization. In *Sixth Conference on Artificial Intelligence for Applications*, pages 320–326, Santa Barbara, California, United States, 1990. IEEE Press.

[63] Philip J. Hayes and Steven P. Weinstein. Construe/tis : A system for content-based indexing of a database of news stories. In *Proceedings of the The Second Conference on Innovative Applications of Artificial Intelligence*, IAAI '90, pages 49–64. AAAI Press, 1991.

[64] Tan A.-H. He, J. and Tan. A comparative study on chinese text categorization methods. In *PRICAI Workshop on Text and Web Mining*, pages 24–35, 2000.

[65] Marti A. Hearst. Noun homograph disambiguation using local context in large text corpora. In *University of Waterloo*, pages 1–22, 1991.

[66] Nathalie Hernandez. *Ontologies de domaine pour la modélisation du contexte en Recherche d'Information*. Thèse de doctorat, Université Paul Sabatier, Toulouse, France, décembre 2005.

[67] William Hersh, Chris Buckley, T. J. Leone, and David Hickam. Ohsumed : An interactive retrieval evaluation and new large test collection for research. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, pages 192–201, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[68] Lobna HLAOUA. *Reformulation de Requètes par Réeinjection de Pertinence dans les Documents Semi-Structurées*. PhD thesis, Institut de recherche en informatique, Toulouse, France, 2007.

[69] Andreas Hotho, Steffen Staab, and Gerd Stumme. Ontologies improve text document clustering. In *Proceedings of the Third IEEE International Conference on Data Mining*, ICDM '03, pages 541–, Washington, DC, USA, 2003. IEEE Computer Society.

[70] Shuqing Huang. *A Comparative Study of Clustering and Classification Algorithms*. PhD thesis, New Orleans, LA, USA, 2007. AAI3258261.

[71] D. Hull and G. Grefenstette. Querying across languages. a dictionary-based approach to multilingual information retrieval. In *PROCEEDINGS OF 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 49–57, 1996.

[72] David Hull. Improving text retrieval for the routing problem using latent semantic indexing. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, pages 282–291, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[73] Christophe Jacquet. *Étude de l ?intégration d ?informations symboliques et analogiques dans le cadre de l ?aide au déplacement de personnes non voyantes*. Rapport de stage dea, Université PARIS-SUD 11, PARIS, France, septembre 2003.

[74] Radwan Jalam. *Apprentissage automatique et catégorisation de textes multilingues*. PhD thesis, Université Lyon2, Lyon, France, 2003.

[75] Radwan Jalam and Jean-Hugues Chauchat. Pourquoi les n-grammes permettent de classer des textes ? recherche de mots-clefs pertinents à l ?aide des n-grammes caractéristiques. In *6es Journées internationales d ?Analyse statistique des Données Textuelles*, JADT, 2002.

[76] Thorsten Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*, ICML '97, pages 143–151, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.

[77] Thorsten Joachims. Text categorization with suport vector machines : Learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning*, ECML '98, pages 137–142, London, UK, UK, 1998. Springer-Verlag.

[78] Thorsten Joachims. A statistical learning model of text classification for support vector machines. In *In Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 128–136. ACM Press, 2001.

[79] G. V. Kass. An exploratory technique for investigating large quantities of categorical data. *Applied Statistics*, 29 :119–127, 1980.

[80] Yu-Hwan Kim, Shang-Yoon Hahn, and Byoung-Tak Zhang. Text filtering by boosting naive bayes classifiers. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, pages 168–175, New York, NY, USA, 2000. ACM.

[81] Yannis Labrou and Tim Finin. Yahoo ! as an ontology : Using yahoo ! categories to describe documents. In *Proceedings of the Eighth International Conference on Information and Knowledge Management*, CIKM '99, pages 180–187, New York, NY, USA, 1999. ACM.

[82] Javier Lacasta, Javier Nogueras-Iso, and Francisco Javier Zarazaga-Soria. *Terminological Ontologies*, volume 9 of *Semantic Web and Beyond*. Springer, New York, 2010.

[83] Ken Lang. Newsweeder : Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 331–339, 1995.

[84] Leah S. Larkey. A patent search and classification system. In *Proceedings of the Fourth ACM Conference on Digital Libraries*, DL '99, pages 179–187, New York, NY, USA, 1999. ACM.

[85] V. Lavrenko. *A Generative Theory of Relevance*. PhD thesis, University of Massachusetts, Amherst, MA,, 2004.

[86] L. Lebart and A. Salem. *Statistique textuelle*. Dunod, 1994.

[87] Frûst F. Leclère L., Trichet T. nstruction of an ontology related to the projective geometry. In *RFIA 13th congrès des Reconnaissance des Frames et Intelligence Artificielle*, 2002.

[88] Phillipe. Lefèvre. La rechreche d'information - du texte intégral au thésaurus, 2000.

[89] Douglas B. Lenat and R. V. Guha. *Building Large Knowledge-Based Systems ; Representation and Inference in the Cyc Project*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1989.

[90] David D. Lewis. An evaluation of phrasal and clustered representations on a text categorization task. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '92, pages 37–50, New York, NY, USA, 1992. ACM.

[91] David D. Lewis and William A. Gale. A Comparison of Two Learning Algorithms for Text Categorization. In *Symposium on Document Analysis and Information Retrieval*, 1995.

[92] David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. Rcv1 : A new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5 :361–397, December 2004.

[93] David Dolan Lewis. *Representation and Learning in Information Retrieval*. PhD thesis, University of Massachusetts, Amherst, MA, USA, 1992.

[94] Elizabeth D. Liddy, Woojin Paik, and Edmund S. Yu. Text categorization for multiple users based on semantic features from a machine-readable dictionary. *ACM Trans. Inf. Syst.*, 12(3) :278–295, 1994.

[95] MichaelL. Littman, SusanT. Dumais, and ThomasK. Landauer. Automatic cross-language information retrieval using latent semantic indexing. In Gregory Grefenstette, editor, *Cross-Language Information Retrieval*, volume 2 of *The Springer International Series on Information Retrieval*, pages 51–62. Springer US, 1998.

[96] Yang Y. Liu, Y. and J Carbonell. Boosting to correct the inductive bias for text classification. In *Proceedings of CIKM-02, 11th ACM International Conference on Information and Knowledge Management*, McLean, New York, NY, USA, 2002. ACM Press.

[97] Claude de Loupy. *Evaluation de l'apport de connaissances linguistiques en désambiguïsation sémantique et recherche documentaire*. PhD thesis, Avignon, Grenoble, 2000.

[98] Simon Marcellin. *Arbres de décision en situation d ?asymétrie*. PhD thesis, Université Lyon2, Lyon, France, 2008.

[99] Litvak Marina, Last Mark, and Kisilevich Slava. Improving classification of multilingual web documents using domain ontologies. In *Proceedings of the Second In-*

ternational Workshop on Knowledge Discovery and Ontologies (in ECML/PKDD 2005), pages 67–74, 2005.

[100] Brian McBride. Rdf vocabulary description language 1.0 : Rdf schema, 2004.

[101] Andrew McCallum and Kamal Nigam. A comparison of event models for naive bayes text classification. In *IN AAAI-98 WORKSHOP ON LEARNING FOR TEXT CATEGORIZATION*, pages 41–48. AAAI Press, 1998.

[102] Andrew McCallum, Ronald Rosenfeld, Tom M. Mitchell, and Andrew Y. Ng. Improving text classification by shrinkage in a hierarchy of classes. In *Proceedings of the Fifteenth International Conference on Machine Learning*, ICML '98, pages 359–367, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[103] George H. Mealy. Another look at data. In *Proceedings of the November 14-16, 1967, Fall Joint Computer Conference*, AFIPS '67 (Fall), pages 525–534, New York, NY, USA, 1967.

[104] C.P. Menzel, R.J. Mayer, University of Houston-Clear Lake. Research Institute for Computing, Information Systems, and Lyndon B. Johnson Space Center. Information Technology Division. *IDEF5 Ontology Description Capture Method : Concept Paper*. Technical report, University of Houston–Clear Lake. Research Institute for Computing and Information Systems. 1990.

[105] Mohamed Ben Ahmed Mhiri, Faïez Gargouri, and Djamal Benslimane. Détermination automatique des relations sémantiques entre les concepts d'une ontologie. In *INFORSID*, pages 627–642, 2006.

[106] Rada Mihalcea and Dan Moldovan. Semantic indexing using wordnet senses. In *Proceedings of the ACL-2000 Workshop on Recent Advances in Natural Language Processing and Information Retrieval : Held in Conjunction with the 38th Annual Meeting of the Association for Computational Linguistics - Volume 11*, RANLPIR '00, pages 35–45, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[107] Dunja Mladenic and Marko Grobelnik. Mapping documents onto web page ontology. In *Web Mining : From Web to Semantic Web*, volume 3209 of *Lecture Notes in Computer Science*, pages 77–96. Springer Berlin Heidelberg, 2004.

[108] Alessandro Moschitti. A study on optimal parameter tuning for rocchio text classifier. In *In Proceedings of the 25th European Conference on Information Retrieval Research (ECIR'03)*, pages 420–435. Springer Verlag, 2003.

[109] Isabelle Moulinier. Feature selection : A useful preprocessing step. In *Proceedings of BCSIRSG-97, the 19th Annual Colloquium of the British Computer Society Information Retrieval Specialist Group, Electronic Workshops in Computing*, IRSG'97, pages 6–6, Swinton, UK, UK, 1997. British Computer Society.

[110] J. Scott Olsson, Douglas W. Oard, and Jan Hajič. Cross-language text classification. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '05, pages 645–646, New York, NY, USA, 2005.

[111] Xiaogang Peng and Ben Choi. Document classifications based on word semantic hierarchies. In *In Proceedings of the International Conference on Artificial Intelligence and Applications (AIA'05*, pages 362–367, 2005.

[112] V. Petridis, V.G. Kaburlasos, P. Fragkou, and A. Kehagias. Text classification using the sigma ;-flnmap neural network. In *Neural Networks, 2001. Proceedings. IJCNN '01. International Joint Conference on Neural Networks*, volume 2, pages 1362–1367, 2001.

[113] S. Pitigala, Cen Li, and Suk Seo. A comparative study of text classification approaches for personalized retrieval in pubmed. In *2011 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW),*, pages 919–921, Nov 2011.

[114] Elena Montiel Ponsoda. *Multilingualism in Ontologies*. PhD thesis, Universidad Politécnica de Madrid E.T.S.I. Montes, Spain, 2011.

[115] M. F. Porter. Readings in information retrieval. chapter An Algorithm for Suffix Stripping, pages 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.

[116] Peter Prettenhofer and Benno Stein. Cross-language text classification using structural correspondence learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1118–1127, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[117] J. R. Quinlan. Induction of decision trees. *Mach. Learn.*, 1(1) :81–106, March 1986.

[118] R. Quinlan. Decision trees as probabilistic classifiers. In *International Conference on Machine Learning*, 1987.

[119] Dr. J. Akilandeswari R. Subhashini. A survey on ontology construction methodologies. *International Journal of Enterprise Computing and Business System, International Systems*, 1(1), 2011.

[120] K. Radwan, F. Foussier, and C. Fluhr. Multilingual access to textual databases. In *Conference on Intelligent Text and Image Handling RIAO91*, pages 475–489, 1991.

[121] Sudha Ram, Jinsoo Park, and Dongwon Lee. Digital libraries for the next millennium : Challenges and research directions. *Information Systems Frontiers*, pages 75–94, 1999.

[122] Ganesh Ramakrishnanan and Pushpak Bhattacharyya. Text representation with wordnet synsets using soft sense disambiguation. In *Proceedings of 8th International Conference on Applications of Natural Language to Information Systems (NLDB 2003*, pages 214–227, 2003.

[123] S. Réhel. *Catégorisation automatique de textes et cooccurrence de mots : Catégorisation automatique de textes et cooccurrence de mots provenant de documents non étiquetés.* Editions universitaires europeennes EUE, 2011.

[124] Leonardo Rigutini, Marco Maggini, and Bing Liu. An em based training algorithm for cross-language text categorization. In *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence*, WI '05, pages 529–535, Washington, DC, USA, 2005. IEEE Computer Society.

[125] C. J. Van Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 2nd edition, 1979.

[126] S. E. Robertson and S. Walker. On relevance weights with little relevance information. In *SIGIR'97 : Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 16–24. ACM Press, January 1997.

[127] J. J. Rocchio. Relevance feedback in information retrieval. In *The SMART Retrieval System*, pages 313–323. 1971.

[128] Monica Rogati and Yiming Yang. High-performing feature selection for text classification. In *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, CIKM '02, pages 659–661, New York, NY, USA, 2002.

[129] Carl Sable and Kenneth W. Church. Using bins to empirically estimate term weights for text categorization. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP-01)*, 2001.

[130] M. Sahami. *Using Machine Learning to Improve Information Access*. PhD thesis, Computer Science Department, Stanford University, 1999.

[131] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11) :613–620, November 1975.

[132] Gerard Salton. Automatic processing of foreign language documents. In *Proceedings of the 1969 Conference on Computational Linguistics*, COLING '69, pages 1–28, Stroudsburg, PA, USA, 1969. Association for Computational Linguistics.

[133] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[134] Robert E. Schapire, Yoav Freund, Peter Barlett, and Wee Sun Lee. Boosting the margin : A new explanation for the effectiveness of voting methods. In *Proceedings of the Fourteenth International Conference on Machine Learning*, ICML '97, pages 322–330, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.

[135] Robert E. Schapire and Yoram Singer. Boostexter : A boosting-based system for text categorization. In *Machine Learning*, pages 135–168, 2000.

[136] Robert E. Schapire, Yoram Singer, and Amit Singhal. Boosting and rocchio applied to text filtering. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 215–223, New York, NY, USA, 1998. ACM.

[137] Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK, 1994.

[138] Sam Scott and Stan Matwin. Text classification using wordnet hypernyms. In *Workshop on usage of WordNet in NLP Systems (COLING-ACL '98)*, pages 45–52, 1998.

[139] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3) :379–423, 1948.

[140] Lei Shi, Rada Mihalcea, and Mingjun Tian. Cross language text classification by model translation and semi-supervised learning. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1057–1067. Association for Computational Linguistics, 2010.

[141] Amit Singhal, Mandar Mitra, and Chris Buckley. Learning routing queries in a query zone. In *Proceedings of the 20th Annual International ACM SIGIR Confe-*

rence on Research and Development in Information Retrieval, SIGIR '97, pages 25–32, New York, NY, USA, 1997.

[142] Noam Slonim and Naftali Tishby. The power of word clusters for text classification. In *Proceedings of the 23rd European Colloquium on Information Retrieval Research*, 2001.

[143] Barry Smith. Ontology : Philosophical and computational. *National Science Foundation*, 2004.

[144] John F. Sowa. *Knowledge Representation : Logical, Philosophical and Computational Foundations*. Brooks/Cole Publishing Co., Pacific Grove, CA, USA, 2000.

[145] Ralf Steinberger, Bruno Pouliquen, and C. Ignat. Navigating multilingual news collections using automatically extracted information. In *Proceedings of the 27th International Conference onInformation Technology Interfaces*, pages 25–32, June 2005.

[146] Mathieu Stricker. *Réseaux de neurones pour le traitement automatique du langage : conception et réalisatin de filtres d'informations*. Thèse de doctorat en sciences et techniques, ESPCI ParisTECH, 2000.

[147] Rudi Studer, V. Richard Benjamins, and Dieter Fensel. Knowledge engineering : Principles and methods. *Data Knowl. Eng.*, 25(1-2) :161–197, March 1998.

[148] Xavier Tannier. Traitement automatique du langage naturel pour l'extraction et la recherche d'informations. Technical report, March 2006. http ://www.emse.fr/spip/IMG/pdf/RR_2006-400-006.pdf.

[149] Yannick Toussaint. Extraction de connaissances à partir de textes structurés. *Document Numérique*, 8(3) :11–34, 2004.

[150] Chih-Fong Tsai. Bag-of-words representation in image annotation : A review. *ISRN Artificial Intelligence*, 2012.

[151] T.Saracevic. a review of and a framework for the thinking on the notion in information science. *American Society for Information Science*, 6 :321–343, 1975.

[152] Kostas Tzeras and Stephan Hartmann. Automatic indexing based on bayesian inference networks. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '93, pages 22–35, New York, NY, USA, 1993. ACM.

[153] Mike Uschold, Michael Gruninger, Mike Uschold, and Michael Gruninger. Ontologies : Principles, methods and applications. *Knowledge Engineering Review*, 11 :93–136, 1996.

[154] Mike Uschold and Martin King. Towards a methodology for building ontologies. In *Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95*, Montreal, Canada, 1995.

[155] G. van Heijst, A. Th. Schreiber, and B. J. Wielinga. Using explicit ontologies in kbs development. *Int. J. Hum.-Comput. Stud.*, 46(2-3) :183–292, March 1997.

[156] Vladimir Vapnik. *Estimation of Dependences Based on Empirical Data : Springer Series in Statistics (Springer Series in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1982.

[157] Romain Vinot. *Classification automatique de textes dans des catégories non thématiques*. PhD thesis, Paris, France, 2007.

[158] Natasha Vleduts-Stokolov. Concept recognition in an automatic text-processing system for the life sciences. *Journal of the American Society for Information Science*, 38(4) :269–287, 1987.

[159] H. Wache, T. Vögele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann, and S. Hübner. Ontology-based integration of information - a survey of existing approaches. pages 108–117, 2001.

[160] Chang Wan, Rong Pan, and Jiefei Li. Bi-weighting domain adaptation for cross-language text classification. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, volume 2 of *IJCAI'11*, pages 1535–1540, Barcelona, Catalonia, Spain, 2011. AAAI Press.

[161] Chih-Ping Wei, Huihua Shi, and ChristopherC. Yang. Feature reinforcement approach to poly-lingual text categorization. In DionHoe-Lian Goh, TruHoang Cao, IngeborgTorvik Sølvberg, and Edie Rasmussen, editors, *Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers*, volume 4822 of *Lecture Notes in Computer Science*, pages 99–108. Springer Berlin Heidelberg, 2007.

[162] Erik D. Wiener, Jan O. Pedersen, and Andreas S. Weigend. A neural network approach to topic spotting. In *Proceedings of SDAIR-95, 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 317–332, Las Vegas, US, 1995.

[163] William A. Woods. Conceptual indexing : A better way to organize knowledge. Technical report, Mountain View, CA, USA, 1997.

[164] Ke Wu, Xiaolin Wang, and Bao-Liang Lu. Cross Language Text Categorization Using a Bilingual Lexicon. In *Proceedings of the Third International Joint Conference on Natural Language Processing*, 2008.

[165] Shih-Hung Wu, Tzong-Han Tsai, and Wen-Lian Hsu. Text categorization using automatically acquired domain ontology. In *Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages - Volume 11*, AsianIR '03, pages 138–145, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[166] Yiming Yang. An evaluation of statistical approaches to text categorization. *Inf. Retr.*, 1(1-2) :69–90, May 1999.

[167] Yiming Yang and Xin Liu. A re-examination of text categorization methods. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on*

Research and Development in Information Retrieval, SIGIR '99, pages 42–49, New York, NY, USA, 1999. ACM.

[168] Yiming Yang and Jan O. Pedersen. A comparative study on feature selection in text categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*, ICML '97, pages 412–420, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.