MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE FERHAT ABBAS – SETIF

(UFAS). (ALGERIE)

MEMOIRE

Présenté à la Faculté des Sciences de l'Ingénieur

Département d'Informatique

Pour l'Obtention du Diplôme de

MAGISTER

Option : Sciences et Technologies de l'Information et de la Communication

Par

MR : **Sadik Bessou**

Thème

Analyse de Données Textuelles pour la Classification Automatique par les Techniques de Text Mining, application à la Langue Arabe

Soutenu le : 17/12/2007 devant la commission d'examen :

Dr A. Khababa	MC à l'université de Sétif	Président
Dr M. Saidi	CC à l'université de Sétif	Examinateur
Dr A. Benaouda	CC à l'université de Sétif	Examinateur
Dr M. Touahria	MC à l'université de Sétif	Rapporteur

Remerciements

الحمد لله الذي بنعمته تتم الصالحات

Je tiens à remercier très sincèrement Monsieur Touahria Mohamed qui a accepté de diriger mes travaux de recherches et qui m'a beaucoup enrichi par ses conseils, son expérience et sa disponibilité. Sa confiance et son soutien m'ont été bénéfique.

Mille mercis au professeur Louis Frécon, ses conseils et ses orientations m'ont, à plusieurs reprises, remis sur les rails et m'ont permis de présenter aujourd'hui un travail que j'espère être digne de sa confiance.

Je suis très reconnaissant à Monsieur Khababa Abdellah de me faire l'honneur de présider le jury.

J'exprime toute ma gratitude à l'égard de Monsieur Saidi Mohamed et Monsieur Benaouda Abdelhafid pour l'intérêt qu'ils ont porté à mon travail et pour avoir accepté la charge de juger mon travail.

Je tiens également à remercier les organisateurs et les participants de COSI 2007, et CITALA 2007 pour leurs remarques et conseils.

Je remercie :

- le Professeur Hadj Salah, directeur du C.R.S.T.D.L.A

- le Dr Lasakri, recteur de l'Université de Badji Mokhtar,Annaba

- le Dr Bouhadjra, Directeur de Laboratoire (glossaire arabe), Université de Jijel

- Mr Achite (Laboratoire T.A.L.N, Alger)

- Mr Mebarki de l'Université de Batna

- Mr Margham de l'Université de Sétif

- le directeur de la bibliothèque de l'INI, le personnel de la bibliothèque du centre de calcul au niveau de l'USTHB, ainsi que le personnel de la bibliothèque centrale de l'USTHB, et le personnel de la bibliothèque (CRSTDLA), Messieurs Kaderi Farid d'Alger et Ayoub de Médéa.

Je salue mes amis qui m'ont aidé par leurs encouragements à surmonter les difficultés. Je remercie également Semchedine Moussa, Saadi Abdelahalim, Lakhal Ayat Raouf, Latrech Wahid, Zaoui Karim…

Je pense aux amis de Jijel pour leurs encouragements et leur gentillesse. Je n'oublie pas, bien sûr, les amis de l'UFC Hamedi Tarek, Rout Yacine, Saidi Khaled, Benammar Nazim, Chick Djebar Abdelaziz, Raouag Abdelghani, Abdi Mohand Idir…

Je remercie enfin tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Dédicaces

Je dédie ce travail premièrement, à notre président Abdelaziz Bouteflika.

Je dédie ce travail à tous ceux qui m'aiment et ceux que j'aime :

- A mon père, ma mère et mes frères et sœurs.
- Tous mes enseignants et tous mes étudiants.
- Tous les amis de Sétif, Jijel et Taher.
- ……

Sommaire

Table des figures

Liste des tableaux

Introduction

Depuis environ trente ans, mais surtout durant les dix dernières années, plusieurs recherches menées dans les domaines des sciences humaines et des lettres ont tenté d'intégrer des dimensions informatiques à leurs objectifs. Grâce à ces efforts d'intégration technologique, les sciences humaines ont su développer plusieurs méthodologies et applications d'analyses de textes assistées par ordinateur. Parmi les types d'applications les plus fréquemment cités, on trouve entre autres ceux portant sur l'analyse qualitative et quantitative des données, sur l'analyse de contenu assistée par ordinateur et, de manière plus générale, sur l'analyse des données textuelles.

D'autre part, le domaine des lettres et de la littérature a, lui aussi, été le lieu de réflexions théoriques et d'efforts pratiques visant à permettre l'émergence d'applications informatiques adaptées aux études littéraires. Ainsi, la rencontre entre les littéraires et les informaticiens a fait émerger un nouvel axe de recherche dont les principales manifestations, inspirées de travaux en statistique et en mathématiques, ont pris la forme de méthodologies d'analyse et de logiciels destinés spécifiquement à l'analyse de textes.

Parmi les nouvelles technologies qu'on tend à appliquer aux textes, la *fouille de données*, qui ne se limite pas au traitement des données structurées sous forme de tables numériques mais offre des moyens d'investigation des corpus en langage naturel.

Les données issues des *entrepôts de données* ne sont pas nécessairement toutes exploitables par des techniques de fouille de données. En effet, la plupart des techniques utilisées ne traitent que des tableaux de données numériques rangées sous forme lignes/colonnes. Certaines méthodes sont plus contraignantes que d'autres. Elles peuvent par exemple exiger des données binaires, comme c'est le cas des techniques de recherche de règles d'association. Les données issues de l'entrepôt peuvent être de types différents. On peut y trouver des textes de longueurs variables, de différentes langues et de sujets différents.

La préparation des données est une étape importante, si ce n'est primordiale, du processus d'extraction de connaissances à partir de données. En schématisant, il s'agit de définir au mieux les éléments traités et la représentation utilisée pour l'apprentissage. La qualité (du modèle) de prédiction dépend donc grandement de la qualité de la préparation effectuée en amont.

La préparation consiste à homogénéiser les données et à les disposer en tableau lignes/colonnes, car il s'agit presque toujours de la structure la mieux adaptée à l'exploitation des données. Formellement, chaque ligne/colonne peut être considérée comme un objet vecteur ayant un nombre fixe de composantes. Ce vecteur ligne/colonne sera vu comme un objet mathématique que l'on pourra manipuler selon ses propriétés. Par exemple, si tous les vecteurs lignes sont des points de l'espace euclidien à p dimensions, on pourra faire appel aux techniques de fouille de données basées sur l'algèbre linéaire.

Les données textuelles diffèrent des autres données : sons, images, vidéos,…etc. qui sont facilement transformables en données numériques. Pour les textes il faut tout un processus d'analyse pour aboutir au format exploitable par les techniques de fouille de données, en passant par l'analyse morphologique de la langue, l'analyse lexicale et syntaxique en résolvant les différentes ambiguïtés et problèmes liés aux langues naturelles.

En fait, le pré-traitement suppose un acte de modélisation d'expert. Si l'expert ne définit pas les bonnes transformations ou les bons attributs, il ne verra alors rien dans ses données. L'expert devra par conséquent choisir un canevas pour représenter ses données et éventuellement effectuer une série de transformations pour obtenir des données adaptées aux méthodes d'exploitation.

Le travail présenté dans ce mémoire s'intéresse à l'*analyse de données textuelles*. Le mot analyse de données est vaste ; ici, il s'agit d'effectuer une suite d'opérations d'analyse et de prétraitement sur les données textuelles pour les préparer aux algorithmes de fouille de données en vue d'une application de classification automatique,

Comment peut-on passer des données sous forme de textes aux données numériques exploitables par les techniques de fouille de données ?

La langue arabe est une langue morphologiquement riche et présente de grands défis pour les applications de traitement automatique du langage naturel.

Le traitement automatique de la langue arabe pose des problèmes majeurs:

- Le problème de l'ambiguïté issue de l'absence des voyelles**,** ceci exige des règles morphologiques complexes.
- Le problème de reconnaissance des formes fléchies, car la langue arabe est fortement flexionnelle.

- Et le problème d'absence de travaux publiés sur l'extraction d'information en langue arabe à travers l'utilisation des modèles statistiques de langage.

  Notre objectif est de:

- préparer les textes par une suite d'analyses linguistiques et statistiques
- les représenter dans un format approprié, exploitable par les techniques de fouille de données tout en résolvant les problèmes de reconnaissance de formes fléchies, l'ambiguïté morphologique et syntaxique,...

Notre travail s'intéresse à l'analyse de données textuelles pour les préparer à la classification par les techniques de fouille de données. Il comporte cinq chapitres et une brève conclusion.

Le premier chapitre donne une présentation générale de la fouille de données (les définitions, objectifs et étapes d'un processus de fouille de données et les tâches principales)

Le deuxième chapitre est consacré aux méthodes de fouille de données, en donnant pour chaque méthode la définition, le principe, les domaines d'applications et les limites, en ciblant chaque fois les méthodes qui peuvent être appliqué pour la classification.

Le troisième chapitre donne une présentation générale sur la fouille de textes en précisant les différences avec la fouille de données

Le quatrième chapitre donne en détail les différents traitements d'analyse (morphologique, lexicale, syntaxique, fréquentielle) des textes en Langue Arabe.

Enfin le cinquième chapitre décrit l'opération de classification et donne une comparaison entre différents algorithmes et travaux effectués dans le domaine.

Chapitre 1. La Fouille des Données

La fouille de données (ou *data mining*) est prédit pour être " un des développements les plus révolutionnaires de la prochaine décennie", selon le webzine technologique *ZDNET News* (2001) [01]. En fait, selon le Gartner Group, la *MIT TechnologyReview* a choisi la fouille de données comme un des dix technologies émergentes qui changeront le monde [02].

Le domaine de la fouille de données n'est pas nouveau puisqu'il s'appuie sur des théories statistiques des années 70. La nouveauté de la fouille de données réside dans le fait qu'elle tire le meilleur de chaque méthode et les couplent entre elles pour arriver à des résultats plus que satisfaisants qui devraient changer les habitudes des êtres humains !

1.1. Définitions

La fouille de données est un sujet brûlant, il dépasse aujourd'hui le cercle de la communauté scientifique pour susciter un vif intérêt dans le monde des affaires. La littérature spécialisée et la presse ont pris le relais de cet intérêt, avec une pléthore de définitions générales de la fouille de données.

Nous en avons sélectionné quelques une :

- "Data mining is the *process of discovering meaningful new correlations, patterns and trends* by sifting through large amounts of data stored in repositories, using pattern recognition technologies as well as statistical and mathematical techniques"[03].
- "Data mining is the analysis of (often large) observational data sets *to find unsuspected relationships* and *to summarize the data* in novel ways that are both understandable and useful to the data owner" [04].
- "Data mining is an interdisciplinary field bringing together techniques from machine learning, pattern recognition, statistics, databases, and visualization to address the issue of *information extraction from large data bases*" [05].
- "Littéralement *fouille de données* ou *fourrage de données* est l'application des techniques d'analyse des données et d'Intelligence Artificielle à l'exploration et l'analyse sans a prio-

ri de grandes bases de données, en vue d'en extraire des informations pertinentes pour l'entreprise, utilisées en particulier dans des systèmes d'aides à la décision " [06].

- " L'extraction d'informations originales auparavant inconnues et potentiellement utiles à partir de données " [11].
- "La démarche de nouvelles corrélations, tendances et modèles par le tamisage d'un large volume de données" [John Page].
- d'autres plus poétiques comme "Torturer l'information jusqu'à ce qu'elle avoue" [Dimitri Chorafas].

En 2002, l'ex-président Bill Clinton a mentionné que peu après les événements du 11 septembre 2001[1], les agents du FBI ont examiné de grandes quantités de données et ont trouvé que les noms de cinq terroristes étaient dans la base de données. Clinton a conclu qu'ils doivent exploiter ce type de données [07].

" Plusieurs compagnies ont implémenté la stratégie de l'entrepôt de données et commencé maintenant à regarder que ce qu'ils peuvent faire avec toutes ces données", dit Dudley Brown, le dirigeant partenaire de BridgeGate LLC.

1.2. Pour quoi la Fouille de Données ?

Beaucoup de données sont rassemblées depuis les premières Bases de Données documentaires (1957). Cependant, qu'a-t-on appris de toutes ces données ? Quelles connaissances pouvons nous tirer de tous ces renseignements ?

Au début de 1984, dans son *Megatrends* [08], John Naisbitt a observé que "nous nous noyons dans les renseignements mais nous sommes affamés de connaissances "[2]. Le problème aujourd'hui n'est pas qu'il n'y a pas assez de données et de renseignements. Nous sommes, en fait, inondés de données dans la plupart des domaines. Le problème est plutôt qu'il n'y a pas assez d'analystes humains disponibles, compétents et synthétiques pouvant traduire toutes ces données en connaissances utiles pour divers domaines.

Les outils de fouille de données regroupent des nouveaux produits du domaine de l'aide à la décision, ces produits résultent des superbases de données (*Data Warehouse*) qui contiennent l'ensemble des informations d'un organisme sous forme harmonisée et accessible. Dans

---

[1] En fait, Bush a reçu auparavant divers avertissements de ses 14 services de renseignement, mais, submergé par les informations, il n'a perçu aucun de ces avertissements.

[2] *Nam et ipsa scientia potestas est / Knowledge itself is a power* (Francis Bacon, 1561-1626, Religious meditations, Of heresies)

les entreprises centralisées, l'expert consacrait beaucoup de temps à extraire les connaissances et les analyser suite à une interrogation d'une base de données. De nos jours, une simple requête d'une base de données peut renvoyer des milliers d'enregistrements à l'expert, cela est dû à la croissance effrénée des volumes d'informations. L'expert doit cependant répondre vite et bien[3] pour satisfaire aux contraintes qui lui sont imposées.

La formalisation et l'industrialisation du processus de création d'expertise sont d'autant plus importantes que l'ancienne démarche qui consistait à augmenter le nombre d'experts est devenue obsolète ; car la formation des experts est très coûteuse en temps et en argent – et elle suppose bien sûr un nombre d'experts suffisant par rapport à la quantité d'information, ce qui n'est pas toujours le cas.

La fouille de données propose des outils d'automatisation de certaines phases très pointues de l'analyse qui était jusque là, réservée aux spécialistes des bases de données et de statistiques ; la fouille de données participe alors d'une manière très significative à l'assouplissement du travail de l'expert et de l'analyste [06].

Les outils de fouille de données permettent aux responsables de produits, aux techniciens de maintenance ou aux contrôleurs de gestion d'être moins dépendant des spécialistes de l'analyse de données pour résoudre leurs problèmes quotidiens (cibler des clients, décrire une clientèle, identifier une machine mal réglée, prévoir les réapprovisionnements, établir des prévisions budgétaires,…)

1.3. A quoi sert la Fouille de Données ?

D'une manière générale, la fouille de données a une raison d'être partout où les informations sont nombreuses et/ou les processus peuvent être améliorés, c'est-à-dire dans presque tous les secteurs d'activités de service aux clients, notamment dans les domaines du marketing, des bases de données de fidélisation du client, ou de détection de fraudes, entre autres dans les secteurs du crédit et des télécommunications. L'application de la fouille de données à l'optimisation d'implantation de réseaux de distribution ou de merchandising, aussi qu'un support aux utilisations ou au contrôle de qualité, est également de plus en plus fréquente.

Aujourd'hui, la fouille de données se répand particulièrement dans les secteurs qui, par leur métier détiennent de nombreuses informations économiques individualisées (vente par correspondance, grande distribution, téléphonie et banque en tête) :

---

[3] La difficulté est dans la conjonction rapidité /pertinence

- identification des prospects les plus susceptibles de devenir des clients ;
- identification des clients les plus rentables et concentration sur eux des efforts commerciaux ;
- détermination des profils de consommateurs, du panier de la ménagère, de l'effet des soldes ou de la publicité, dans la grande distribution ;
- adaptation de la communication marketing à chaque segment de clientèle ;
- recherche des caractéristiques des clients à risque afin d'adapter la tarification ;
- prévention des impayés et détection automatique et en temps réel de la fraude ;
- personnalisation des pages du site web de l'entreprise, en fonction du profil de l'internaute ;
- analyse des lettres de réclamation de la clientèle (données textuelles) ;
- veille technologique (« fouille de texte » des études, articles spécialisés de la presse technique ou économique, dépôts de brevets). [06]

De plus en plus, la fouille de données doit cesser d'être l'assemblage de techniques d'analyse de données, pour devenir un véritable processus industriel, avec ses impératifs de disponibilité, d'automaticité et fiabilité, sans oublier la simplicité pour l'utilisateur.

Les techniques de fouille de données sont plus complexes que de simples statistiques descriptives. Au delà de statistiques et d'analyses des données traditionnelles (analyse factorielle, classification hiérarchique, analyse discriminante,…) elles s'appuient sur [06] :

- des algorithmes sophistiqués (analyse relationnelle, algorithmes génétiques, ..),
- des outils d'Intelligence Artificielle et de connexionisme (les réseaux de neurone),
- la théorie de l'information (arbres de décision).

1.4. Les différents types de données

Les tables de l'entrepôt des données sont formées d'enregistrements dont les champs sont de divers types, qu'il est important de préciser.

En effet, la plupart des méthodes sont sensibles à la nature des données manipulées.

On distingue les données **quantitatives, qualitatives et textuelles**.

Pour les **données quantitatives**, on distingue souvent

- les **données entières** comme les codes numériques, le nombre d'enfants, le nombre de produits achetés,

- les **données continues,** comme un niveau d'eau, une température, une longueur d'onde, une concentration, un pH, ou, *si on veut*, un salaire, un revenu moyen, le montant des achats effectués (exprimables de fait par un nombre entier de centimes).

Les **données qualitatives** (symboliques, catégoriques ou catégorielles) sont des données dont l'ensemble des valeurs est fini. Ces valeurs sont le plus souvent alphanumériques ; elles peuvent être numérisées par des codes distincts, mais non comme des quantités arithmétiques de plein statut.

Exemple : la catégorie socioprofessionnelle, le numéro de département, les données issues de la discrétisation de données continues (1 : solde moyen < 1000 ; 2 : 1000 ≤ solde moyen < 5000 ; ...)

Le système N.O.I.R. proposé par Stevens (1951) pour les besoins de la psychométrie, hiérarchise plus finement 4 classes de données élémentaires, du plus général au plus traitable :

Données nominales (N), discrètes, numériques ou symboliques : ici les nombres ne sont que des indices de catégorie, et 2 ne signifie rien par rapport à 1. Les seules opérations possibles sont les comparaisons = et ≠.

Exemple: client : 0 : normal ; 1 : bon ; 9 : mauvais payeur.

Données ordinales (O) discrètes, numériques ou symboliques : données nominales qui reflètent un ordre, par exemple

- l'attribution de rangs (1, 2, 3, etc) aux individus ou objets d'un groupe en regard d'une ou plusieurs caractéristiques ordonnées : rangs scolaires, rangs au fil d'arrivée dans les sports…
- la discrétisation d'une valeur continue (1 : solde moyen < 1000 ; 2 : 1000 ≤ solde moyen < 5000 ; ...)
- une échelle[4] symbolique (réponses non/plutôt non/plutôt oui/oui ou jamais/ rarement/ parfois/ souvent/ toujours ou blanc/clair/foncé/noir)

Les opérations = ,≠, <, > s'appliquent à ce type de données, et souvent max/min.

Données à intervalles égaux (I) : valeurs ordonnées dont la différence a un sens par rapport au trait mesuré. De fait, dans ce type de données, les intervalles entre les nombres représentent une unité de mesure spécifique ; cependant, la valeur 0 est arbitraire.

[4] Au sens d'ordre total

Exemple : les échelles de températures usuelles. Il faut la même quantité de chaleur pour passer de 32°C à 33°C qu'il en faut pour passer de 20°C à 21°C Par conséquent, on ne peut établir de proportions entre ces données. En effet, à 28°C, on ne peut pas dire qu'il fait 2 fois plus chaud qu'à 14°C. De plus, 0°C ne signifie pas l'absence de chaleur, simplement le point où l'eau gèle ; et $0°C \neq 0°F$.

Données de ratio ou proportionnelles (R) : données à intervalles égaux, avec un 0 non arbitraire, et telles que les quotients ont un sens :

Exemples : on peut dire qu'une personne de 200 kg pèse 2 fois plus qu'une personne de 100 kg ; température absolue (°Kelvin), dont les quotients sont utiles en thermodynamique ; degrés d'angles…

Remarque. Si le temps est ici de classe I, les durées de classe R.

Les données textuelles ont pour valeurs des textes non codés, écrits en langue naturelle : lettres de réclamation, rapport, dépêches,… [5]. Un texte peut, pour certaines applications, être résumé par une suite plus limitée de données élémentaires, par exemple un n-uplet constitué du nombre d'occurrences dans le texte de mots-clés [09].

L'analyse des données textuelles nécessite un mélange de linguistique et de statistiques. Il est à noter que les données textuelles en langage naturel contiennent des ellipses, des abréviations (plus ou moins personnelles), [06] des fautes d'orthographe ou de syntaxe et des ambiguïtés syntaxiques ou sémantiques et même pragmatiques (termes dont le sens dépend d'un contexte implicite, non facilement détectable automatiquement).

1.5. Le processus de Fouille de Données

Dans ce domaine encore appelé KDD (Knowledge Discovery in Data base), diverses démarches peuvent être observées [06] [12] [13], décrivant les étapes d'un projet de fouille de données. Ces démarches se ressemblent dans plusieurs parties, nous avons choisi parmi elles CRISP-DM ; processus hiérarchique avec un ensemble de tâches décrit par quatre niveaux d'abstraction (du général au spécifique) : les phases, tâche générique, tâche spécialisé et instances du processus.

[5] Par rapport au système NOIR, on les classeraient N* : suites (structurées) de valeurs nominales.

Il est applicable à divers problèmes ; mêmes les petites investigations bénéficient de l'utilisation de CRISP-DM [15]. CRISP-DM a obtenu 51% des voix dans le site *kdnuggets* comme le meilleur processus utilisé [14]

CRISP-DM, le Cross Industry Standard Process for Data mining

Développé en 1996 par des analystes représentant Daimler-Chrysler, SPSS, et NCR, CRISP fournit un processus générique approprié pour les stratégies de fouille de données pour la résolution de problèmes généraux.

Selon CRISP–DM [10], un projet quelconque de fouille de données a un cycle de vie de six phases, illustré dans la figure 1.1. Notons que la séquence des phases est adaptative : la prochaine phase dans la séquence dépend souvent des résultats issus de la phase précédente. Les dépendances les plus considérables entre phases sont indiquées par les flèches.

Par exemple, supposons que nous sommes dans la phase de modélisation. Selon le comportement et les caractéristiques du modèle, nous pouvons revenir à la phase de la préparation des données pour affinage supplémentaire avant de passer à la phase d'évaluation du modèle.

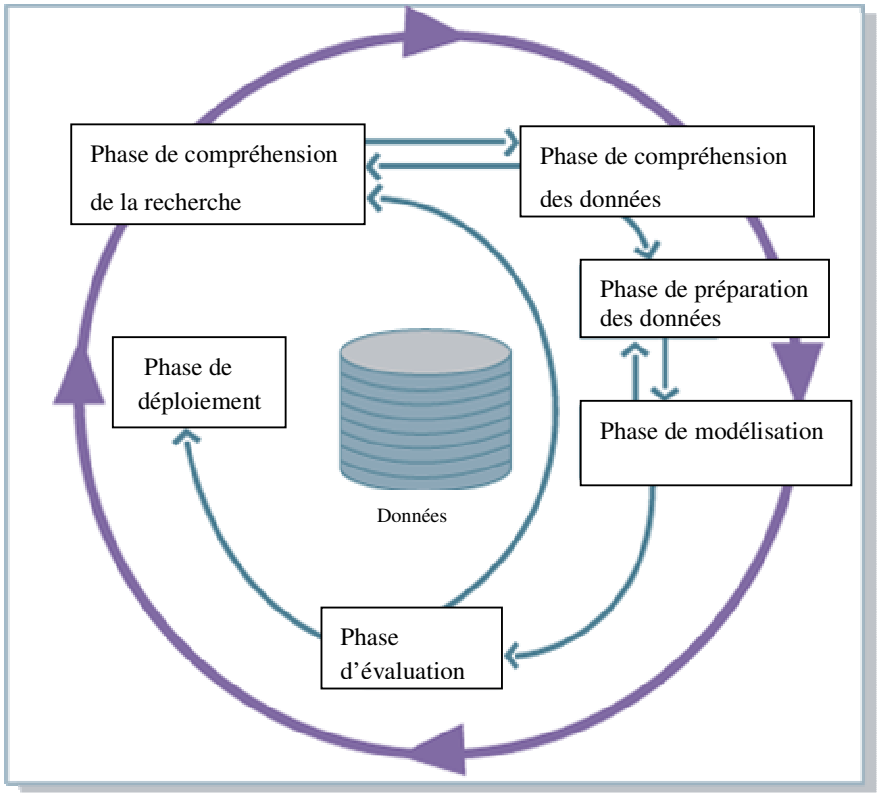


Figure1.1 Processus de CRISP-DM

La nature itérative de CRISP est symbolisée par le cercle externe dans la figure 1.1. Souvent, la solution à un problème particulier mène à des questions supplémentaires intéressantes, qu'on peut attaquer en utilisant le même processus général. Les leçons apprises des projets passés devraient toujours être considérées dans de nouveaux projets. Bien qu'il soit concevable que les questions rencontrées pendant la phase d'évaluation puissent renvoyer l'analyste à des phases antérieures pour amélioration, pour la simplicité nous montrons seulement la boucle la plus commune.

Nous résumons dans ce qui suit chaque phase.

Les phases de CRISP–DM

Il faut préciser en premier lieu le *domaine* et *l'objet d'étude de la recherche* avant d'entamer les différentes phases :

1. **Phase de compréhension de la recherche /** ***Business understanding phase***

a. Énoncer les objectifs du projet et exigences quant à l'affaire ou unité de la recherche clairement dans son ensemble.

b. Traduire ces buts et restrictions dans la formulation d'une définition du problème de fouille de données (Formulation de la problématique).

c. Préparer une stratégie préliminaire pour accomplir ces objectifs.

2. **Phase de compréhension des données /** ***Data understanding***

a. Rassembler les données.

b. Utiliser l'analyse exploratoire des données pour se familiariser avec les données et découvrir des perspectives initiales.

c. Évaluer la qualité des données.

d. Sélectionner, éventuellement des sous-ensembles intéressants qui peuvent contenir des modèles possibles.

3. **Phase de préparation des données /** ***Data preparation phase***

a. Préparer les données initiales pour l'ensemble des données qui sera utilisé pour les phases suivantes. Même intensive, cette phase est difficile.

b. Sélectionner les cas et variables que vous voulez analyser et qui sont appropriés pour votre analyse.

c. Exécuter des transformations sur certaines variables, (si besoin).

d. Nettoyer les données afin qu'elles soient prêtes pour les outils du modélisation.

4. **Phase de modélisation /** ***Modeling phase***

a. Sélectionner et appliquer des techniques de modélisation appropriées.

b. Identifier les paramètres du modèle pour ajuster la modélisation.

c. Différentes techniques peuvent être utilisées pour le même problème[6].

d. Si nécessaire, rebouclez sur la phase de la préparation des données pour aligner la forme des données aux exigences spécifiques d'une technique particulière.

5. **Phase d'évaluation /** ***Evaluation phase***

a. Évaluer un ou plusieurs modèles repérés dans la phase de modélisation pour mesurer la qualité et l'efficacité avant de les exploiter.

b. Déterminer si le modèle accomplit l'ensemble des objectifs assignés dans la première phase.

c. Établir si quelques facettes importantes n'ont pas été sous-estimée/mésestimée

d. Prendre une décision quant à l'usage des résultats.

6. **Phase de déploiement /** ***Deployment phase***

a. Faites usage des modèles créés : la création du modèle ne signifie pas l'achèvement d'un projet.

b. Exemple d'un déploiement simple : Produire un rapport.

c. Exemple d'un déploiement plus complexe : Rendre effectif un processus parallèle dans un autre département [10].

d. Pour les entreprises, le client emporte souvent le déploiement basé sur le modèle.

1.6. Les Tâches de Fouille de Données

Les principales tâches qui peuvent être accomplies par la fouille de données sont les tâches de *description – estimation / prédiction – classification – segmentation – association*.

[6] Çà peut même être une technique de preuve si le résultat semble improbable.

1.6.1 La description

Les chercheurs et analystes essaient simplement de trouver des chemins pour décrire des modèles et des tendances qui se dessinent dans les données.

Les descriptions des modèles et tendances suggèrent souvent des explications.

Les modèles de fouille de données devraient être aussi transparents que possible, les résultats des modèles de fouille de données devraient décrire des tendances claires qui amènent à des interprétations et des explications intuitives.

Quelques méthodes de fouille de données sont plus convenables que d'autres pour des interprétations transparentes, par exemple les arbres de décision fournissent des explications intuitives des résultats. D'un autre côté, les réseaux de neurones sont comparativement opaque à des non-spécialistes, dû à la non linéarité et la complexité du modèle [13].

Une description de qualité peut souvent être obtenue par l'analyse de données transcrite en représentations graphiques permettant de dégager modèles et tendances.

1.6.2 L'estimation

Semblable à la classification, à l'exception que la variable cible est numérique plutôt que catégorique. Les modèles sont construits en utilisant " des enregistrements complets " qui fournissent la valeur de la variable cible aussi bien que les valeurs des variables de prédiction. Puis, l'estimation de la valeur de la variable cible est faite pour les nouvelles observations, basées sur les valeurs de prédiction.

On trouve l'estimation dans par exemple:

- Le montant d'argent aléatoirement dépensé par une famille choisie dans une occasion donnée ;
- La notification d'un candidat à un prêt[7] ; cette estimation peut être utilisée pour attribuer un prêt (classification), par exemple, en fixant un seuil d'attribution [09] ;
- L'estimation de la baisse du pourcentage du mouvement rotatif pour un joueur d'une Ligue du Football Nationale qui court avec une blessure du genou ;
- L'estimation du nombre de points par jeu que Monsieur X marquera sous certaines conditions ;
- L'estimation de la moyenne d'un étudiant basé sur sa moyenne l'année passée.

[7] Voir aussi Schärlig sur les choix multicritères des banquiers (aux PPUR).

L'analyse statistique fournit plusieurs résultats vénérables, par les méthodes d'estimation. Ceux-ci incluent l'estimation de l'intervalle de confiance, la régression linéaire simple, la corrélation et la régression multiple [13].

1.6.3 La prédiction

Similaire à la classification et l'estimation, sauf que les résultats de la prédiction concernent le futur. En général, les valeurs connues sont historiées. On cherche à prédire la valeur future d'un champ.

On trouve l'estimation pour [13] :

- Prédire le prix d'un stock trois mois dans le futur ;
- Prédire la variation en pourcentage pour les morts de la circulation l'an prochain si la limite de vitesse est augmentée[8] ;
- Prédire le vainqueur d'un sport donné, basé sur une comparaison de statistiques de performances ;
- Prédire si une molécule particulière dans la découverte d'une drogue mènera à une nouvelle drogue avantageuse pour une compagnie pharmaceutique.

Les méthodes et techniques utilisées pour la classification et l'estimation peuvent aussi être utilisées, sous certaines circonstances appropriées, pour la prédiction. Celles-ci incluent

- les méthodes statistiques traditionnelles d'estimation du point et estimations de l'intervalle de confiance, la régression linéaire simple, la corrélation et la régression multiple ;
- comme les méthodes propres aux fouilles de données telles que les arbres de décision, les réseaux de neurones, et la méthode des k-plus-proches-voisins.

1.6.4 La Classification

Dans la classification, il y a une variable cible catégorique.

Le modèle de fouille de données examine un ensemble d'enregistrements, chaque item contient des informations perspectives sur la variable cible aussi bien qu'un ensemble d'entrée ou variables de prédiction.

---

[8] Avec ou sans (a) fluctuation du prix de l'essence (b) apparition d'une technologie comme l'ABS (qui augmente la sécurité comme une baisse de 10 km/h) (c) modulation du prix des contraventions.

Supposons que le chercheur souhaite classer les supports du revenu d'une personne qui n'est pas actuellement dans la base de données, basé sur d'autres caractéristiques associées avec cette personne, tel que l'âge, le sexe, et l'occupation. C'est une tâche de classification, très appropriée aux méthodes et techniques de fouille de données.

L'algorithme se déroulerait en gros comme suit :

En premier, l'algorithme examine l'ensemble des données contenant les variables de prédiction et la variable cible associée. Dans cette phase, l'algorithme " apprend " quelles combinaisons de variables sont associées avec quelle valeur cible. Cet ensemble de données est appelé *l'ensemble d'apprentissage*.

Puis l'algorithme examine de nouveaux items, pour lesquels il n'y a pas d'information disponible sur la variable cible. Basé sur les classifications dans l'ensemble d'apprentissage, l'algorithme assigne des classifications aux nouveaux items.[9]

On trouve la classification pour [09]:

- Déterminer si une transaction avec une carte du crédit particulière est frauduleuse ;
- Placer un nouvel étudiant dans une filière particulière quant aux prérequis ;
- Déterminer si une candidature à hypothèque est un bon ou mauvais risque du crédit ;
- Attribuer ou non un prêt à un client ;
- Diagnostiquer si une maladie particulière est présente ;
- Déterminer si une volonté a été écrite par le défunt réel, ou frauduleusement par quelqu'un d'autre ;
- Identifier si un comportement financier ou personnel indique ou pas une menace terroriste possible ;
- Attribuer un sujet principal à un article de presse,...

Les graphiques sont utiles pour mettre en perspective les relations entre deux ou trois dimensions (variables ou facteurs).

Mais quelquefois les classifications ont besoin d'être basées sur beaucoup de variables de prédiction, en exigeant une intrication d'un grand nombre de dimensions. Par conséquent, nous avons besoin de nous tourner vers des modèles plus sophistiqués pour exécuter nos tâ-

[9] Il y a souvent une phase de validation intermédiaire ; cf Zighed.

ches de classification [13]. Les méthodes les plus utilisées pour la classification sont alors les k-plus-proches-voisins, les arbres de décision, et les réseaux de neurones.

1.6.5 La segmentation

La segmentation fait référence au groupement des enregistrements, les *observations*, dans des classes d'objets semblables. Elle consiste à former des groupes (clusters) homogènes à l'intérieur d'une population. Pour cette tâche, il n'y a pas de classe à expliquer ou de valeur à prédire définie a priori, il s'agit de créer des groupes homogènes dans la population (représentée par l'ensemble des enregistrements). Il appartient ensuite à un expert du domaine de déterminer l'intérêt et la signification des groupes ainsi constitués. Cette tâche est souvent effectuée avant les précédentes pour construire des groupes en vue des tâches de classification ou d'estimation [09].

Un segment est une collection d'enregistrements qui sont semblables l'un à l'autre, et dissemblable aux enregistrements dans les autres segments. La segmentation diffère de la classification, car il n'y a aucune variable cible pour segmenter ? on cherche des groupements naturels ? La tâche de segmentation n'essaie pas de classer, estimer, ou prédire la valeur d'une variable cible. Cependant, les algorithmes de segmentation cherchent à répartir les données en sous-groupes relativement homogènes, où la ressemblance des enregistrements dans le sous groupe est maximisée et la ressemblance à des enregistrements à l'extérieur du sous groupe est minimisée.

[13] indique comme segmentations :

- le marketing ciblé d'un produit donné pour des affaires de petite capitalisation ;
- la segmentation des comportements financiers ;
- la réduction de dimensions quand l'ensemble des données a des centaines d'attributs.

La segmentation est souvent exécutée comme étape préliminaire dans un processus de fouille de données ; les segments résultants seront utilisés comme entrées dans différentes techniques comme les réseaux de neurones, la segmentation hiérarchique , *k*-means , les réseaux de Kohonen.

1.6.6 L'association

La tâche d'association pour la fouille de données consiste à trouver quels attributs "vont ensemble". Connue dans le monde des affaires comme analyse d'affinité ou analyse du panier de la ménagère, la tâche d'association cherche à découvrir des règles pour mesurer le

rapport entre deux attributs ou plus. *"Cette tâche peut être effectuée pour identifier des opportunités de vente croisée et concevoir des groupements attractifs de produit. C'est une des tâches qui nécessite de très grands jeux de données pour être effective"* [09]. Les règles d'association cherchées sont de la forme *Si< antécédent>, Alors <conséquent>*, avec une mesure du support et de confiance associée à la règle.

Par exemple, un supermarché particulier a eu 1000 clients qui font le marché un jeudi de nuit, 200 ont acheté le produit A, et sur ces 200 qui ont acheté A, 50 achètent le produit B. Donc, la règle d'association serait " *Si achète A, Alors achète B* " avec un support de 200/1000 = 20% et une confiance de 50/200 = 25%.

On recherche l'association dans divers cas :

- Enquête sur la proportion de souscripteurs au plan du téléphone portable d'une compagnie qui répond à une offre d'une amélioration du service positivement.
- Examen de la proportion d'enfants à qui les parents font la lecture et qui eux-mêmes sont des bons lecteurs.
- Prédire la déchéance dans les réseaux des télécommunications.
- Trouver quel sont les articles dans un supermarché qui sont achetés ensemble et quels articles qui ne sont jamais achetés ensemble

Il y a plusieurs algorithmes pour la génération des règles d'association : cf *the a priori algorithm* dans [13].

Chaque tâche parmi les tâches précédentes est accomplie par une ou plusieurs techniques de fouille de données résumées dans le tableau suivant :

Tableau 1.1 Tâches de Fouille de données et Algorithmes correspondants

tâche	techniques pour cette tâche
description	Analyse de données
estimation	approches statistiques
prédiction	approches statistiques
classification	• K plus proches voisins • arbres de décision • réseaux de neurones
segmentation	K-means et segmentation hiérarchique réseaux de Kohonen
association	règles d'association

Les principaux algorithmes de fouille de données et d'analyse de données se répartissent en deux grandes familles de techniques (tableau 1.2).

Les techniques descriptives (exploratoires), sans variables privilégiées, visent à mettre en évidence des informations présentes mais cachées par le volume de données (c'est le cas des segmentations de clientèle et des recherches d'association de produits sur les tickets de caisse), elles s'intéressent aux individus (au sens statistique large : client, produits, des transactions,…) et en particulier aux groupes homogènes qu'ils peuvent former.

Les techniques prédictives (explicatives) (une variable à expliquer) visent à extrapoler de nouvelles informations à partir des informations présentes (c'est le cas du scoring), elles s'intéressent aux variables et aux relations entre elles [06].

Tableau 1.2 Algorithmes descriptifs et Algorithmes prédictifs

Algorithmes descriptifs	Algorithmes prédictifs
-ACP -ACM -K-means / nuées dynamiques /segmentation hiérarchique/ segmentation neuronale - réseaux de Kohonen - recherche d'association	- Arbres de décision - réseaux à apprentissage supervisé - réseaux à fonction radiale - régression linéaire - régression logistique - analyse discriminante - K plus proches voisins

1.7. Conclusion

L'exploitation des données est devenue un enjeu stratégique à la fois commercial, industriel, et scientifique Mais la fouille de données devra relever plusieurs défis :

- Les outils devront être plus accessibles, et simples d'utilisation.
- Les techniques d'extraction devront évoluer afin de pouvoir traiter rapidement et précisément des bases de données toujours plus volumineuses.

Parallèlement, la fouille de données devra devenir capable de traiter plusieurs types de données (textuelle, image, vidéo …) simultanément.

Pour plus de clarté, cette présentation n'a parfois qu'effleuré le sujet, sans aborder d'autres aspects. Résoudre un problème avec un processus de fouille de données impose généralement l'utilisation d'un grand nombre de méthodes et algorithmes différents, objets du prochain chapitre.

Chapitre 2. Les Méthodes

La Théorie des Préférences montre qu'en général, dès qu'une comparaison utilise deux critères ou plus, il n'existe pas de méthode plus performante que toutes les autres.

Par conséquent, à tout jeu de données et tout problème correspond une ou plusieurs méthodes intéressantes. Le choix se fera en fonction :

- de la tâche à résoudre,
- de la nature (type) et de la disponibilité des données,
- des connaissances et des compétences disponibles,
- de la finalité du modèle construit ; pour cela, les critères suivants sont importants : complexité de la construction du modèle, complexité de son utilisation, ses performances, sa pérennité, et, plus généralement :
- de l'environnement de l'entreprise.

Nous présentons dans ce qui suit les méthodes qui nous sembles intéressantes dans le domaine de la classification.

2.1. Les k plus proches voisins

2.1.1 Définition

La méthode PPV est une méthode de raisonnement à partir de cas. Elle part de l'idée de prendre des décisions en recherchant en mémoire un ou des cas similaires déjà résolus. Elle est simple à comprendre sur le plan mathématique ; elle a fait ses preuves et est souvent utilisée en comparaison avec des méthodes plus récentes [81].

Contrairement aux autres méthodes de classification qui seront étudiées dans les sections suivantes (arbres de décision, réseaux de neurones), il n'y a pas d'étape d'apprentissage consistant en la construction d'un modèle à partir d'un échantillon d'apprentissage. Mais la méthode suppose la définition d'une similarité ou d'une distance entre cas. Bien que la méthode ne produise pas de règle explicite, la classe attribuée à un exemple peut être expliquée en exhibant les plus proches voisins qui ont amené à ce choix (Figures 2.1, 2.2).

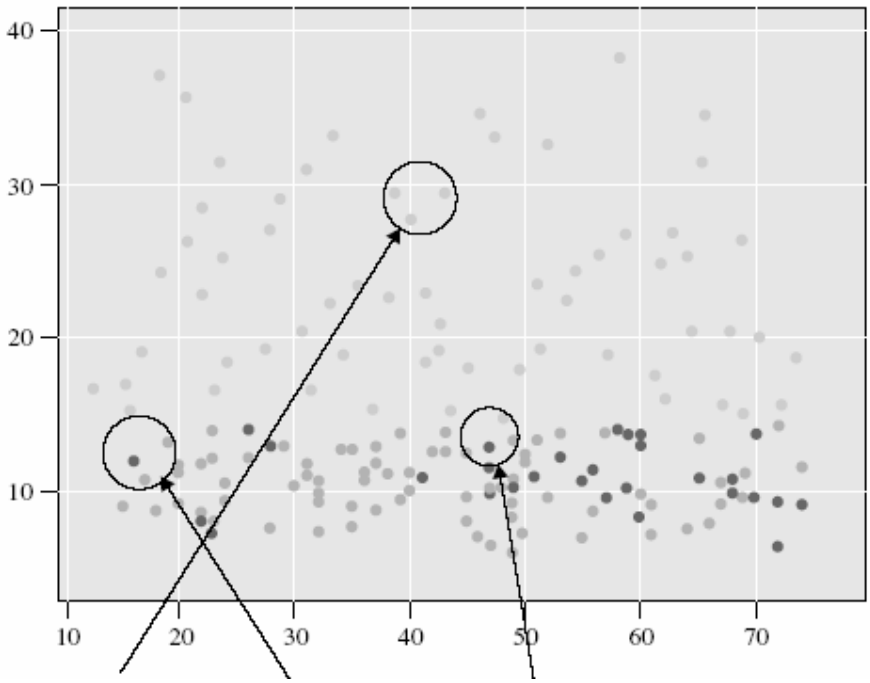


Figure 2.1 Graphe de proportion de sodium/potassium par rapport à l'âge [13].

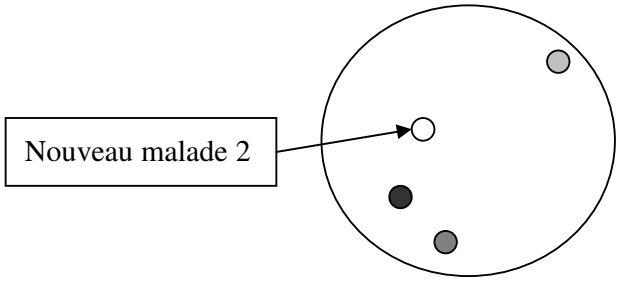


Figure2.2 Les 3 plus proches voisins du malade 2 [13].

2.1.2 Principe

C'est l'échantillon d'apprentissage, associé à une fonction de distance et d'une fonction de choix de la classe en fonction des classes des voisins les plus proches, qui constitue le modèle.

La fonction distance la plus commune est la distance Euclidienne (ou distance à vol d'oiseau) donnée par :

$$d(x, y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

Où $\mathbf{x} = (x_1, x_2, \ldots, x_m)$, et $\mathbf{y} = (y_1, y_2, \ldots, y_m)$ représentent les *m valeurs* d'attributs des deux enregistrements (Figure 2.3).

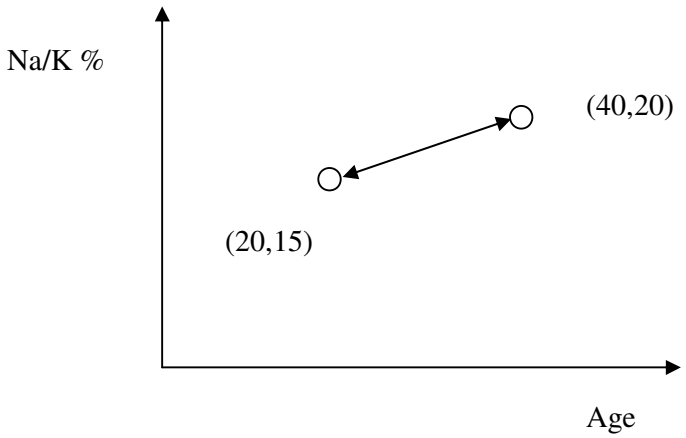


Figure 2.3. Calcul d'une distance Euclidienne.

Cependant, quand on évalue la distance, certains attributs qui ont de grandes valeurs peuvent l'emporter sur d'autres attributs à valeur sur une plus petite échelle. Pour éviter ceci, l'analyste des données devrait s'assurer de normaliser les valeurs des attributs. Il peut également utiliser diverses distances plus spécifiques.

Pour les variables nominales, la distance Euclidienne n'est pas appropriée. Nous utilisons alors la *distance discrète* définie par

$$d(x_i, y_i) \equiv \text{(si } x_i = y_i \text{ alors 0 sinon 1)}$$

voire *la différence des rangs pour des variables ordinales.*

Ces distances peuvent être combinées par sommation pour un nombre quelconque de critères discrets.

2.1.3 La fonction de combinaison

Maintenant que nous avons une méthode pour déterminer quels enregistrements sont très semblables au nouvel enregistrement non classé, nous avons besoin d'établir comment combiner ces enregistrements semblables pour fournir une décision de la classification pour

un nouvel enregistrement. Pour cela nous avons besoin d'une fonction de combinaison. La fonction de combinaison la plus simple est ***le vote simple***.

1 Vote simple / Simple Unweighted Voting

- Avant de parcourir l'algorithme, on décide la valeur de k, c'est-à-dire combien d'enregistrements auront une voix dans la classification d'un nouvel enregistrement.
- Puis, on compare le nouvel enregistrement aux k voisins les plus proches, c'est à dire, aux k enregistrements qui sont à une distance minimum du nouvel enregistrement (pour la distance retenue), qui reçoivent chacun une voix.
- La classe du nouvel enregistrement est alors la classe ayant le maximum de voix.

2 Vote pondéré / Weighted Voting

Est-il judicieux qu'un enregistrement X ait la même voix qu'un enregistrement Y plus proche au nouvel enregistrement? Peut-être pas. L'analyste peut choisir d'appliquer un ***Vote Pondéré***[10], où les voisins les plus proches ont une plus grande voix dans la décision de la classification que les voisins plus distants.

Remarque

Une fois que nous commençons à pondérer les enregistrements, aucune raison théorique interdit d'augmenter k arbitrairement afin que tous les enregistrements existants soient inclus. Cependant, cela s'oppose à la considération pratique d'un calcul très lent pour réévaluer les poids de tous les enregistrements chaque fois qu'un nouvel enregistrement a besoin d'être classé.

2.1.4 La mesure de la pertinence des attributs

Allongement des axes

On constate que ce ne sont pas tous les attributs qui sont pertinents pour la classification.

Dans les arbres de décision, par exemple, seuls les attributs utiles à la classification (attributs discriminants) sont considérés. Dans l'algorithme des k plus proches voisins, les distances sont par défaut calculées sur tous les attributs. Il est possible, par conséquent, pour des enregistrements pertinents qui sont proches du nouvel enregistrement pour toutes les va-

[10] En raison inverse de la distance.

riables importantes, mais distant du nouvel enregistrement pour les attributs non pertinents, d'avoir une grande distance au nouvel enregistrement, et par conséquent qu'ils ne soit pas considérés pour la classification. Les analystes peuvent restreindre l'algorithme aux champs connus pour être importants pour classer de nouveaux enregistrements, ou au moins dissimuler l'algorithme aux champs sans rapports connus. Ou bien, plutôt que restreindre des champs a priori, l'analyste des données peut préférer indiquer quels champs sont de plus ou moins importance pour classer la variable cible.

Cependant dans les problèmes du monde réel, les axes allongés peuvent mener aux classifications plus exactes, représente une méthode pour mesurer la pertinence de chaque variable dans la décision de classification. C'est particulièrement important que les classifications rares soient représentées suffisamment, donc l'algorithme ne fait pas seulement prédire les classifications communes.[11]

Par conséquent, l'ensemble des données aurait besoin d'être équilibré, avec un suffisamment grand pourcentage des classifications moins communes. Une méthode d'équilibrage est de réduire la proportion d'enregistrements relevant des classes communes, en retenant seulement les enregistrements qui sont près des frontières des classes [13].

La méthode ne nécessite pas de phase d'apprentissage. Le modèle sera constitué de l'échantillon d'apprentissage, de la distance et de la méthode de combinaison des voisins.

Il faut choisir l'échantillon, c'est-à-dire les attributs pertinents pour la tâche de classification considérée et l'ensemble des enregistrements. Il faut veiller à disposer d'un nombre assez grand d'enregistrements par rapport au nombre d'attributs et à ce que chacune des classes soit bien représentée dans l'échantillon choisi. Le choix de la distance par champ et du mode de combinaison des distances se fait en fonction du type des champs et des connaissances préalables du problème. Il est possible de choisir la distance en faisant varier cette distance et, pour chacun des choix, estimer l'erreur réelle. On choisit alors la distance donnant la meilleure erreur réelle estimée [16].

2.1.5 Choisir k

Comment choisir la valeur de k ? En fait, il ne peut y avoir de meilleure solution évidente. Si on choisit une petite valeur pour k, la classification peut être affectée par les valeurs extrêmes ou les observations exceptionnelles (bruit). Avec k petit (par exemple, k = 1), l'algo-

[11] On peut pondérer les attributs (a priori ou par apprentissage).

rithme rendra simplement la valeur de la variable cible de l'observation la plus proche, un processus qui peut mener l'algorithme vers un surapprentissage, en ayant tendance à mémoriser l'ensemble des données d'apprentissage.

2.1.6 K plus proches voisins pour les textes

k-PPV est un algorithme de reconnaissance des formes qui a prouvé son efficacité face au traitement de données textuelles (Yang, 1997). La phase d'apprentissage consiste à stocker les exemples étiquetés. Le classement de nouveaux textes s'opère en calculant la distance Euclidienne entre la représentation vectorielle du document et celles des exemples du corpus ; les k éléments les plus proches sont sélectionnés et le document est assigné à la classe majoritaire (le poids de chaque exemple dans le vote étant éventuellement pondéré par sa distance) [32].

2.1.7 Les domaines d'application

La méthode peut s'appliquer dès qu'il est possible de définir une distance sur les champs. Or, il est possible de définir des distances sur des champs complexes tels que des informations géographiques, des textes, des images, et du son. C'est parfois un critère de choix de la méthode PPV car les autres méthodes traitent difficilement les données complexes. On peut noter, également, que *la méthode est robuste au bruit* [16].

2.1.8 Limites

Le nombre d'attributs : la méthode permet de traiter des problèmes avec un grand nombre d'attributs. Mais, plus le nombre d'attributs est important, plus le nombre d'exemples doit être grand. En effet, pour que la notion de proximité soit pertinente, il faut que les exemples couvrent bien l'espace et soient suffisamment proches les uns des autres. Si le nombre d'attributs pertinents est faible relativement au nombre total d'attributs, la méthode donnera de mauvais résultats car la proximité sur les attributs pertinents sera noyée par les distances sur les attributs non pertinents. Il est donc parfois utile de sélectionner d'abord les attributs pertinents.

Le temps de classification : si la méthode ne nécessite pas d'apprentissage, tous les calculs doivent être effectués lors de la classification. C'est la contrepartie à payer par rapport aux méthodes qui nécessite un apprentissage (éventuellement long) mais qui sont rapides en classification (le modèle créé, il suffit de l'appliquer à l'exemple à classifier). Certaines méthodes permettent de diminuer la taille de l'échantillon en ne conservant

que les exemples pertinents pour la méthode PPV, de toutes façons, il faut un nombre d'exemples suffisamment grand relativement au nombre d'attributs.

La distance et le nombre de voisins : les performances de la méthode dépendent du choix de la distance, du nombre de voisins et du mode de combinaison des réponses des voisins. En règle générale, les distances simples fonctionnent bien. Si les distances simples ne fonctionnent pour aucune valeur de *k*, il faut envisager le changement de distance, ou le changement de méthode [16].

Les variables prédictives non pertinentes : la méthode est sensible aux variables prédictives non pertinentes ; le choix des variables prédictives est donc important.

la fonction de similarité : la méthode est sensible au choix de la fonction de similarité de l'algorithme [82].

Le haut coût computationnel de classification, c'est à dire pour chaque élément, il faut calculer sa ressemblance à tous les éléments d'apprentissage [26].

2. 2. Les arbres de décision

2.2.1 Définition

La méthode des arbres de décision est une technique d'apprentissage supervisée dont le but est de calculer automatiquement les valeurs de la variable à prédire (endogène), à partir d'autres informations (variables exogènes ou prédictives) [83].

Un arbre de décision est un enchaînement hiérarchique de règles logiques de choix construites automatiquement à partir d'une base d'exemples, en vue de choisir au mieux une issue (par exemple, une classe) parmi N. Un exemple est constitué d'une liste d'attributs, dont la valeur détermine l'appartenance à une classe donnée. La construction de l'arbre de décision consiste à utiliser les attributs pour subdiviser progressivement l'ensemble d'exemples en sous ensembles de plus en plus fins [12].

Les arbres de décision produisent des modèles compréhensibles : elles suggèrent une explication empirique ou une justification qui sera ensuite présentée à un décideur ou un expert [82].

Les systèmes d'apprentissage inductifs s'appuient, pour la plupart, sur le système ID3. Le principe de base repose sur la fabrication d'un arbre de classification à partir d'un ensemble d'exemples expérimental. La technique ID3 calcule l'arbre de décision minimal en recherchant à chaque niveau, le paramètre (ou la formule) le plus discriminant pour classifier un exemple.

Il détermine pour cela la séquence d'attributs qui conduit le plus rapidement possible à une classification correcte. La visualisation de l'arbre de décision permet d'interpréter immédiatement l'ensemble des découpages successifs (Figure 2.4). On mesure la qualité du modèle généré par sa capacité à affecter les exemples aux bonnes classes [12].

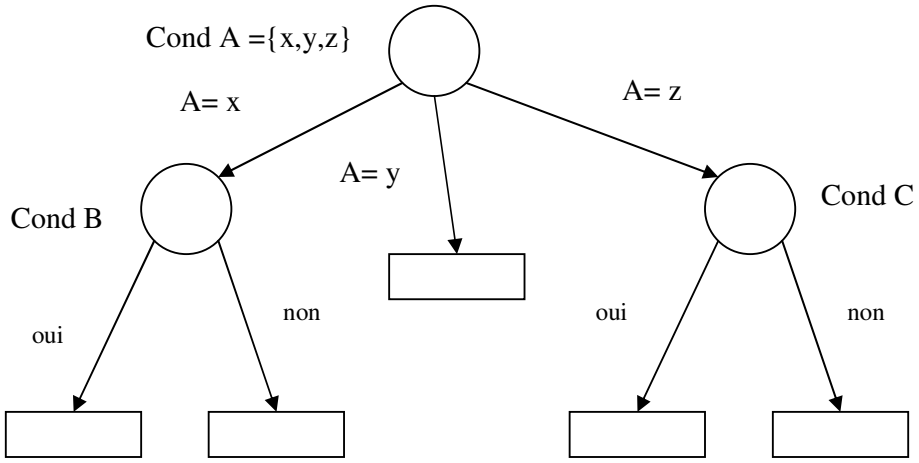


Figure 2.4. Arbre de décision.

2.2.2 Principe

Le principe des arbres de décision repose sur un partitionnement récursif des données. Le but du partitionnement est d'obtenir des groupes homogènes du point de vue de la variable à prédire. Le résultat est un enchaînement hiérarchique de règles. Un chemin, partant de la racine jusqu'à une feuille de l'arbre, constitue une règle d'affectation du type « *Si* <conditions> *Alors* <conclusion> ». L'ensemble de ces règles constitue le ***modèle de prédiction*** [83].

L'algorithme de détermination de la variable la plus significative est la base de la technique de construction des arbres de décision. L'algorithme cherche à diminuer le désordre apparent des données en s'appuyant sur une fonction d'évaluation. Il existe de nombreuses variantes de cet algorithme. Néanmoins, le principe commun consiste à choisir à chaque niveau, la variable qui permet d'extraire le maximum d'informations, ou *variable discriminante* [12]. Un bon arbre de décision permet de classifier le mieux possible en posant le minimum de questions (un minimum de profondeur)[12]

[12] Cf C.F. Picard, 1970, *Théorie des Questionnaires,* Gauthier-Villars.

2.2.3 Le descripteur qualitatif

Les algorithmes dérivés de la théorie de l'information

Dans le cas d'une variable qualitative, la mesure de l'incertitude emprunte un formalisme basé sur l'estimation de la probabilité d'appartenance de la variable à une classe. L'incertitude peut s'apprécier au moyen de l'entropie de Shannon (1948) définie comme :

$$-\sum P_i \log_2 (P_i)$$

où P_i représente la probabilité d'appartenance à la classe i. Cet indicateur est minimal lorsque la probabilité d'une classe est égale à 1. Si quatre classes sur quatre sont représentées de manière équiprobable, l'incertitude est maximale.

Cet indicateur est une bonne mesure de l'incertitude ou du désordre. La principale technique mise au point par J R.Quinlan [164] compare l'évolution de cet indicateur lors du test d'une variable pour détecter la valeur la plus discriminante

Pour chaque descripteur, on calcule le désordre qui reste après son utilisation. Celui qui laisse le moins de désordre est choisi comme étant le prochain nœud de l'arbre.

Les algorithmes issus du khi2

Une autre approche de création des arbres de décision est issue de l'algorithme CHAID : la définition de la variable la plus significative est basé sur le test de khi2, ce dernier permet de vérifier la conformité d'un phénomène aléatoire à une loi de probabilité posée comme hypothèse, il s'appuie sur la comparaison entre les fréquences observées pour chacune des classes et les fréquences théoriques.

2.2.4 Le descripteur quantitatif

L'objectif est identique, toutefois, la méthode change car le nombre de valeurs du concept peut être infini.

La méthode des grappes

Elle consiste à partitionner le domaine de la variable continue en intervalles ordonnés. Ce découpage est construit à partir des indicateurs traditionnels tels que la moyenne, la médiane ou les quantiles[13]. Le découpage par quantiles permet de définir les limites de chacune des classes, le nombre de classes étant égal pour l'ensemble des variables quantitatives. La

[13] Quartiles, déciles, centiles… pour 4, 10, 100 tranches.

perte d'information est identique pour toutes les variables. La fonction permet de sélectionner la variable la plus discriminante

La méthode des grappes ne garantit pas un seuil optimal de découpage de la variable. Néanmoins cette méthode requiert un temps de calcul court et s'approche de la bonne valeur.

La méthode exhaustive

Cette méthode détermine le seuil optimal de découpage de la variable. Ce seuil est choisi de sorte que les partitions de la variable explicative permettent de discriminer au mieux l'attribut. On évalue tous les seuils possibles pour retenir le meilleur :

- toutes les valeurs que l'attribut est susceptible de prendre, sont parcourues dans l'ordre croissant,
- pour chaque valeur on réalise une partition de l'attribut et on calcule le pouvoir discriminant de la variable,
- lorsque le domaine des valeurs a été entièrement parcouru, on retient comme seuil pour les partitions binaires celui auquel correspond le meilleur pouvoir discriminant

La technique exhaustive est très coûteuse en temps de calcul si les attributs numériques sont nombreux, mais si l'éventail des valeurs possibles pour chaque variable numérique est large elle assure en revanche un meilleur découpage de l'attribut [12].

2.2.5 Les domaines d'application

Les études, pour comprendre les critères prépondérants dans l'achat d'un produit, l'impact des dépenses publicitaires,

Les ventes, pour analyser les performances par région, par enseigne, par vendeur,

L'analyse de risques, pour détecter les facteurs prédictifs d'un comportement de non-paiement,

Le domaine médical, pour étudier les rapports existant entre certaines maladies et des particularités physiologiques ou sociologiques [12].

2.2.6 Les limites

Les arbres de décision présentent ces limites :

- L'arbre détecte des optimums locaux et non globaux. Il n'utilise pas simultanément, mais séquentiellement, toutes les variables explicatives.

- Le choix d'une division pour un nœud à un certain niveau de l'arbre n'est jamais remis en cause.
- En conséquence, la modification d'une seule variable, si elle est placée prés du sommet de l'arbre, peut entièrement modifier l'arbre.
- La distribution des variables discriminantes doit être la même dans l'échantillon d'apprentissage et l'échantillon de test car la classification par arbres de décision est sensible aux légères fluctuations aléatoires des données les plus discriminantes [06].
- Sensible au nombre de classes : les performances tendent à se dégrader lorsque le nombre de classes devient trop important [82].
- Incapacité, avec les algorithmes classiques (C4.5, CART, CHAID, etc.), à détecter les combinaisons de variables[14] ; ceci est dû au principe de construction pas à pas de l'arbre, entraînant une certaine ' myopie ' [33].
- La nécessité de disposer d'un échantillon d'apprentissage de grande taille (ou très représentatif). Certes, l'arbre peut reproduire approximativement toutes formes de frontières, mais au prix d'une fragmentation rapide des données, avec le danger de produire des feuilles avec très peu d'individus [33]. (L'apprentissage d'un arbre de décision nécessite un nombre suffisamment grand d'individus pour avoir au moins 20 à 30 individus par nœud, même dans la partie la plus basse de l'arbre [06]). Corollaire à cela, les arbres sont en général instables ; les bornes de discrétisation notamment dans les parties basses de l'arbre sont entachées d'une forte variabilité. Ainsi, certains chercheurs préconisent de procéder à la discrétisation préalable des variables avant la construction de l'arbre [34].
- Évolutivité dans le temps : l'algorithme n'est pas incrémental, c'est-à-dire, que si les données évoluent avec le temps, il est nécessaire de relancer une phase d'apprentissage sur l'échantillon complet (anciens et nouveaux exemples) [06].

Remarques :

1. Si chaque élément est décrit par un vecteur binaire, on peut trouver des facteurs discriminants à l'aide de techniques booléennes : « ou exclusif » bit à bit, techniques de couverture, distance de Hamming (nb de bits à 1 du $\oplus$ bit à bit, utilisée notamment par des codes détecteurs ou correcteurs d'erreur). Soit à comparer :

[14] gros problème en génétique.

100100100

⊕ 010101010

110001110 -> $d_H = 5$ (discrimination par l'une des variables 1,2,6,7,8)

2. L'analyse discriminante sera beaucoup plus intéressante, car elle dégage des *facteurs discriminants* plutôt que des variables discriminantes.

Ainsi**,** x est-il riche ? pauvre ? à l'aise ? Soit R son revenu mensuel et N le nombre de personnes à sa charge. Un bon facteur discriminant serait de style

$$x.R > \alpha \cdot x.N + \beta$$

La recherche de facteurs discriminants devrait minimiser le nombre de décisions.

2.3. Les réseaux de neurones

2.3.1 Définition

Les *réseaux de neurones* constituent un modèle de traitement informatique qui imite le fonctionnement de base du cerveau humain. Le cerveau est modélisé comme un vaste réseau de neurones formels structuré en un système doté de multiples interconnexions [12].

En exploitation, il est considéré comme une boîte noire, pouvant par exemple assigner une classe à chaque cas soumis.

Le plus ancien exemple connu est le *Perceptron* de Rosenblatt (1957) ; il s'agissait d'une rétine artificielle, devant reconnaître des chiffres ou des lettres majuscules définis à l'aide d'une petite matrice de points lumineux. Après une certaine désaffection, ce genre de dispositif, critiqué par Minsky[15] et Papert[16], repris puis dûment étendu et théorisé dans les années 1980, a fait un important retour dans le domaine de la reconnaissance de formes, au moyen des réseaux neuronaux multicouches.

---

[15] Inventeur des frames, des scripts, de l'analyse du cerveau comme un Système Multi-Agents

[16] Créateur du langage Logo, pour une approche conviviale et créative de l'IA.

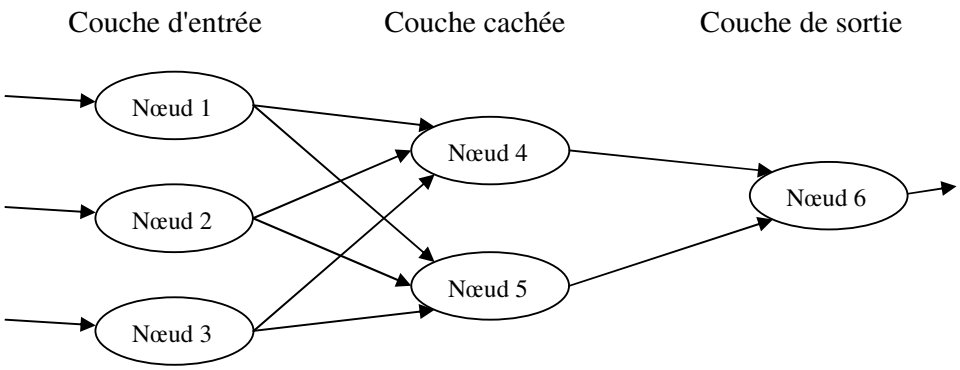


Figure 2.5. Un réseau de neurones

2.3.2 Le neurone formel

Un neurone formel est un composant combinatoire élémentaire qui réalise la transformation par une fonction d'activation φ d'une somme pondérée par des poids $W_1,W_2,,…..W_n$ des entrées (synapses) $X_1,X_2,….X_n$ qu'il reçoit.

Le modèle du neurone est dit binaire si les entrées et les sorties sont binaires.

Le modèle est dit analogique si les entrées et la sortie sont à valeur dans [0 1]. La fonction φ est alors une sigmoïde ou une fonction similaire paramétrée par un seuil θ.

Le Réseau de Neurones est caractérisé par une règle ou fonction d'activation, une organisation en couches et une règle d'apprentissage

2.3.3 L'organisation en couches

Le réseau se compose d'un ensemble de nœuds connectés entre eux par des liens orientés, ou connexions. Bien que n'importe quelle structure soit imaginable, la forme la plus utilisée est l'organisation en couches successives, les sorties des neurones d'une couche devenant les entrées de la couche suivante.

Une telle structure diffuse l'information de la couche d'entrée vers la couche de sortie à travers une ou plusieurs couches cachées. La couche d'entrée reçoit les données du problème. La couche de sortie affiche le résultat. Selon les problèmes, on intercale un nombre variable de couches intermédiaires, dites couches cachées (dépendant aussi de la fonction d'activation).

Le réseau est un système non linéaire qui associe aux états de la couche d'entrée des états de la couche de sortie. Chaque configuration de poids d'un réseau d'une architecture déterminée définit une fonction spécifique

2.3.4 L'auto-apprentissage

La *règle d'apprentissage* régit la capacité du réseau à changer son comportement d'après les résultats de son expérience passée, en dictant la façon dont les poids des connexions (les entrées) varient en fonction du temps. Une des particularités des réseaux de neurones est cette capacité à s'auto-adapter[17] sans qu'aucun agent extérieur (programmeur) n'intervienne dans ce processus d'amélioration. Pour un cas donné, la règle d'apprentissage a pour but de minimiser l'erreur entre la sortie prévue par le réseau et la valeur à produire selon le fichier d'apprentissage [12].

2.3.5 Les domaines d'application

Les réseaux de neurones sont largement utilisés dans de nombreux secteurs d'activité :

La reconnaissance de formes

Elle consiste à affecter un signal d'entrée à une classe prédéfinie. La qualité de classification dépend de la durée et de l'exhaustivité de la phase d'apprentissage, qui consiste à présenter au réseau de neurones des formes connues et à lui indiquer ses erreurs et ses succès.

C'est une forme de traitement du signal. Lorsqu'un système informatique doit traiter des (segments de) signaux analogiques d'origine physique, ceux-ci sont d'abord numérisés par échantillonnage, mais il est souvent souhaitable de remplacer ces longues suites de valeurs, d'une variabilité pas toujours significative, par des codes concis convenant aux traitements informatiques ultérieurs.

Au titre de l'instrumentation et de la condensation des signaux, on trouve des applications du traitement du signal en reconnaissance de la parole, dans le domaine médical, dans le domaine militaire, en gestion de procédés....

La classification

Les applications couvrent notamment le domaine du marketing (pour l'identification de segments de clients) ou le domaine industriel (pour la détection des défauts ou des pannes)

[17] Auto-adaptation, plutôt qu'auto-organisation car il n'y a pas de changement de structure.

La prévision

Les réseaux de neurones de prévision sont souvent mis en œuvre pour la prévision des cours des valeurs boursières[18], les prévisions météorologiques, les modèles de prévision des ventes en marketing, en marketing direct ou dans la modélisation de processus industriels complexes.

Le contrôle adaptatif

Les applications principales se trouvent dans le domaine de la robotique, avec le déplacement des robots sur des parcours par simple ajustement de la vitesse de rotation des roues [12].

De façon générale les applications des réseaux de neurones se trouvent dans les secteurs :

- *Aérospatial* : pilotage automatique, simulation du vol…
- *Automobile* : système de guidage automatique,…
- *Défense* : guidage de missile, suivi de cible, reconnaissance du visage, radar, sonar, traitement du signal, compression de données, suppression du bruit…
- *Electronique* : prédiction de la séquence d'un code, vision machine, synthétiseur vocal, modèle non linéaire,…
- *Finance* : Prévision du coût de la vie.
- *Secteur médical* : Analyse EEC et ECG.
- *Télécommunications et compression de données* [35].

2.3.6 Les limites

Les réseaux de neurone se présentent comme des boites noires, et il est bon, après avoir obtenu une segmentation, de l'analyser pour découvrir la composition des segments.

Les segments sont moins différenciés en taille et en contenu.

Un bon apprentissage du réseau nécessite un échantillon important pour un bon calcul de tous les poids des nœuds, l'échantillon devant comprendre le plus grand nombre possible de modalité des variables.

Les variables doivent être numériques et leurs modalités comprises dans l'intervalle [0,1], ce qui peut impliquer une normalisation des données

[18] Les résultats n'excèdent pas la compétence du financier qui a conduit l'apprentissage

La segmentation neuronale est sensible aux distances extrêmes entre individus, aux individus isolés. Deux individus A et B semblables en tout point sauf un, peuvent ainsi se trouver séparés, car B a une valeur aberrante pour cet attribut, et B peut se trouver rapproché d'un individu C par ailleurs dissemblable, qui possède la même valeur aberrante que B [06].

Plus généralement, les réseaux neuronaux utilisés ont une capacité cognitive faible : 6 plans de neurones donnent les capacités intellectuelles d'une grenouille, autrement dit des réflexes plus ou moins complexes.

2.4. Les Réseaux Bayésiens

2.4.1 Définition

Les Réseaux Bayésiens sont une méthode classique utilisée pour estimer la probabilité d'apparition d'un événement étant donné la connaissance de certains autres événements.

Un Réseau Bayésien est un modèle graphique qui encode les probabilités liant les variables les plus pertinentes. La visualisation des dépendances entre variables permet d'identifier certaines relations causales existant entre les variables, ou la conjonction de certaines variables pour déclencher une action. [12]

Graphiquement, un réseau bayésien est un graphe acyclique dont les nœuds représentent des variables d'état, reliés par des flèches marquant des relations probabilistes entre variables (Figure 2.6).

Les probabilités sont documentées à raison d'une table par nœud.

(a) si un noeud ne possède pas de flèche entrante, alors la table contiendra la probabilité de son activation.

(b) si un nœud a des prédécesseurs, sa table contient les probabilités conditionnelles liant son état à l'état des prédécesseurs immédiats pour les différentes combinaisons possibles [178].

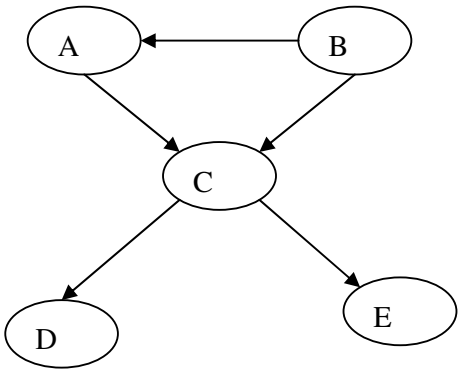


Figure 2.6. Un réseau bayésien.

2.4.2 Le principe

Un réseau bayésien est un graphe orienté dans lequel les nœuds représentent des variables et les arcs symbolisent des dépendances entre les variables. L'état d'une variable est caractérisé par sa probabilité. La dépendance entre variables est caractérisée par les probabilités conditionnelles.

La probabilité d'apparition d'un événement A portant sur l'objet x, la durée *y* et le montant *z*, notée p(A/x,y,z) s'exprime en fonction de p(x) par :

$$p(x) \cdot p(z/x) \cdot p(y/x,z) \cdot p(A/z,y)$$

Connaître l'ensemble des probabilités reliant les événements est souvent difficile. Les principales possibilités sont

(a) les relevés systématiques

(b) l'évaluation subjective par un expert ; mais cette estimation des probabilités peut être biaisée par des facteurs tels que la mémorisation, le jugement ou les circonstances particulières d'apparition d'un événement : les probabilités associées à des événements lointains sont sous-estimées (oublis), alors que celles associées aux événements récents ou remarqués sont surestimées.

Un réseau bayésien fournit la probabilité d'apparition d'un événement connaissant le résultat observé sur d'autres variables, par propagation dans le réseau des valeurs relatives au cas : la complexité du calcul (inférence bayésienne) dépend de la structure du réseau construit comme de la disposition dans ce réseau des valeurs connues.

Le but de la mise en place de ces réseaux bayésiens est d'avoir d'une part une représentation graphique du problème et, d'autre part, un outil qui permette une prise de décision en fonction de n'importe quelle situation possible découlant du scénario représenté. Cette

prise de décision se fait en fonction des valeurs de probabilité entrées dans les tableaux de chaque variable (n'excluant pas 1 ou 0 dans les cas certains). Ces probabilités ne reposent pas toujours sur des valeurs réelles disponibles, mais doivent souvent être estimées par une méthode ou par une autre. Ainsi, il faut considérer la décision – basée sur la probabilité d'une proposition par rapport à d'autres, prise par le réseau – comme une décision issue d'un contexte très précis qui ne correspond pas totalement à la réalité.

2.4.3 La complexité du réseau

Un premier élément de complexité est lié aux variables. Les variables discrètes sont représentées par autant de nœuds que de valeurs, alors que les variables continues sont discrétisées par grappes.

Le deuxième facteur de complexité concerne la croissance des connexions.

De toute évidence, plus le réseau est complexe, plus le temps de calcul est long (tendance exponentielle). L'ampleur du réseau dépend du nombre de variables, du nombre d'occurrences par variables et du nombre de liens parents autorisés. Limiter le nombre de variables, de valeurs par variables et de liens est un moyen de lutter contre cette complexité.

L'élagage du réseau

La limitation des valeurs peut passer par un regroupement au sein d'une même variable, soit sous le contrôle de l'expert, soit par un algorithme de regroupement semblable à la technique des arbres de décision.

La limitation des liens

La limitation des liens se construit en fixant l'entrance maximale par nœud. Cette solution réduit le temps de calcul, mais peut conduire à la perte de dépendances entre variables. En autorisant plus de parents, il est plus facile de représenter les dépendances connues. A l'inverse, un réseau trop pauvre se révèle inapte à la représentation du problème. La recherche d'un réseau optimal est donc un équilibre entre une couverture minimale pour assurer une représentation correcte du problème et une complexité limitée pour maintenir des temps de calcul raisonnables [12].

2.4.4 Domaines d'application

Les applications opérationnelles des réseaux bayésien sont un peu moins nombreuses que celles des autres techniques exposées ici. Les premières applications dont la littérature a

fait état sont la modélisation des processus d'alertes dans le domaine industriel et la prédiction du risque d'impayés dans le domaine des télécommunications [12]. L'application de ces réseaux bayesiens est réalisée dans le cadre de plusieurs types de trace comme l'interprétation des traces d'accélérant dans les incendies, du verre, des fibres, des traces biologiques, etc.

Les réseaux bayésiens apportent une solution au traitement de nombreuses données.

Par exemple, sur certains sites Internet, au besoin de personnaliser la relation pour fidéliser le visiteur et de susciter des clics pour augmenter les ressources publicitaires, oblige les responsables de tels sites à rechercher la meilleure adaptation possible du message à la cible. La capacité d'identifier toutes les relations entre les variables des réseaux bayésiens autorise la construction automatique d'un modèle qui maximise le taux de clics des données recueillies.

2.4.5 Les limites

La recherche du meilleur réseau est une tâche très consommatrice de puissance informatique, car le nombre de combinaisons variables/arcs possibles est de nature combinatoire. Les algorithmes existants déterminent un réseau probable. Néanmoins, comme les algorithmes génétiques, ils ne garantissent pas qu'il s'agit du réseau optimal. Ils recherchent la solution optimale en démarrant d'un réseau simple, ils évaluent les réseaux dérivés de chaque modification résultant de l'ajout d'un nœud ou d'une dépendance. Pour effectuer cette tâche de recherche, ils doivent lire les données $n\cdot(u+1)$ fois, *n* étant le nombre de variables et *u* le nombre de liens parents, où $u=O(n^2)$, soit une complexité globale $O(n^3)$.

La recherche du modèle est très consommatrice en puissance de calcul. Ceci conduit à réduire la formalisation du problème, en collaboration avec des experts du domaine. Le recours aux experts est en effet souvent nécessaire pour réduire la complexité initiale, ordonner les variables ou identifier les dépendances les plus importantes [12]. La puissance informatique requise par la construction d'un réseau bayésien est encore relativement incompatible avec les bases de plusieurs gigaoctets, mais l'augmentation de la puissance des processeurs devrait progressivement gommer ce handicap[19].

[19] Si la complexité est polynomiale en n. Au-delà, l'accélération des processeurs a trop peu d'impact.

2.5. La Segmentation Hiérarchique Ascendante

2.5.1 Définition

Contrairement aux méthodes précédentes, qui sont non-hiérarchiques, c'est à dire produisent directement une segmentation en un certain nombre de segments disjoints, la segmentation hiérarchique ascendante produit des suites de partitions emboîtées, de la partition discrète où chaque individu est isolé, jusqu'à la partition universelle qui regroupe tous les individus.

La suite des partitions est représentée par un arbre, aussi appelé dendrogramme (Figure 2.7) :

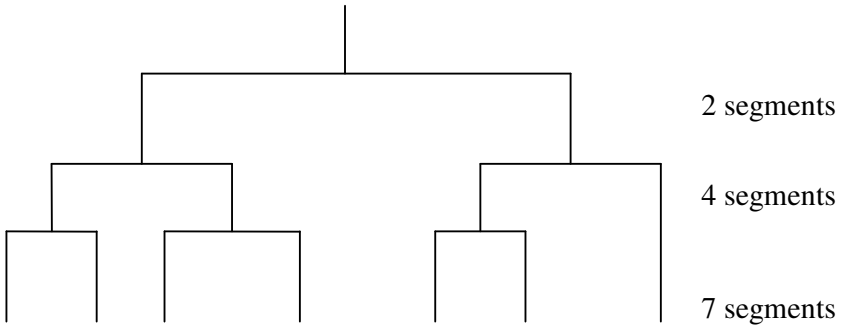


Figure 2.7. Segmentation hiérarchique ascendante.

2.5.2 Principe

Le dendrogramme peut être coupé à une hauteur plus ou moins grande pour obtenir un nombre plus ou moins restreint de classes, ce nombre de classes pouvant être choisi par le statisticien pour optimiser des critères basés par exemple sur l'*inertie interclasse.*

L'algorithme de la segmentation hiérarchique ascendante fonctionne donc en recherchant à chaque étape les deux classes les plus proches pour les fusionner, et le sel de l'algorithme réside dans la définition de la distance de deux classes. On peut la définir comme étant la distance de leurs points les plus proches, les plus éloignés, de leurs barycentres, etc., mais la notion correspondant le mieux à l'objectif de la segmentation est celle d'*inertie interclasse.*

Comme une bonne segmentation est une segmentation pour laquelle l'inertie interclasse est élevée, et comme le passage d'une segmentation en k+1 segments à une segmentation en k segments (regroupement de deux segments) ne peut que faire baisser l'inertie interclasse, les deux segments à fusionner seront ceux qui feront le moins baisser l'inertie interclasse. Autrement dit, la distance de deux segments est la perte d'inertie interclasse résultant de leur fusion.

En même temps que le critère de fusion, cette méthode de segmentation (méthode de Ward) fournit un critère de hauteur de coupe de l'arbre : elle doit correspondre à un minimum local de la perte d'inertie interclasse [06]. La méthode de Ward, *aisée à mettre en œuvre lorsque la classification est effectuée après une analyse factorielle* (les objets à classer étant repérés par leurs coordonnées sur les premiers axes factoriels), constitue une excellente méthode de classification ascendante hiérarchique.

La segmentation hiérarchique ne présente pas les deux inconvénients majeurs de la méthode des centres mobiles : fixation à priori du nombre de segments et dépendance du choix des centres initiaux [06].

Domaines d'application

- Classification, apprentissage non supervisé [39].
- Les méthodes de classification ascendante hiérarchique (CAH) sont utilisées à la fois dans le domaine de l'analyse de données et celui de l'apprentissage [38].

Limites

1. La complexité de l'algorithme est polynômiale, car, pour passer de k+1 segments à k segments, il faut calculer ((k+1)·k)/2 distances, puis réunir les deux segments les plus proches, recalculer les distances, avant de recommencer. Si n est le nombre d'individus à segmenter, la complexité de l'algorithme est en $O(n^3)$, complexité algorithmique qui reste par ailleurs raisonnable [40]. Par contre, la construction d'une hiérarchie strictement binaire conduit à une segmentation des données qui est assez arbitraire [38]

2 Les segments construits ont tendance à être de même taille (même problème que la segmentation neuronale).

3 A chaque étape, le critère de partitionnement n'est pas global, mais dépend des segments déjà obtenus : deux individus placés dans des segments différents ne sont plus jamais comparés. En d'autres termes, une telle segmentation en *n* segments n'est jamais la meilleure possible, mais seulement la meilleure parmi celles obtenues en réunissant deux segments d'une segmentation en n+1 segments. Certains segments naturels peuvent ainsi être occultés par une bifurcation antérieure [06].

Du point de vue de l'explicabilité, les méthodes ascendantes sont intéressantes car la construction de chaque classe résulte du regroupement d'un petit nombre d'individus similaires [38].

2.6. Méthode des Centres Mobiles (K means)

2.6.1 Définition

La méthode des centres mobiles est une méthode de *partition* d'un groupe d'objets, et non une méthode de classification hiérarchique.

K-means (MacQueen, 1967) est un des algorithmes supervisés les plus simples qui résolvent les problèmes de segmentation [17].

La procédure suit une façon simple et facile de classer un ensemble donné de points[20] ou en K groupes.

L'idée principale est de choisir K centroïdes (ou centres présumés), un pour chaque groupe. Ensuite, on prend chaque point restant à classer, et on l'associe au centroïde le plus proche. Quand aucun point ne reste, un premier groupage est fait.

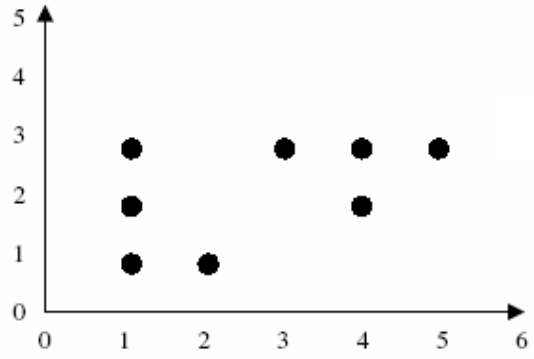
Pour s'affranchir de l'arbitraire du premier choix, on calcule les barycentres des groupes résultant de l'étape précédente, pour être les k centroïdes d'une nouvelle passe, et si on répète le processus d'agglutination, à la longue les k centroïdes se stabilisent (avec le groupage résultant), le processus minimisant une fonction objective [13].

2.6.2 Principe

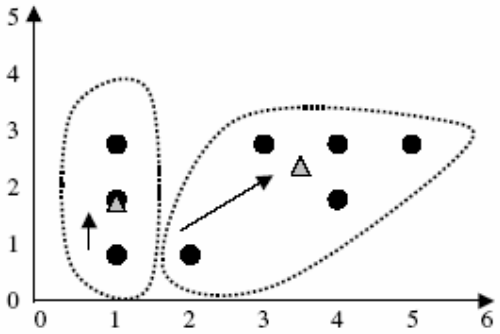
La méthode des centres mobiles se déroule comme suit :

1. On choisit K individus c_1, c_2, c_k (on tire au sort, ou l'on prend les k premiers, ou l'on prend 1 sur n/k,..) ;
2. On regroupe les autres individus autour des centres choisis en 1, de tel sorte que le groupe de c_i soit constitué des individus plus proches de c_i que de tout autre centre ;
3. On remplace les individus de (1) par les barycentres des groupes définis en (2) (barycentres qui ne sont pas nécessairement des individus de la population);
4. comme en 2
5. dès que l'on a 2 partitions successives, on reprend en 2 sauf si l'inertie interclasse ne décroît plus sensiblement d'une partition à la suivante [06].

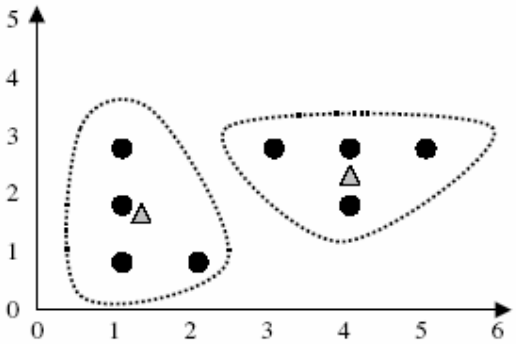
[20] Rappel : tout objet est présumé point dans l'espace des critères ou attributs.



Étape 1, K=2



Étape 2, deux segments avec deux barycentres après la première passe



Étape 3, les segments et les barycentres après la deuxième passe

Figure 2.8. Exemple sur l'utilisation de l'algorithme de k-means.

Le groupage obtenu à stabilité est présumé justifié par la stabilité atteinte.

De fait, les différents emplacements des centroïdes initiaux causeront des résultats différents. Le meilleur choix est de les placer loin l'un de l'autre autant que possible.

2.6.3 Variantes

Dans la variante des K-means de Mac Queen, le barycentre de chaque groupe est re-calculé à chaque nouvel individu introduit dans le groupe, au lieu d'attendre l'affectation de tous les individus et l'étape 3 avant de calculer les barycentres.

La méthode des ***nuées dynamiques*** (Diday / INRIA) se distingue principalement de celle des centres mobiles en ceci : chaque classe n'est plus représentée par son barycentre (éventuellement extérieur à la population), mais par un sous-ensemble de la classe, appelé *noyau*, qui, s'il est bien composé (des individus les plus typiques, ou prototypes), sera plus représentatif de la classe que son barycentre.

L'avantage de cet algorithme est que sa complexité est linéaire, c'est-à-dire que son temps d'exécution est proportionnel au nombre n d'individus (on calcule à chaque étape les $n \cdot k$ distances entre les individus), ce qui le rend applicable à de grands volumes de données. Cela est d'autant plus vrai que le nombre d'itérations nécessaires pour minimiser l'inertie intraclasse est généralement faible [06].

2.6.4 Domaines d'application

La méthode peut s'appliquer dans les problèmes de segmentation et classification (Segmentation d'images, catégorisation des pages HTML dans les moteurs de recherche).

2.6.5 Limites

1. Cette méthode ne s'applique qu'à des données continues, ce qui nécessite, si ce n'est pas le cas des données étudiées, de les transformer en données continues par une analyse des correspondances multiples en espérant avoir une notion correcte de distance entre les individus.
2. La partition finale dépend beaucoup du choix initial (arbitraire) des centroîdes c_i. On n'a donc pas un optimum global, mais seulement la meilleure partition possible à partir de celle de départ. (mais on peut recommencer avec d'autres centres et comparer -- ou carrément modifier l'algorithme avec des algorithmes génétiques).
3. Le nombre K de segments est fixé dans cette méthode, et n'est inférieur à K que si certains segments sont vides. Si ce nombre ne correspond pas à la configuration véritable du nuage des individus, la qualité de la segmentation risque d'être douteuse. On peut essayer de pallier cet inconvénient en testant diverses valeurs de K, mais cela augmente la durée du traitement [06].

2.7. Associations

2.7.1 Définition

La *recherche d'association* vise à construire un modèle de décision fondé sur des règles conditionnelles à partir d'un fichier de données.

Une règle conditionnelle se définit sous la forme d'une suite

si **<***condition***>** ***alors*** **<***résultat***>,**

où la condition peut utiliser des opérateurs logiques, comme dans :

si A et B et non D alors C.

La recherche des associations peut s'appliquer à l'ensemble des données.

2.7.2 Les enjeux

Les applications de la recherche d'associations sont multiples. Par exemple, elles vont d'une meilleure connaissance du client, et donc de son panier, jusqu'à l'optimisation des stocks ou du merchandising.

La découverte d'une séquence logique des transactions permet l'optimisation des procédures d'approvisionnement d'un magasin.

Optimisation des stocks

La découverte d'associations entre des produits peut entraîner une réorganisation des commandes ou de la surface de vente ; l'observation d'associations entre des articles alimentaires, des vêtements, de la parapharmacie et des meubles pour les tout-petits peut conduire à définir un *espace puériculture* dans un catalogue.

Ventes croisées

La découverte d'association permet la réalisation de campagnes promotionnelles personnalisées avec l'édition de bons de réduction en fonction des achats. Cette forme de marketing d'intimité est essentielle pour faciliter les achats du client et optimiser la politique de réapprovisionnement du magasin, mais l'analyse d'associations apparaît avant tout comme un moyen de fidélisation, ce qui contribue à augmenter le chiffre d'affaires. Il faut utiliser la *connaissance client* pour faire revenir dans la même enseigne un client qui visite régulièrement plus de trois hypermarchés pour faire ses achats sans les différencier [12].

2.7.3 Principe

Prenons l'exemple des tickets de caisse émis par un supermarché. La base d'analyse se compose de l'ensemble des transactions réalisées sur une période donnée.

Une transaction T est représentée par un ticket de caisse, qui comprend un ensemble d'articles. Une association est une implication de la forme $x \rightarrow y$, où x et y sont 2 éléments distincts de la transaction T.

Le niveau de confiance et de support

Une association s'apprécie au travers de deux indicateurs

- le niveau de confiance ;
- le niveau de support.

Par exemple, un certain supermarché a eu 1000 clients un jeudi de nuit, 200 ont acheté le produit A, dont 50 ont acheté le produit B, Donc, la règle d'association serait " *Si achète A, alors achète B* " avec un *support* de 200/1000 = 20% et une *confiance* de 50/200 = 25%.

L'extraction des associations pertinentes

L'objectif est de détecter les associations qui présentent un niveau de confiance et un niveau de support élevé. Le processus d'extraction des associations se déroule en deux phases distinctes : il isole les articles présentant un niveau de support supérieur à un certain seuil, puis il combine les articles les plus représentés pour générer les associations. Cette phase de sélection des articles qui présentent un taux de support correct est primordiale. Elle permet d'améliorer les temps de réponse en restreignant la taille de la base.

Tout d'abord, on dénombre le nombre d'occurrences d'un article dans l'ensemble des transactions. On décide de retenir un taux de support supérieur à un seuil donné, les articles qui ont un taux de support inférieur sont éliminés.

La deuxième étape combine les articles restants pour former l'ensemble de toutes les associations et dénombrer le nombre d'occurrences de chacune. De la même manière, on élimine les associations qui présentent un taux de support inférieur à un seuil.

La troisième étape consiste à créer les triplets possibles, puis les quadruplets et ainsi de suite [12].

2.7.4 Domaines d'application

Les domaines d'applications sont nombreux et les utilisations les plus fréquentes touchent :

- l'analyse des achats dans la grande distribution pour agencer les rayons,
- la gestion de stocks pour éviter ruptures et sur stockage,
- l'analyse des mouvements dans la banque,
- l'analyse des incidents dans l'assurance ou l'analyse des communications dans les télécommunications [13].

Plus généralement, l'analyse des associations s'applique avec succès à tous les problèmes dans lesquels l'apparition d'un événement est conditionnée par des événements passées : analyse des pannes dans l'industrie ou étude des décisions en sociologie.

2.7.5 Limites

La méthode est coûteuse en temps de calcul. Le regroupement d'articles et la méthode du support minimum permettent de réduire les calculs au risque d'éliminer malencontreusement des règles importantes.

Il est difficile de déterminer le bon niveau d'articles. Les traitements préalables sur les achats peuvent être complexes. La méthode est plus efficace pour les articles fréquents que pour les articles rares. Pour les articles rares, on peut restreindre la forme des règles choisies ou faire varier le support minimum.

La méthode peut produire des règles triviales ou inutiles. Les règles triviales sont des règles évidentes qui, par conséquent, n'apportent pas d'information. Les règles inutiles sont des règles difficiles à interpréter qui peuvent provenir de particularités propres à la liste des achats ayant servi à l'apprentissage. Les règles inintéressantes submergent les règles pertinentes[21]. L'utilisation de taxonomie permet de réduire cet handicap.

2.8. Les Systèmes Experts

2.8.1 Définition

Les systèmes experts sont des logiciels simulant une expertise, et donc très utiles en matière de diagnostic, de conseil, d'aide à la décision…[06].

[21] S'il y a transitivité des choix.

Les systèmes experts sont apparus dans les années 70 et ont connu une forte notoriété dans les années 80.

2.8.2 Principe

Ils comportent une base de connaissances (spécifique à l'application) contenant des faits et un ensemble de règles logiques du style

si <*condition*> *alors* <*conclusion*>.

Ils sont animés par un moteur d'inférence (générique), chargé d'appliquer ces connaissances au cas, et qui simule un raisonnement en appliquant aux faits les règles activables, ce qui engendre de nouveaux faits…, jusqu'à stabilisation sur un ensemble de conclusions.

Les faits initiaux et les règles sont obtenus à partir de la connaissance d'experts soigneusement choisis et interviewés.

Par opposition aux applications antérieurement programmées, le moteur d'inférence dispense de concevoir un algorithme à partir des règles recueillies. La base de connaissances, plus proche qu'un algorithme de la pensée des experts, est de ce fait plus facile à maintenir [06].

2.8.3 Pour quoi les Systèmes Experts ?

Les Systèmes à Base de Connaissance, ou systèmes experts, ne font pas à proprement parler, partie des techniques de fouille de données. Cela étant, ils méritent une mention à part dans le contexte de la fouille de données, pour différentes raisons :

- les acteurs de fouille de données, tant sur le plan de la recherche que sur le plan commercial, viennent souvent du monde de l'intelligence artificielle et des systèmes experts ;
- la percée en fouille de données est principalement le résultat de la recherche de solutions au goulet d'étranglement que constitue la phase de recueil d'expertise dans la constitution d'un système expert ; en ce sens, les SE sont *demandeurs* de fouilles de données vis-à-vis des gisements potentiels de connaissances que constituent les bases de données, les fonds documentaires….
- si les systèmes experts ne découvrent pas de connaissances dans les données, ils sont en revanche parfaitement indiqués pour appliquer de la connaissance sur ces mêmes données [12].

2.8.4 Les domaines d'application

Des outils tels que les arbres de décision ou les associations permettent d'obtenir rapidement des règles de décision, facilement exploitables par les moteurs d'inférence. Il est donc naturel de coupler fouille de données et système expert.

La répartition des tâches est simple : la fouille de données extrait des règles qui sont ensuite implémentées sous formes de base de connaissance dans un moteur d'inférence afin d'être appliquées à de nouvelles données. Alors, pourquoi pas un retour des systèmes experts grâce à la fouille de données ? [12].

Les systèmes experts sont utilisés pour le crédit immobilier, quand les historiques de données ne sont pas assez profonds pour calculer un modèle de score. De plus, dans ce domaine, l'aide à la décision ne porte pas seulement sur la notation du risque, mais également sur les dispositifs légaux et réglementaires qui peuvent entourer le dossier ; si ces dispositifs changent, il suffit de mettre à jour ponctuellement la base de connaissances [06].

Les outils Internet ont bien compris les possibilités qu'offrent les systèmes experts à base de règles couplées aux technologies de fouille de données, en matière de personnalisation de la relation avec le client. L'introduction de règles est simple et permet à un professionnel d'imposer ses critères. Les règles introduites permettent d'adapter le comportent du site au profil et aux actions de l'internaute.

2.8.5 Limites

Après une période d'engouement, la mode des systèmes experts est progressivement retombée[22]. Les causes sont multiples, mais il est possible d'en dégager plusieurs :

- La pérennisation des a priori des experts.
- Le goulet d'extraction du recueil de la connaissance : une base de connaissances se constitue à partir d'interviews d'experts. Cette tâche de recueil est complexe et longue. Elle alourdit les temps de réalisation et le coût des applications.
- Une synthèse des avis d'experts qui n'est pas toujours optimale[23]

---

[22] Les SE sont souvent devenus des fonctions expertes ou des modules experts enfouis dans des applications plus vastes.

[23] On évite d'avoir plusieurs experts sur une même question ; les systèmes multi-experts correspondent à des activités composites comme le DIAGNOSTIC D'ENTREPRISE, avec un expert pour chacune des fonctions majeures, ce qui pose de gros problèmes (notamment de terminologie univoque).

- La prise en compte des seules données déclaratives, et non des données comportementales enregistrées dans le système d'information.
- Le *conflit in vivo/in vitro* : l'expert peut raisonner sur un cas moyen et oublier les cas extrêmes ; en particulier, les professionnels (ou l'expert en action) changent souvent de stratégies selon le contexte, ce qui supposerait de coiffer les stratégies de résolution de problèmes classiques par une méta-résolution basée sur un cadrage du problème (J. Viallaneix, 1989).
- Une maintenance lourde : une base de connaissance présente un degré de formalisme important. L'intégration de nouvelles connaissances est complexe. Elle risque d'entrer en conflit avec des connaissances existantes (→ contrôles d'intégrité).
- La structure hiérarchisée de l'ensemble des règles rend délicate la modification de quelques règles sans modifier l'ensemble; de ce fait, les activités de maintenance sont lourdes et coûteuses[24] [06], [12].

2.9. Conclusion

Les méthodes de fouille de données sont nombreuses. On s'est intéressé surtout aux méthodes de classification et segmentation. On a aussi brièvement abordé les règles d'associations, et les systèmes experts qui peuvent s'intégrer aux méthodes précédentes pour améliorer la classification.

Notons que les méthodes précédentes sont complémentaires : on peut utiliser plusieurs méthodes pour aboutir à une bonne classification, par exemple utiliser la segmentation comme tâche préparatoire à la classification.

Il reste de nombreuses méthodes qu'on n'a pas abordé comme l'analyse de données exploratoire, les algorithmes génétiques, les cartes de Kohonen, l'analyse discriminante, la régression logistique, les SVM, les méthodes de réduction des dimensions, la régression linéaire, la régression multiple, le raisonnement à base de cas, les agents intelligents, ….

[24] Beaucoup moins que s'il faut changer des algorithmes.

Chapitre 3. La Fouille de Textes

3.1. Introduction

Qui ne s'est jamais perdu dans la profusion de ses favoris Internet ? N'a jamais téléchargé en cas de besoin, des études, rapports et autres livres blancs finalement jamais lus ? Recherché sur son disque dur un contenu « mis de côté » sans se souvenir où ni sous quelle forme (mel, document Word, PDF, page html, fichier multimédia) ? Rebondi indéfiniment de thématique en thématique au cours d'une séance de veille et été asphyxié par les résultats proposés par un moteur de recherche… ?

En 2000, une étude de Cyveillance estimait que 7 millions de nouvelles pages venaient s'ajouter quotidiennement au Web, le faisant doubler de volume tous les 6 mois. En 2003, une étude de l'université de Californie estimait la production annuelle à 250 Mo d'information par habitant de la planète.

Aujourd'hui, face à la progression exponentielle du nombre d'internautes, aux nombreux programmes de numérisation en cours et aux coûts dérisoires du stockage de masse, difficile d'imaginer la taille réelle du Web et le volume d'informations disponibles en ligne.

Pour conclure avec les chiffres, les internautes consacreraient :

- 70 % de leur temps à chercher l'emplacement de l'information,
- 25 % à isoler l'information utile de la multitude des informations accolées,
- 4 % à consulter des documents relatifs au thème de la recherche,
- …et seulement 1 % à comprendre ce qu'ils sont venus chercher.

L'*infobésité* est le mal de notre société de l'information. Il ne s'agit pas seulement d'un problème d'abondance, mais aussi d'assimilation et de réemploi de l'information, pour l'individu et pour l'organisation, avec les conséquences qu'on imagine sur la productivité et l'efficacité.

Comment lutter ? Quels sont les outils émergents, au-delà de la stricte organisation personnelle ? [19].

Lorsque les volumes de documents en jeu sont tels qu'une analyse manuelle de leur contenu n'est plus possible, un ensemble de technologies, désignées sous le nom de *fouille de textes*, permettent de simplifier l'organisation et l'analyse des contenus numériques entrant et sortant de l'entreprise.

Nous avons beaucoup d'outils à notre disposition pour ce problème. Beaucoup de ces outils sont dérivés de travaux de recherche d'information [27], traitement de langage naturel (analyse de texte et extraction d'information) [28], de statistiques et d'analyse de données (analyse factorielle, classification,…), d'intelligence artificielle (techniques d'apprentissage, règles d'inférences) et de théorie de l'information [29].

Le terme fouille de texte a été inventé pour décrire ces types d'outils. Il n'y a aucune définition standard de fouille de textes.

Une définition générale inclut tous les types de texte qui traite ce problème de rechercher, organiser, et analyser les renseignements textuels. Une définition plus formelle restreint la fouille de texte pour signifier la création de nouveaux renseignements qui ne sont pas évidents dans une collection de documents [30]. Les nouveaux renseignements sont définis comme un modèle, une tendance, ou un rapport qui ne peut pas être facilement glané en lisant des documents individuels. Nous utilisons le terme document pour faire référence à toute unité de texte, tel qu'une page de web, un email, un article formaté, un ensemble de diapositives, ou un fichier de texte ordinaire.

Une autre définition précise que la fouille des textes est une technique permettant d'automatiser le traitement de gros volumes de contenus texte pour en extraire les principales tendances et répertorier de manière statistique les différents sujets évoqués [23]

On appelle fouille de texte un ensemble de techniques d'analyse linguistique, essentiellement statistique, visant à faire émerger des relations, a priori inconnues, entre éléments de connaissance (matérialisés par des séquences textuelles). La fouille de texte est particulièrement utilisée pour la veille, où la découverte de nouvelles connaissances à partir de texte est primordiale. [22]

La fouille de texte a pour objectif de :

- quantifier un texte ou les parties d'un texte pour en extraire les structures signifiantes les plus fortes ;
- établir des liens entre les termes et les documents ;
- analyser les documents en leur associant des informations qualitatives et quantitatives structurées ;

- établir des règles de classification automatique de documents par la construction de modèles statistiques capables de prédire l'appartenance d'un nouveau document à une catégorie déjà définie [20].

3.2. Fouille de Données Vs Fouille de Textes

Une des spécificités de la fouille de textes est que les documents sont écrits pour des lectures par l'homme. La fouille de textes est une extension des techniques traditionnelles de fouille de données à des données semi-structurées qui seront transformées en données analysables.

La démarche de fouille de textes est similaire à une démarche de fouille de données classique, sa particularité réside dans les étapes spécifiques de préparation des données dans la mesure où les données nécessitent un travail de "structuration". C'est le passage du texte au nombre qui le singularise.

Par contre, le travail de préparation des données effectué, des méthodes statistiques traditionnelles peuvent être intégrées, il en ressort donc qu'un outil de fouille de textes doit disposer des fonctionnalités usuelles d'un outil de fouille de données [20].

Parce que la fouille de données suppose ses données déjà entreposés dans un format structuré, son prétraitement se concentre sur deux tâches critiques : *nettoyer* et *normaliser* les données et créer de nombreuses tables. Par contre dans les systèmes de fouille de textes les opérations du prétraitement sont centrées sur l'identification et l'extraction des traits représentatifs pour les documents en langue naturelle. Les opérations de prétraitement sont chargées de transformer les données semi-structurées entreposées dans les collections des documents dans un format intermédiaire explicitement structuré.

Cependant, les systèmes de fouille de textes n'appliquent pas habituellement leurs algorithmes de découverte de connaissances sur des collections de documents non préparées.

Une part considérable de la fouille de textes est consacrée à ce qui est connu communément sous le nom d'opérations de prétraitement [26].

La fouille de données et la fouille de textes, cherchent des informations cachées et utilisent des algorithmes communs d'intelligence artificielle, apprentissage automatique, et statistiques.

- Pendant que la fouille de données travaille avec les données numériques structurées, la fouille de texte travaille avec des textes semi-structurés. Typiquement, les données utilisées pour la fouille de données sont extraites, transformées, et chargées dans un entrepôt de données. L'usage efficace des applications de fouille de données est conditionné par des sources fiables.
- Par contre la fouille de textes essaie de construire un modèle de données qui est supposé imprécis. Transformer ces données pour enlever toutes les inconsistances n'est pas toujours possible et on suppose que le texte non structuré a quelques informations qui ne peuvent pas être utilisés et peuvent être même fausses. Les bonnes méthodes de fouille de texte utilisent juste assez d'informations pour construire un modèle général – et ignorer les données qui ne peuvent pas être utilisées – pour faire des prédictions exactes. C'est accompli en cherchant des traits majeurs qui définissent les données d'un ensemble d'apprentissage et en négligeant les traits qui sont rares ou exceptionnels : de tels traits rares ne sont pas jugés utiles dans un modèle général.

Les entrepôts de données peuvent être énormes et les compagnies ont une portion substantielle de données dans ces magasins de données structurés. Cependant, jusqu'à 80 ?% ou plus des renseignements d'une compagnie sont entreposées dans la forme de textes semi-structurés. Ces renseignements sont éparpillés sur beaucoup d'ordinateurs et contrairement à la fouille de données il n'y a pas un seul dépôt où miner. Les problèmes de fouille de textes ne sont pas magnifiés seulement par l'environnement distribué et le grand volume de renseignements, mais parce que le texte possède de nombreuses dimensions.

Une représentation d'un document par vecteur utilise une dimension par mot unique. Le modèle vectoriel est commun dans la recherche d'information. Les méthodes de segmentation et de classification calculent des milliers de similarités entre les vecteurs même quand les collections de documents sont petites. Heureusement, la puissance de calcul a augmenté régulièrement et l'exactitude est de plus en plus grande.

Au début, on supposait qu'une fois le texte converti en données numériques, les mêmes outils de fouille de données pourraient être appliqués. La signification des données numériques dans une base de données est facile à comprendre et à évaluer. Cependant, les données numériques textuelles sont souvent basées sur les probabilités dues à l'incertitude dans l'extraction du sens à partir des textes. Bien que quelques algorithmes de fouille de données et

de fouille de textes aient des origines communes, les outils séparés pour chaque type de données sont préférés [25].

3.3. Les tâches

La fouille de textes n'est pas un remplacement pour la recherche d'information ou le traitement du langage naturel. Les techniques qui permettent d'organiser un corpus de documents textuels selon leur contenu ont un spectre d'utilisation très large [84]. La fouille de textes cherche des réponses aux questions difficiles ou impossibles à résoudre avec les seuls moteurs de recherche. Des exemples de tels services incluent :

- Résumer des documents qui décrivent une consommation du produit dans certaine région.
- Étudier des réclamations des clients, raisons des changements de comportements de consommation, analyse de l'image de l'entreprise, ….
- Faire la gestion de la relation client : orienter les mails clients reçus sur le site vers les services adéquats et les aider à répondre le plus rapidement et correctement possible [87].
- Connaître les réseaux relationnels des personnes ou entreprises.
- *Veille technologique et stratégique* : sur les produits et les tendances d'un marché, sur la concurrence, sur la qualité des prestations fournies, … [20], identifier les actions ou faits relatifs aux stratégies des entreprises (prise de participation, fusion, acquisition, ouverture de filiales) [85], classer des news, rapports, brevets, … selon des rubriques métiers ou selon des profils de veille [86].
- *Veille scientifique* : identifier les thématiques de recherche sur un domaine, identifier les relations existantes entre les acteurs (co-auteurs, citations, co-citations) de la recherche à partir de la littérature scientifique et technique telle qu'elle est représentée dans les bases documentaires. Aussi, Identifier des universités bien connues pour la recherche dans l'énergie solaire, et trouver les adresses de courrier électronique et noms d'individus pour contacter ces universités
- *Sécurité civile* :
    - identifier les actions ou faits relatifs aux personnes ou organisations citées dans des rapports de police, établir des corrélations entre des liens, des types d'événements, des armes, des drogues,…
    - assurer une surveillance épidémiologique,

- contrôler niveaux et nature de pollution dans la ville X en vue d'alertes automatiques.

- Citer les différents types d'automobiles électriques fabriquées, basées sur des traits communs tels que la dimension et le type du moteur, la matière du corps, la capacité, et l'espace de la cargaison,…
- Questions ouvertes, sondage, enquête d'opinion et de satisfaction, [20]
- Comprendre mieux le positionnement d'un discours, d'une thèse, d'un communiqué,
- Comparer des textes sur un même thème afin d'en déterminer les points communs comme les différences stylistiques ; analyse de la presse, synthèse d'articles, … pour mesurer les points faibles et les points forts [21] ; synthétiser des contenus (Résumés de larges documents, Résumés statistiques d'enquêtes d'opinion) [78]
- Alimenter des bases de données (Prise de participation des entreprises, Agenda de manifestations culturelles)
- Regrouper les documents en classes homogènes (Routage et analyse d'email, filtrage, catégorisation de documents, Filtrage de mails, Classification de dépêches, Elimination de spams [82]). Classifier les spams reçus dans les trois mois derniers. Décrire les catégories et spécifier les fréquences dans chaque catégorie.
- construction d'annuaires dans le E-business.
- Analyse socio-psychologique : contenu d'entretiens, interviews, récits d'enfants, … [20]

3.4. La recherche des modèles et /ou des tendances

Le noyau des fonctionnalités des systèmes de fouille de textes réside dans l'analyse des concepts, la co-occurrence des modèles dans le document de la collection. En effet, les systèmes de fouille de textes comptent sur les approches algorithmiques et heuristiques pour dégager des distributions, ensembles fréquents, et associations de concepts inter-document pour permettre à l'utilisateur de découvrir la nature et les rapports de concepts reflété par la collection dans son ensemble.

Exemple

Un rapport potentiel peut être inféré entre deux protéines p1 et p2 par un modèle relationnel inter-document :

1. plusieurs articles mentionnent la protéine p1 par rapport à l'enzyme e1,

2. quelques articles décrivent des similitudes fonctionnelles entre les enzymes e1 et e2 sans faire référence à aucuns noms de protéines,
3. plusieurs articles lient l'enzyme e2 à la protéine p2.

La relation p1/p2 n'est pas fournie par un document seul par la collection. Les méthodes d'analyse de modèle cherchent à découvrir des rapports du co-occurrence entre concepts comme reflété par la totalité du corpus [26].

3.5. Les fonctions

Dans la section précédente nous avons vu plusieurs tâches de fouille de texte. Comment ces fonctions s'articulent en applications complètes ? Une fouille de textes peut s'organiser en un modèle multicouches. La figure 3.1 pressente un modèle à multicouches dont l'information circule vers le haut.

À la couche inférieure, les documents d'entrée sont reçus. Ceux-ci sont convertis en texte non structuré avec un filtre. L'entrée à la couche inférieure reçoit les textes non structurés. Les données structurées et informations formatées sont entreposées comme métadonnées La première couche pour traiter le texte découpe le flux de texte en unités appelées des lexies (tokens) Une lexie est un mot, nombre, signe de ponctuation, ou toute autre séquence de caractères qui devraient être traités comme une unité seule.

<table>
<tr><td>Résumé
Segmentation
classification</td><td>Questions/réponses</td><td>Moteur de recherche</td></tr>
<tr><td>Génération des vecteurs</td><td>Extraction des entités</td><td>Index</td></tr>
<tr><td colspan="3">Assemblage des lexies</td></tr>
<tr><td colspan="3">Création des lexies</td></tr>
<tr><td colspan="3">Filtre des documents</td></tr>
<tr><td colspan="3">↑ mels, page web, document ↑</td></tr>
</table>

Figure3.1 fouille de textes : modèle multicouches

(l'information circule de bas en haut).

Dans la couche du texte, les lexies sont assemblés en lexies composés Un ensemble de fonctions dans la prochaine couche utilise les lexies assemblés comme entrée. Nous n'avons pas besoin de l'ordre de lexies pour représenter un document [25].

L'ensemble des mots uniques apparents dans un document ou dans la collection pondérés selon leur importance dans le document est une représentation raisonnable. Nous utilisons le terme vecteur pour telle représentation depuis que chaque lexie est une dimension et a quelque poids.

Tandis que l'ordre des mots n'est pas considérable dans la génération des vecteurs, il est important dans le contexte pour extraire des entités, des parties de discours, des phrases, et des expressions (micro structures).

La segmentation du texte en phrases paraît assez simple. Cependant le point ne marque pas toujours une fin de phrase. Souvent le texte extrait des pages de web n'est pas grammaticalement correct et aucune ponctuation ne délimite des phrases. L'index d'un moteur de recherche traque des lexies, associées à des documents Pour chaque lexie d'un document, l'index maintient une entrée correspondante qui inclut l'information au sujet de l'occurrence de la lexie dans le document. Le moteur de recherche utilise l'index pour localiser des documents et retourne une liste de documents à l'utilisateur final.

3.6. Méthodes utilisées pour la fouille de textes

La fouille de textes fait appel à l'analyse de données qui se caractérise par deux grandes familles de méthodes :

- *Les méthodes de classification* qui produisent un regroupement d'objets ou d'individus décrits par un certain nombre de variables ou de caractères (classification de type nuées dynamiques, Classification Ascendante Hiérarchique - CAH -). La fouille de textes fait appel aux *techniques de modélisation*

    - Régression logistique ;
    - Arbres de décision ;
    - Analyse discriminante de Fisher ;
    - Réseaux de neurones [20]

- *les méthodes factorielles* qui font essentiellement des représentations graphiques caractérisant les liens entre les différents critères (Analyse en Composantes Principales, Analyse Factorielle des Correspondances, Analyse des Correspondances Multiples)

3.7. Les étapes de la fouille de textes

La fouille de textes comprend une succession d'étapes permettant de passer des documents au texte, du texte au nombre, du nombre à l'analyse, de l'analyse à la prise de décision.

La première étape consiste en la récupération des documents :

- étude du document texte ;
- création de la base de données ;
- stockage des documents dans la base de données.

La seconde étape consiste à identifier les unités textuelles : les formes graphiques qui serviront de base à l'analyse :

- Normalisation ;
- Segmentation ;
- Lemmatisation ;
- Numérisation ;
- Segments répétés.

L'étape suivante, par la construction de tableaux lexicaux, permet de procéder à des analyses statistiques descriptives et discriminantes (Construction de tableaux lexicaux ; Etude lexicométrique ; AFC, ACP, ACM (réduction des dimensions des données) ; Classification ; Modélisation [20]).

3.8. Applications

On présente dans cette section quelque applications de fouille de textes [25]

3.8.1 Les études

Dans leur rivalité, la plupart des compagnies offrent constamment plus ou mieux, ce qui fait évoluer les tâches et l'environnement de l'employé. Avant faire des changements, ce serait utile de mesurer les réactions de l'employé. Par exemple, si une compagnie projette de changer les avantages de retraite pour ses employés, une étude peut être conduite sur leurs opinions. Même les questions à choix multiples sont trop fermées pour analyser ces opinions

sous chaque angle et en fournir l'état complet. Les questions (ouvertes) qui acceptent des réponses en langue naturelle sont difficiles à traiter.

La méthode de fouille de données travaille bien quand les choix multiples mènent rapidement à des tableaux et peuvent être résumés. Mais face à des milliers de réponses en langue naturelle, c'est très difficile de traiter chaque réponse manuellement et de disposer en tableau les résultats.

3.8.2 Intelligence économique

Elle est utile aux compagnies d'une même industrie pour suivre les produits de l'un et de l'autre, et développements. Les constructeurs automobiles comme Ford surveillent probablement directement les sites web de Toyota (pour se faire une idée des actions commerciales, de la production, des investissements, développements et recherches) et indirectement Toyota à travers la presse professionnelle, les forums, les nouvelles du METI[25].

Faire ceci sur une base journalière et manuelle est lent. Les sites des grandes compagnies telles que Ford ou Toyota peuvent avoir plusieurs centaines de pages web, sans compter celles des sites plus ou moins connexes. Une approche automatisée qui télécharge périodiquement et analyse les pages relatives à un concurrent est plus effective.

3.8.3 La gestion des clients

Un client peut appeler un centre d'assistance technique, ou envoyer un mel directement ou via une page de web. Ces renseignements peuvent être rassemblés et peuvent être entreposés dans un dépôt pour les analyser. Une vue globale des réactions de client dans la forme d'une taxonomie rend facile à identifier les classes de produits et les produits individuels objets des plaintes. Le problème le plus commun avec un produit peut ainsi être extrait. La catégorisation des réclamations des clients facilite le routage des messages vers l'expert approprié comme la rédaction d'un rapport sur le problème. Quelques produits sont complexes et la grande compagnie typique a plus d'un niveau de support. Si une question ne peut pas être ré-

[25] d'après wikipedia, « Le METI (Ministry of Economy, Trade and Industry) est le ministère des finances, du commerce extérieur et de l'industrie du Japon. Il s'appelait initialement MITI (Ministry of International Trade and Industry) et a été fondé en 1949. Il encadre l'activité économique pour l'État. Il oriente les stratégies des keiretsu (conglomérats). Il agit à travers l'imbrication entre la classe politique, l'administration et les firmes multinationales. C'est ce qui donne à l'économie japonaise sa force.

* Il informe les entreprises japonaises sur les marchés étrangers et les nouvelles technologies.
* Il surveille les échanges extérieurs et intérieurs du Japon.
* Il incite les groupes dans leurs choix.
* Il favorise le développement des technopôles.
* Il se pose comme le chef d'orchestre de la croissance japonaise .

pondue au premier niveau de support, la requête du client est déroutée sur un spécialiste du niveau suivant.

3.8.4 La recherche médicale

Localiser des articles qui utilisent certains termes médicaux dans des contextes définis intéresse souvent les chercheurs. Ce type de recherche n'est pas faisable avec un moteur de recherche : il peut y avoir des myriades de combinaisons de termes et ce n'est pas pratique de soumettre manuellement des questions individuelles pour chaque combinaison à un moteur de recherche.

3.8.5 La recherche légale

Il n'est pas toujours facile de corréler des rapports de police, des déclarations écrites ou des actes notariés spécifiques à une affaire, avec le droit (législation, réglementation, jurisprudence). Les outils automatisés qui peuvent extraire et résumer les renseignements ont alors beaucoup d'avantages sur un moteur de recherche, car le chercheur ne peut toujours savoir les mots-clé ou la terminologie spécifique touchant les renseignements dont il a besoin.

3.8.6 Connaître l'opinion publique

Savoir « qui pense quoi » au sujet d'une question précise est d'un grand intérêt pour les politiciens et les sociologues. Les outils de fouille de textes peuvent augmenter les résultats des élections.

Les citoyens s'expriment sur le web[26], et la plupart de ces informations sont disponibles au public. Malheureusement, ces renseignements sont éparpillés et utiliser un moteur de recherche ne résoudra pas le problème du recouvrement. Après avoir choisi certaines sources de renseignements pertinentes sur le web, les données rassemblées pourraient être analysées pour corréler opinions impliquées, fréquence, et catégories de citoyens. Si une question est fréquemment discutée, comment est-ce que les gens ressentent le sujet ?

3.8.7 Shopping

Le magasinage sur le web veut trouver le bon produit au bon prix. Les prix varient d'un site à un autre et ce n'est pas pratique de visiter et parcourir chaque site manuellement.

[26] Biais classes moyennes et supérieures.

Un site comparatif pourra se baser sur des analyses automatiques des sites vendeurs, à partir d'une liste de critères.

3.8.8 La recherche académique

Un crawler[27] est un logiciel d'indexation conçu pour passer en revue des sites Web et pour télécharger l'information contenue dans ces sites. L'information qu'il télécharge est le code source HTML. Dans le cas des moteurs de recherche, ce code source est stocké dans une grande base de données et plus tard analysé et classé dans l'index des moteurs de recherche.

Il peut visiter pour un département toutes les pages web sur un thème donné, et extraire les titres et autres renseignements pertinents de toutes les publications et rapports rencontrés. Cette liste peut être catégorisée et les auteurs-clés dans un département peuvent être identifiés. La même procédure peut être appliquée à de multiples départements à travers les universités et les résultats peuvent être coulés dans une seule hiérarchie de catégories. Les universités qui ont des intérêts communs peuvent être localisées et les sujets qui paraissent populaires, détectés. Les rapports entre départements peuvent être identifiés à travers les liens hypertextes comme à travers les citations dans les publications.

3.8.9 Le triage automatisé

On se pose la question du classement automatique d'un ensemble de mémoires académiques [31]. Un programme peut-il distinguer automatiquement les bons essais des mauvais, et leur assigner des niveaux basés sur les analyses textuelles ? Les premiers systèmes de triage automatisé basés sur les traits simples traitaient des spams et leur assignaient les plus hauts niveaux à des essais pauvrement écrits. Cependant, il y a une corrélation forte entre une bonne écriture et certains traits du texte. Le problème principal est de trouver automatiquement et d'extraire ces traits.[28]

3.8.10 Catégorisation des textes

Une des raisons de construire une taxonomie de documents est de trouver plus facilement des documents pertinents.

---

[27] Spiders ou robots.

[28] Il y a aussi des outils de détection de plagiat…

Le problème de catégorisation peut être décrit comme la classification de documents dans de multiples catégories (figure 3.2). Nous avons un ensemble de catégories n $\{c_1, c_2, \ldots, c_n\}$ auxquelles nous devons assigner m documents $\{d_1,d_2,\ldots d_m\}$, avec n<<m.

Chaîne du type :

document -> descripteur ou profil du document-> classement, en cherchant la classe dont le-profil est le plus corrélé au profil du document.

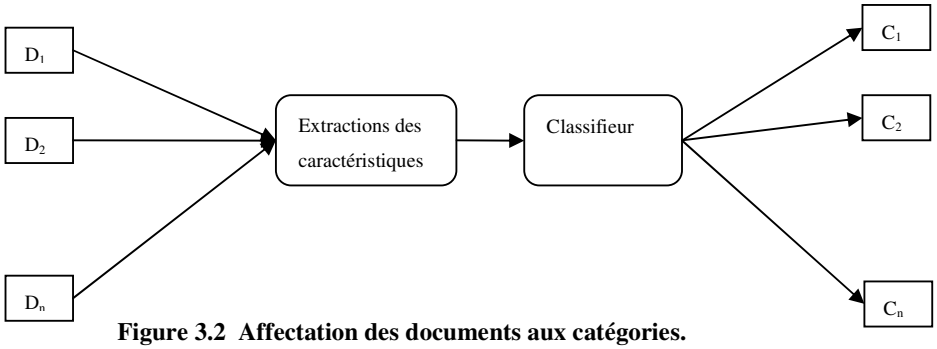


Figure 3.2 Affectation des documents aux catégories.

Les catégories n sont prédéfinies avec des mots-clés spécifiques qui différencient toute catégorie C_i de chaque autre catégorie C_j[29]. Le processus d'identifier ces mots-clés est appelé l'extraction du trait. Ce ne sont pas tous les mots d'un document qui sont des discriminateurs utiles.

La catégorisation de textes peut être utilisée sur des flux de renseignements dynamiques qui ont besoin d'être organisé. Nous définissons les renseignements dynamiques comme des mels, des articles et nouvelles, blogs, articles scientifiques, brevets, et données légales. Les applications incluent l'acheminement automatique des questions des clients, les demandes médicales, et la recherche d'entités dans le flux de renseignements. Ce type de renseignements est produit journellement, et l'utiliser est difficile sans quelque catégorisation [25].

3.9. Conclusion

Les données textuelles en format « libre » disponibles sur support informatique représentent 70% des données disponibles et se retrouvent sous forme de rapports, de courriers, de publications, de manuels, etc. Les textes contiennent des informations et des connaissances utiles et parfois critiques pour la gestion et la prise de décision dans les entreprises, et

[29] Les jeux caractéristiques doivent être très différents, mais des mots peuvent être communs à des catégories connexes.

s’attaquer aux textes signifie une maturation de l’informatique, débutée timidement en 1957 avec les balbutiements de l’informatique documentaire.

La fouille de données textuelles vise à définir des stratégies pour exploiter les textes en format libre. Elle fédère des thématiques issues des sciences de l'information, de la linguistique, de la statistique et des méthodes d'apprentissage (I.A).

La préparation des données est une étape importante, si ce n'est primordiale, du processus d'extraction de connaissances à partir de données. En schématisant, il s'agit de définir au mieux les individus et la représentation utilisée pour l'apprentissage. La qualité du modèle de prédiction dépend donc grandement de la qualité de la préparation effectuée en amont.

Chapitre 4. Analyse de Données Textuelles en Langue Arabe

4.1. Introduction

La masse d'information véhiculée à travers les réseaux est de plus en plus profuse et incontrôlable. La tendance est celle d'une surinformation tentaculaire, et le problème de la maîtrise de l'information risque de devenir très délicat voire nocif, à moins que des outils de collecte, de traitement, de diffusion ciblée et d'exploitation de l'information ne viennent aider les utilisateurs à mieux gérer ces avalanches d'informations [41]. La recherche accorde, ces dernières années, beaucoup d'importance au traitement des données textuelles [88, 89]

Dans ce cadre, ce chapitre expose un ensemble de procédés pour l'analyse de données textuelles et leur préparation pour les algorithmes de classification. Il s'agit en premier lieu d'un procédé de lemmatisation dans l'analyse morpho-lexicale. Cette étape est cruciale car elle est la base de tous les applications de *Traitement Automatique des Langues Naturelles* (TALN): traduction automatique, indexation, recherche d'information, réponses aux questions, … et aussi de la fouille de textes.

Une deuxième étape est d'appliquer un processus de filtrage et d'indexation,

Troisièmement un raffinement syntaxique, et enfin l'application de méthodes statistiques pour le calcul des tables de fréquences lexicales, qui sont l'entrée des systèmes de catégorisation.

La langue arabe est une langue riche morphologiquement et représente de grands défis pour les applications de traitement automatique du langage naturel [90].

Le traitement automatique sur la langue arabe pose des problèmes majeurs [42] [43], dont :

- l'ambiguïté issue de l'absence des voyelles [44], qui exige des règles morphologiques complexes [45] ;

- la reconnaissance des formes fléchies, car l'arabe est une langue fortement flexionnelle [46] ;
- et l'absence de travaux publiés sur l'extraction d'information en langue arabe à travers l'utilisation de modèles statistiques de langage [80].

La lemmatisation

C'est une procédure ramenant un mot fléchi (par exemple, la forme conjuguée d'un verbe) à sa forme de référence (dite *lemme*), quelle que soit la forme sous laquelle le mot apparaît dans un texte. La lemmatisation sert ainsi à la reconnaissance morphologique des mots d'un texte.

" Stemming forms of the same word are usually problematic for text data analysis, because they have different spelling and similar meaning so stemming is a process of transforming a word into its stem (normalized form) "[47].

4.2. Analyse morphologique

L'analyse morphologique a pour but de vérifier si un mot fait partie de la langue traitée ou non. Elle consiste à décomposer les mots en morphèmes [48], [49] sans tenir compte des liens grammaticaux entre ces derniers.

On peut souligner que l'analyse envisagée ici diffère de l'analyse lexicale classique en compilation, car elle ne se basera pas sur un automate, mais sur certaines notions concernant la morphologie des entités de la langue à analyser.

Contrairement aux langues néo-latines, l'arabe est une langue agglutinante ; les articles, les prépositions et les pronoms collent aux adjectifs, noms, verbes et particules auxquels ils se rapportent ; ce qui engendre une ambiguïté morphologique au cours de l'analyse des mots [91].

4.2.1 Principe de l'analyseur

Pour analyser une phrase, un système d'extraction d'informations effectue successivement les analyses : [95, 96, 97, 98, 99]

- lexicale (segmentation de la phrase en mots)
- morpho-syntaxique(étiquetage des mots par leur catégorie syntaxique et association de chaque mot à sa forme canonique ou pseudo-racine)

Il procède en premier lieu par une normalisation qui transforme le document en un format plus facilement manipulable [50] (elle est nécessaire à cause des variations qui peuvent

exister lors de l'écriture d'un même mot) puis la segmentation découpe le mot pour que chacune des parties obtenues soit une entité lexicale. Cette segmentation isolera les préfixes et suffixes du mot. La partie restante correspondra à la racine dans le cas où la segmentation est poussée jusqu'à la fin (par l'utilisation de la notion du schème). Dans le cas contraire la partie restante est appelée une base. Cette étape est une tâche délicate du fait que l'arabe est une langue flexionnelle et fortement dérivable [51].

L'analyseur morphologique ne peut fonctionner sans l'aide d'un dictionnaire contenant les unités lexicales. Cette étape d'analyse lexicale permet de vérifier si l'unité lexicale appartient bien à la langue, mais doit aussi vérifier la compatibilité entre les différents constituants du mot. Une troisième étape interprète les différents éléments obtenus par la segmentation en se basant toujours sur la notion de schème, et retourne comme résultat les valeurs morphologiques (hors contexte) des différentes unités lexicales, avec l'intervention probable de règles micro-syntaxiques. Le schéma qui suit récapitule les étapes de l'analyse [52].

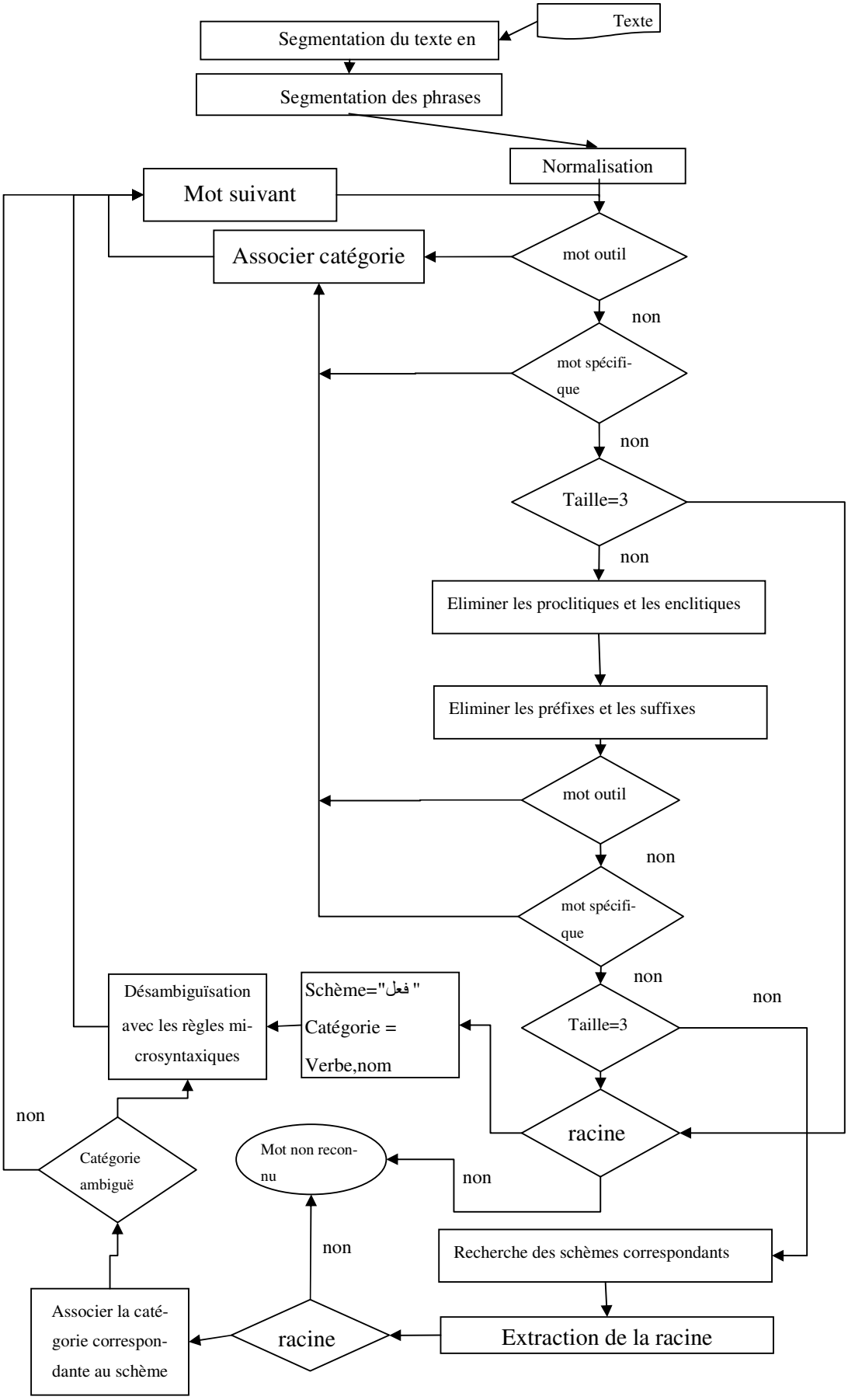


Figure 4.1: Schéma général de l'analyse morphologique.

4.2.2 Les dictionnaires nécessaires

Dictionnaire des schèmes

Dans ce dictionnaire, on trouve toutes les informations utiles lors de la codification des mots sous forme de structure de traits. Il faut spécifier pour chaque schème son type : nominal ou verbal.

On trouve aussi dans ce dictionnaire une codification des schèmes pour faciliter la recherche du schème et de la racine de l'entité comme il va être expliqué ci-après.

Dictionnaire des racines

Dans ce dictionnaire sont stockées toutes les racines représentatives de la langue, dans notre système les racines les plus utilisées dans la langue arabe. Ceci nous permettra de vérifier la compatibilité entre la racine et le schème, évitant ainsi toute décomposition erronée.

Dictionnaire des mots-outils

Nous considérons comme mot-outil, tout mot qui reste invariant quel que soit son contexte (à l'exception des noms propres et mots communs) tels que les pronoms, particules,…etc. Nous ne mettons dans ce dictionnaire que les mots-outils isolés.

Ce dictionnaire contient les prépositions, les particules qui ont un effet de rection sur les verbes à l'accompli et à l'inaccompli, les particules de coordination et d'autres particules. Il faut préciser que les particules vont être divisées dans quelques mots outils, des spécifications qui aident dans l'analyse syntaxique [53].

Remarque

Les spécifications mises dans les dictionnaires : schème, racine, mot-outil, sont les traits qui seront utilisés dans la partie analyse syntaxique.

Dictionnaire des mots spécifiques

Les mots de la langue arabe ne sont pas tous décomposables en racine et schème. Il existe en plus des mots-outils, une catégorie dite de *mots spécifiques*, formée de mots qui n'ont pas une origine arabe, de noms propres et communs. La création d'un dictionnaire de ces mots spécifiques s'avère nécessaire.

Pour les noms communs et noms propres, un fichier est créé en laissant à l'utilisateur la possibilité de rajouter les noms qui manquent d'une part, d'autre part il suffit de considérer les décompositions mises en échec comme étant nom propre ou commun.

Dictionnaire des mots vides du domaine

Cet *antidictionnaire* est réservé aux mots non significatifs et les moins porteurs de sens dans un domaine donné (mots non pertinents).

4.2.3 Structure des dictionnaires

Structure du dictionnaire des schèmes

Rappelons que le schème utilisé par les grammairiens arabes comme point de départ de toute morphologie de la langue est 'فعل'. Par conséquent, à chaque mot du lexique arabe est associé un schème qui est le même mot sauf les lettres de sa racine qui sont remplacées par les lettres de la racine 'فعل'.

Exemple

mot = 'صالح' schème = 'فاعل'.

Le dictionnaire des schèmes est structuré de la façon suivante :

Wzn i	ListeInfixe i	Catégorie
افتعل	13	Verbe

Figure 4.2 : structure du dictionnaire des schèmes.

Le champ Wzn contient la chaîne consonantique du schème. Le champ listeInfixe i contient la position des lettres autres que les lettres de la racine dans le schème et le champ catégorie donne la catégorie grammaticale (nom, verbe, ...).

Cette structure facilite la recherche des schèmes qu'on va présenter dans la suite.

Structure du dictionnaire des racines

Asl
.....
دخل
……..
هرب

Figure 4.3 : structure du dictionnaire des racines.

Le champ Asl contient la chaîne consonantique de racine. Les racines sont insérées par ordre alphabétique. Il est composé toujours de trois caractères.

Remarque

Il existe dans la langue arabe des verbes quadrétaires comme دحرج،زلزل

Notre système ne traite pas ces exceptions car ils sont rares.

Structure du dictionnaire des mots outils

Hrf	Classe du successeur
في	Nominale
قد	Verbale
و	Commune

Figure 4.4 : structure du dictionnaire des mots outils.

Le champ Hrf contient la chaîne consonantique du mot-outil, le champ classe détermine le type du successeur du mot-outil en précisant s'il est un verbe ou nom ou commun entre les deux.

Structure du dictionnaire des mots spécifiques

Jmd
أحمد
سطيف
كمبيوتر

Figure 4.5 : structure du dictionnaire des mots spécifiques.

Le seul champ *Jmd* contient la chaîne consonantique du mot spécifique.

4.2.4 Méthodologie utilisée

Les grammairiens de la langue arabe s'accordent à dire que tout lexème arabe à l'exception des noms propres, de quelques noms communs et des mots-outils, est le résultat de la combinaison d'une racine et d'un schème spécifique. On peut schématiser l'extraction du schème et de la racine à partir du mot: مفاتيح (clés) par le schéma suivant :

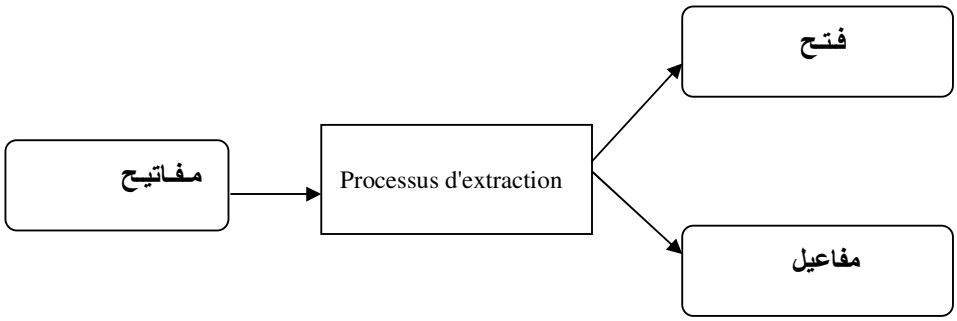


Figure 4.6 : extraction du schème et racine.

Pour vérifier qu'un mot appartient aux lexèmes arabes il suffit de trouver sa racine et le schème correspondant suivant cette méthode. Le travail proposé se basera sur trois étapes essentielles :

- le découpage (segmentation) ;
- la recherche des schèmes et des racines ;
- l'interprétation.

Normalisation

A cause des variations qui peuvent exister lors de l'écriture d'un mot arabe, une normalisation s'avère nécessaire, il s'agit de transformer le texte original dans un format standard plus facilement manipulable. Le texte est normalisé comme suit [50]:

- suppression des caractères spéciaux, des chiffres, de la ponctuation ;
- suppression du signe diacritique *chadda ;*
- remplacement de أ آ par ا ;
- remplacement de ي par ى;
- remplacement de ة par ه .

Découpage

Découper un mot ou le segmenter consiste à extraire ses différentes parties (préfixe, racine, suffixe,…).

Techniques de segmentation

Résumons quatre techniques.

a) Première technique [54]

Elle consiste à découper le mot en trois éléments :

proclynique + base + enclynique.

b) Deuxième technique [55]

Elle consiste à découper le mot en 5 éléments

proclitique + préfixe + base + suffixe + enclitique.

c) Troisième technique [56]

Elle découpe le mot en : préfixe + racine + infixe + suffixe. Une procédure vérifie l'existence de la chaîne "ال " dans le mot. Si cette chaîne existe alors le mot sera considéré comme un nom ou un adjectif, et la décomposition se fera avec les règles morphologiques des noms ou des adjectifs, sinon avec la décomposition des règles des verbes.

d) Quatrième technique [57]

Elle décompose le mot en Préfixe + base + suffixe, la liste des préfixes et suffixes est la combinaison de toutes les lettres additionnelles (greffes).

Comparaison des différentes techniques

Table 4.1 : comparaison des techniques de segmentation.

Méthodes	Avantages	Inconvénients
[57]	Simplicité.	Trop d'ambiguïté. Beaucoup de tests inutiles.
[56]	Diminuer le nombre de tests dans le cas où il trouve la chaîne"ال".	Ambiguïté si la chaîne "ال" appartient aux radicaux. Si la chaîne "ال" n'appartient pas au mot alors perte de temps. La suppression des voyelles longues risque la perte des radicaux.
[55]	Moins d'ambiguïté.	Nécessité de plusieurs tables de compatibilité, ce qui complique le module de découpage.
[54]	Nombre de constituants limités à 3 ce qui accélère le découpage.	Difficile de former tous les enclyniques et proclyniques de la langue arabe, d'où trop d'ambiguïté.

Ce tableau comparatif nous amène à choisir la troisième technique [55] car elle extrait des racines avec leurs valeurs morphologiques avec moins d'ambiguïté [52].

Pour résoudre l'ambiguïté, Aljlayl et Frieder montrent que la *lemmatisation légère* (approche basé sur suppression de suffixe et de préfixe) surpasse significativement celle basée sur la détection de racine dans le domaine de recherche d'information [58]. Pour notre cas nous avons dépassé la lemmatisation légère qui consiste à déceler si des préfixes ou des suffixes ont été ajouté au mot [59], jusqu'à la suppression de toutes les lettres additionnelles afin d'obtenir la racine seule.

Le principe du découpage est le suivant:

- Découpage du mot en : PRO + BASE1 + ENC.
- Découpage de *base1* en : PRE + BASE + SUF.
- Découpage de *base* en : RACINE + SCHEME.

Application de la technique

Rappelons que le rôle de l'analyseur morphologique est d'extraire la racine des mots.

L'analyse morphologique de la langue arabe est plus complexe que celle de la plupart des langues européennes. On décompose l'analyse morphologique en trois phases principales:

- l'élimination des proclitiques et des enclitiques ;
- l'élimination des préfixes et des suffixes ;
- l'extraction de la racine et du schème correspondant si c'est possible ; sinon le mot est considéré comme un mot spécifique et par conséquent sa décomposition se limite aux deux phases précédentes.

Dans ce travail, nous avons construit un analyseur morphologique qui prend en charge la majorité des formes du lexique arabe (les verbes trilitères sains et les noms sains).

Les étapes de découpage

Comme mentionné précédemment, il existe trois étapes principales de découpage :

- Découpage du mot en *proclitique+base1+enclitique* qui consiste à repérer tous les proclitiques et les enclitiques qui apparaissent dans le mot. Base1 est en général une base (racine + des infixes) munie de préfixes et de suffixes.
- Découpage de la base1 (résultat de la phase précédente) en *préfixes+base+suffixes* ; le principe de cette phase est le même que celui de la phase précédente.
- Découpage de la base en *racine* et *schème* c'est-à-dire trouver un schème parmi les schèmes stockés dans le dictionnaire des schèmes correspondant à la base. La méthode de reconnaissance du schème sera décrite ci-après.

Reconnaissance des proclitiques et des enclitiques

a) Notion de proclitique / enclitique

Certaines particules s'additionnent au début ou à la fin d'un mot pour en changer le sens ou pour avoir un effet sur la rection du mot.

Elles s'appellent les *enclises*. Celles qui viennent au début de mot sont les *proclitiques* comme : «ل » (لام التوكيد), «ل » (لام الأمر),

comme dans les entités «لكريم » ; «ليضرب », et aussi le signe de la détermination « ال».

Alors que celles qui viennent à la fin sont appelées *enclitiques*, par exemple les pronoms affixes compléments comme (هم،ها) dans : (ضربهم،مجلسها)

Remarque

Proclitiques et enclitiques ne sont pas libres d'apparaître au hasard : il existe une certaine compatibilité entre eux illustré dans la table 4.3.

b) Recherche des proclitiques et enclitiques

Fort heureusement la liste des proclitiques et des enclitiques de la langue arabe est limitée. On peut utiliser la liste proposée par [60] ; plusieurs d'entre eux ont été utilisés par [61] pour la lemmatisation. Le découpage du mot en "proclitiques + base1+ enclitiques" ne se limite pas à la recherche d'un proclitique (respectivement un enclitique) parmi la liste au début (respectivement la fin) du mot, mais ainsi à une certaine compatibilité antre les proclitiques et les enclitiques repérés dans le mot à décomposer.

La table suivante donne la liste des différents proclitiques et enclitiques existants :

Table 4.2 : liste des proclitiques et des enclitiques.

Proclitiques	' '	ب	ك	ل	ف	س	أ	ال	كال	لل	فب	فس	فال	فك	فل	فلل	أف	أس	فبال	فكال	بال
Enclitiques	' '	ه	ي	ك	هم	هن	هما	ها	كم	كن	كما	ني	نا								

La table suivante présente les différents cas de compatibilité et d'incompatibilité entre les proclitiques et les enclitiques

- Les cases contenant le caractère ° correspondent à une incompatibilité du proclitique indiqué en tête de ligne et de l'enclitique indiqué en tête de colonne.
- Proclitique et enclitique compatibles donnent une décomposition correcte.

Table 4.3 : table de compatibilité entre proclitiques/ enclitiques.

P\E	''	ه	ي	ك	هم	هن	هما	ها	كم	كن	كما	ني	نا
''													
ب												°	
ك												°	
ل													

ف													
س			○										
أ			○										
ال		○	○	○	○	○	○	○	○	○	○	○	○
بال		○	○	○	○	○	○	○	○	○	○	○	○
كال		○	○	○	○	○	○	○	○	○	○	○	○
لل		○	○	○	○	○	○	○	○	○	○	○	○
فب												○	
فس			○										
فال		○	○	○	○	○	○	○	○	○	○	○	○
فك												○	
فل												○	
فلل		○	○	○	○	○	○	○	○	○	○	○	○
أف													
أس			○										
فبال		○	○	○	○	○	○	○	○	○	○	○	○
فكال		○	○	○	○	○	○	○	○	○	○	○	○

Test de compatibilité

Après l'extraction du proclitique P et de l'enclitique E du mot à analysé, ces deux sous-chaînes sont fusionnées en une chaîne C pour les tester dans une table des incompatibilités (figure 4.7). Si la chaîne C n'y est pas trouvée alors ce proclitique P est compatible avec cet enclitique E.

Exemple

L'analyse du mot (**فبتعلمكم**) donne comme proclitique (**فب**) et enclitique (**كم**), la fusion donne (**فبكم**) ; cette chaîne est compatible selon la table (elle n'existe pas dans la table), alors le découpage est correct.

Liste
بني
…….
كالهما
…….
فكالنا

Figure 4.7 : table des paires incompatibles.

Remarque

La table recense les incompatibilités, moins nombreuses que les compatibilités.

c) Principe de l'analyse

Lors du découpage du mot en proclitique+base1+enclitique, le processus identifie le plus long proclitique (respectivement enclitique) du mot, puis il accède à la table pour vérifier la compatibilité entre les deux (proclitique et enclitique).

Si c'est compatible la décomposition est acceptée, elle sera stockée dans la table du résultat de cette phase, puis on continue avec une nouvelle décomposition pour traiter tous les cas possibles.

Sinon la décomposition est fausse, on passe à une autre décomposition.

Exemple

La décomposition du mot '**أستخرجانها**'d'après le processus précédent donne :

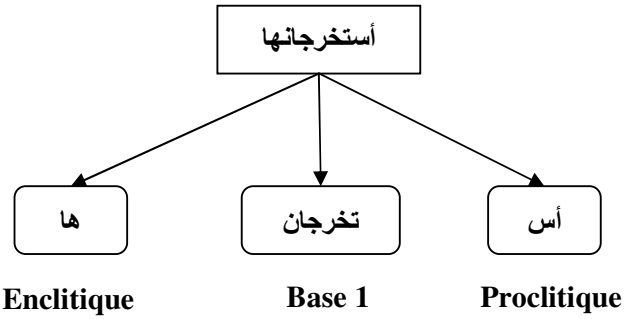


Figure 4.8 : Décomposition en proclitiques, enclitiques.

qui est une décomposition correcte.

Malheureusement, ce n'est pas le cas pour tout le lexique arabe, du fait qu'un verbe peut renfermer un radical (lettre de sa racine) comme un proclitique (respectivement un enclitique).

Exemple

La décomposition du mot 'فسمعهم' donne:

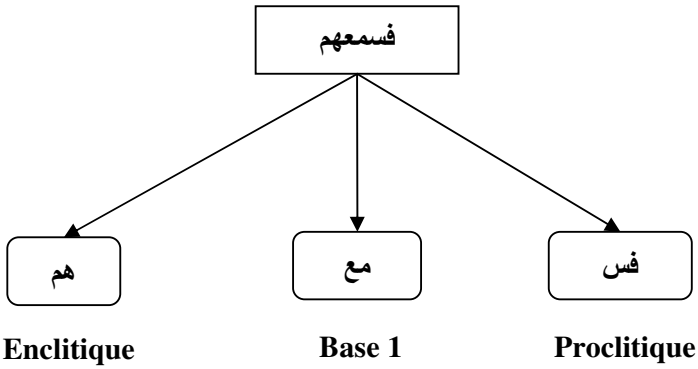


Figure 4.9 : Décomposition erronée.

C'est une décomposition fausse bien que le mot 'مع' existe en arabe, car la racine de 'سمعهم' est 'سمع' et (non pas 'مع'). Le problème qui s'est posé dans ce cas est dû au fait que 'س' est un radical de 'سمع' et un proclitique en même temps. La prise en compte de tous les cas envisageables de décomposition garantit la rencontre de la racine exacte du mot.

Reconnaissance des préfixes et des suffixes

a) Notion de préfixe / suffixe

Comme les enclises, les affixes sont concaténés au début (préfixes) ou à la fin (suffixes) du mot.

Les grammairiens arabes catégorisent les *préfixes* par les lettres qui sont ajoutées aux verbes pour exprimer l'inaccompli c'est-à-dire les lettres assemblées dans le mnémonique «انيت » (حروف المضارعة).

Les *suffixes* sont les lettres qui donnent une information sur le genre comme le «تاء » dans «الكريمة » (la généreuse) et le nombre «ان », «ات » dans «المجلسان », «الكريمات » et les pronoms affixes sujets comme le «تم » dans «ضربتم ».

Pour les préfixes et les suffixes, il existe aussi une certaine compatibilité qui va être illustrée dans une table.

b) Principe de l'analyse

Le principe de cette étape est pratiquement le même que la précédente sauf que la table de compatibilité utilisée est celle des préfixes et des suffixes ainsi que les lettres constituant les préfixes et les suffixes sont présenté dans la table suivante:

Table 4.4 : liste des préfixes et des suffixes.

préfixes	ا	ت	ن	ي	إ	' '											
suffixes	' '	ات	ية	ة	يات	نا	ت	تما	تم	تن	ن	ين	ان	ون	وا	ا	ي

Table 4.5 : table de compatibilité préfixes / suffixes.

P\S	' '	ات	ية	ة	يات	نا	ت	تما	تم	تن	ن	ين	ان	ون	وا	ا	ي
' '																	
ا		o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o
ت		o	o	o	o	o	o	o	o	o							
ي		o	o	o	o	o	o	o	o	o		o					o
ن		o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o
إ		o	o	o	o	o	o	o	o	o		o	o	o			

Exemple

La décomposition de la base 1 '**تخرجان**'d'après le processus précédent donne :

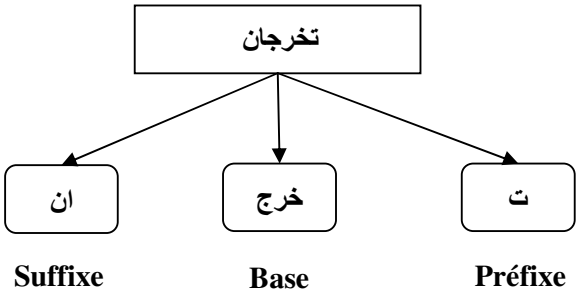


Figure 4.10 : Décomposition en préfixes suffixes.

Exemple

La décomposition du mot 'فأعلن' d'après le processus de décomposition en préfixe+ base +suffixe et celui de proclitique+ base1+enclitique donne :

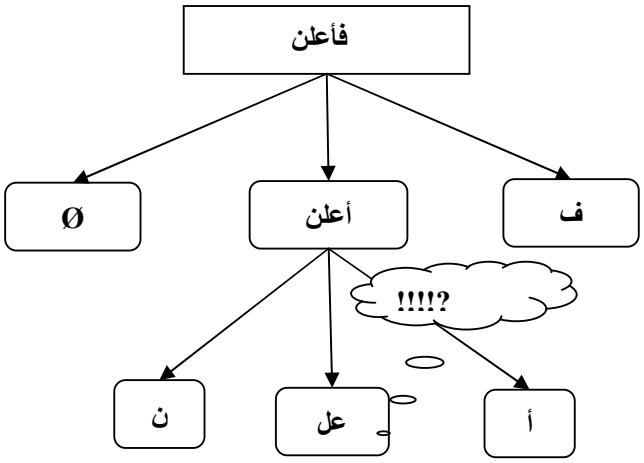


Figure 4.11 : Décomposition complète. erronée.

La décomposition en *proclitique+base1+enclitique* est bonne, mais celle en *préfixe+ base +suffixe* ne l'est pas.

La solution proposée pour ce problème, comme dans la phase précédente, est de traiter tous les cas possibles ; c'est-à-dire, pour un seul mot on doit engendrer tout un tableau de décompositions et traiter les décompositions une par une jusqu'à ce qu'on arrive à une base correcte (racine unique).

Exemple

Découpage du mot'فسأعلنه', la table suivante donne toutes les décompositions possibles, la base est à l'intérieur des cases et les cases vides représentent des cas interdits :

	Pré(‘’), suffi(‘’)	Pré(‘ا’), suffi(‘’)	Pré(‘’), suffi(‘ن’)	Pré(‘ا’), suffi(‘ن’)
Proc(‘’), encl(‘’)				
Proc(‘فس’) encl(‘’)	أعلنه	علنه		
Proc(‘ف’) encl(‘’)	سأعلنه			
Proc(‘’) Encl('ه’)	فسأعلن		فسأعل	
Proc(‘ف’) encl(‘ه’)	سأعلن		سأعل	
Proc(‘فس’) encl(‘ه’)	أعلن	**علن**	أعل	عل

Figure 4.12 : Découpages d’un mot.

De toutes les décompositions précédentes seule ‘علن’ est correcte puisque toutes les autres décompositions seront refusées dans la troisième partie (recherche de la racine et du schème), soit par comparaison avec le dictionnaire des mots outils et spécifiques, soit pour un schème existant avec racine introuvable (sans entrée dans le dictionnaire).

Certains cas de la décomposition donnent plus d’un mot comme indiqué précédemment, et comme dans cet exemple :

Découpage du mot ‘فسأحسنه’

Découpage1= [proc (‘فس’) ; encl (‘ه’) ; pré (‘ أ’) ; suffi (‘ ’) ; base ('حسن’)].
Découpage2= [proc (‘فس’) ; encl (‘ه’) ; pré (‘أ’) ; suffi (‘ ن’) ; base (‘حس’)]

Les deux bases (racine +schème) sont a priori correctes. Pour quelqu’un qui connaît la morphologie de l’arabe, la racine correcte est ‘حسن’ et non pas ‘حس’, mais pour un analyseur automatique les deux sont correctes.

Ceci implique que le processus puisse donner plusieurs racines, la solution est qu'une seule racine doit se retrouver dans le dictionnaire.

Recherche de schème et de la racine

Méthode de recherche de schème

Le principe de la méthode de recherche est très simple. Pour un mot X, un schème i du dictionnaire des schèmes correspond au mot X si la taille du schème est égale à la taille de mot X et si toutes les lettres correspondantes aux positions dans le champ listeinfixe se trouvent dans le mot X aux mêmes positions révélées par ce champ (listeinfixe). (figure 4.13)

Voici un exemple qui nous aide à comprendre le processus :

Mot = ‘صالح’. Le processus de recherche de schème parcourt tous les enregistrements qui ont la même taille que le mot jusqu’à rencontrer le schème ‘فاعل’. Le champ listeinfixe

correspondant est ‘2’ la lettre ‘ا’ se trouve à la position 2 du mot ‘صـالح’ donc c’est probablement le bon schème.

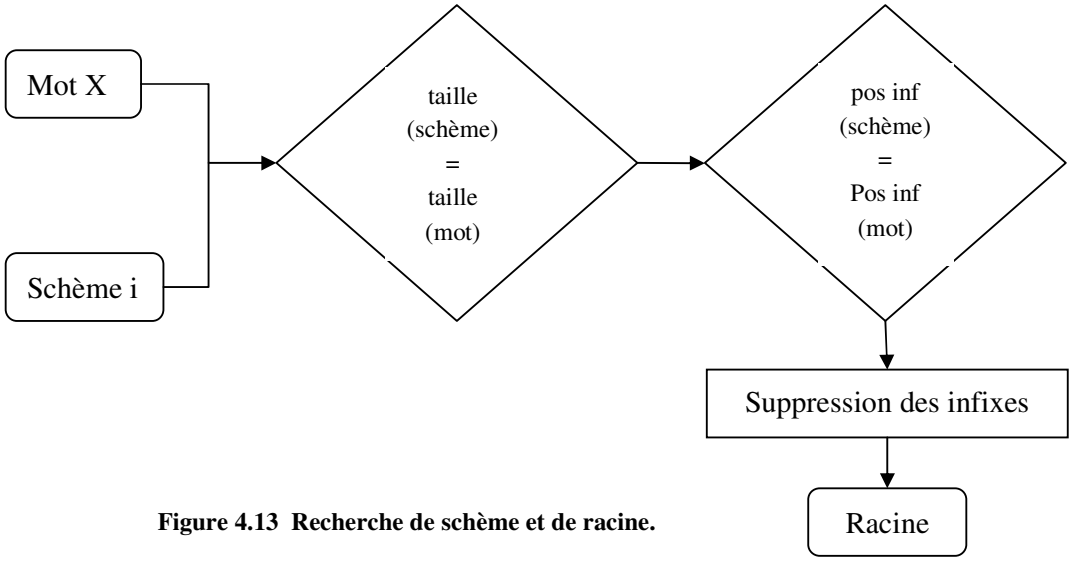


Figure 4.13 Recherche de schème et de racine.

Recherche de la racine

Après la détermination du schème, l’extraction de la racine se limite à la suppression de toutes les lettres correspondantes aux positions de champs listeinfixe dans le mot à décomposer.

Exemple

Le mot ‘صالح’ a pour schème ‘فاعل’, le champ listeinfixe est ‘2’.

L’élimination de la lettre ‘ا’ de la position 2 du mot ‘صـالح’ qui est la même du champ listeinfixe donne ‘صلح’. Ainsi on a retrouvé la racine correcte du mot ‘صـالح’ qui est la racine ‘صلح’ (figure 4.14).

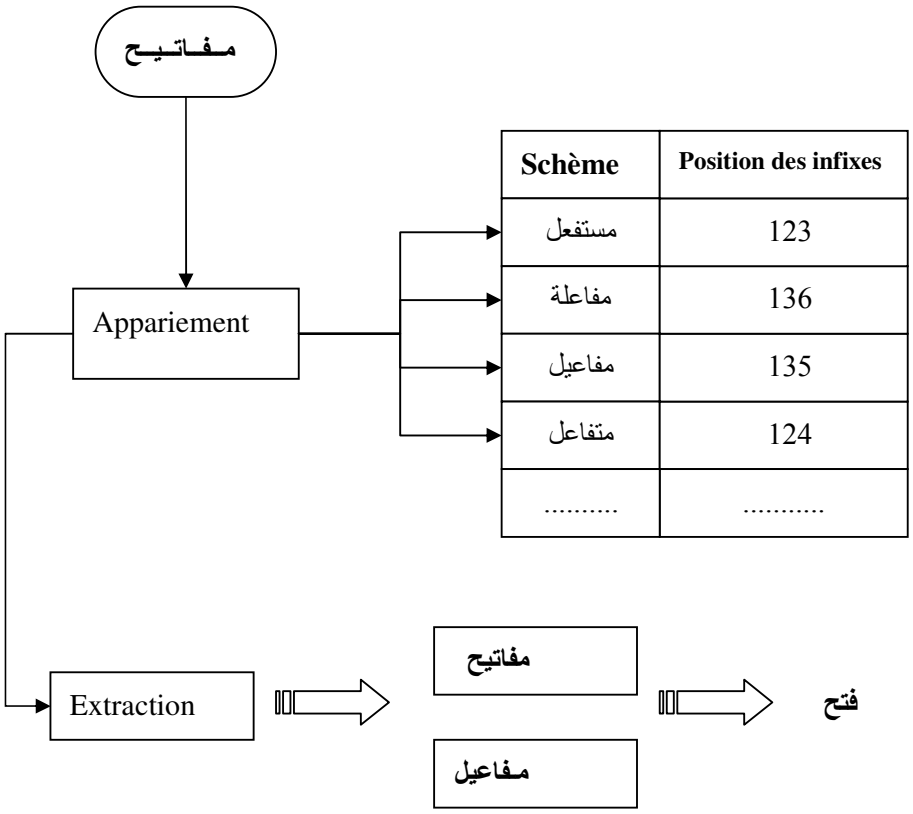


Figure 4.14 Recherche de schème et de racine.

Interprétation

Après extraction de la racine et recherche lexicale, on procède à la génération des valeurs morphologiques du mot, ces informations seront utiles dans le module d'analyse syntaxique qui sert a extraire les syntagmes pertinents.

Exemple

(مكتـوب) à pour schème (مفعـول) ; dans le champ catégorie du schème (مفعـول) on trouve la catégorie (nom), on affecte cette catégorie au mot (مكتوب).

Remarque

Ces quatre étapes ne sont pas les seules, et des sous-étapes peuvent être introduites en cas de besoin (traitement de compatibilité entre les proclitiques et enclitiques et préfixes et suffixes, désambiguïsation dans le cas où il y a plusieurs interprétations, extraction de la racine).

Ces valeurs morphologiques sont extraites du dictionnaire des schèmes qui fait correspondre à chaque schème une catégorie grammaticale.

Exemple

مفاعلة: nom

انفعل: verbe

فاعل: verbe, nom.

4.3. Indexation

4.3.1 Définition

Il s'agit ici d'indexation automatique (non supervisée) : le programme indexe les documents sans intervention humaine. Le processus d'indexation effectue le transfert de l'information contenue dans le texte d'un document vers un autre espace de représentation traitable par un système informatique [64].

Le principal enjeu de la catégorisation de texte, par rapport à un processus d'apprentissage classique, réside dans la recherche des descripteurs (ou termes) les plus pertinents pour le problème à traiter [100]. Il existe différentes méthodes proposées pour le choix des descripteurs et des poids associés à ces descripteurs [101]. [72, 100] utilisent, à titre d'exemples, les mots comme descripteurs, tandis que d'autres préfèrent utiliser les lemmes (racines lexicales) [102] ; ou encore des stems (la suppression d'affixes) [103]. Il existe une autre approche de la représentation des textes : les n-grammes[30] [104].

Les descripteurs donc peuvent être les mots du texte, les lemmes, les concepts (mots-clé ou termes) et, plus rarement, les N-grammes, ou encore les contextes (cas du "Latent Semantic Indexing" ou des méthodes basées sur l'Analyse Factorielle des Correspondances). Les modèles utilisant les mots peuvent fonctionner avec la langue anglaise (peu de flexions, peu d'homographies), mais se révèlent nettement insuffisants pour les autres langues (particulièrement pour les langues agglutinantes). On peut alors utiliser les lemmes, mais, pour avoir de bonnes performances, il faut recourir à une analyse linguistique pour lever certaines ambiguïtés [62]. L'indexation par des pseudo-racines donne de meilleurs résultats pour le modèle vectoriel, ce que confirme [58].

[30] Les n –grammes sont plus utilisés pour la classification des documents que pour la recherche d'information, voir [65].

4.3.2 Objectif

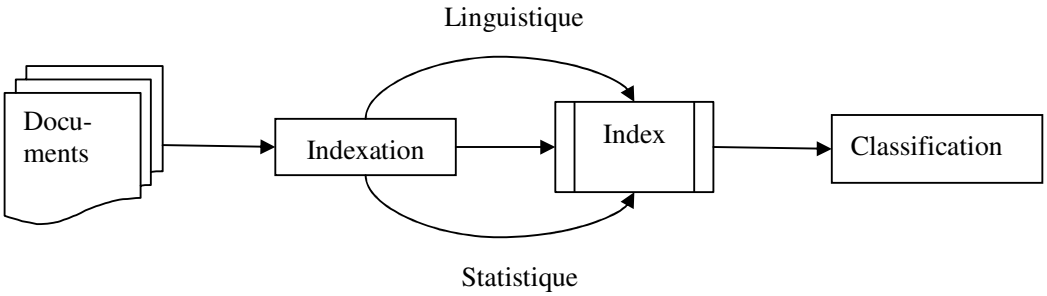


Figure 4.15 Processus de classification.

A partir d'une collection de documents, le processus d'indexation revoie une liste d'index structurés. Ce résultat, est utilisé le plus souvent, pour effectuer des recherches d'information. Mais, il peut également servir à comparer et classifier des documents (figure4.15), proposer des mots clés, faire une synthèse automatique de documents, calculer des co-occurrences de termes…

L'objectif du niveau morphologique est de diminuer fortement le nombre de mots analysés (par lemmatisation). La dimension d'un index peut être réduite d'au moins 50% quand les lemmes de mots sont utilisés au lieu des formes fléchies des mots [72].

En plus on élimine les mots vides en utilisant un anti-dictionnaire, pour alléger encore les index.

4.3.3 Application

L'analyse thématique des textes relève d'un effort d'abstraction se situant au niveau macro-structurel, et est régie par quatre principes :

- la suppression de l'information non pertinente,
- la sélection de l'information pertinente,
- la généralisation des propositions retenues et
- l'intégration des propositions dans un tout structuré et cohérent [109].

Exemple

(مما صح عنه عليه الصلاة و السلام قوله: " عليكم بالصدق، فان الصدق يهدي إلى البر. و إن البر يهدي إلى التقوى. و لا يزال الرجل يصدق و يتحرى الصدق، حتى يكتب عند الله صديقا. و إياكم و الكذب، فان الكذب يهدي إلى الفجور. و إن الفجور يهدي إلى النار. و لا يزال الرجل يكذب و يتحرى الكذب، حتى يكتب عند الله كذابا ". في هذا الحديث

الشريف، دعوة قوية من النبي صلى الله عليه و سلم إلى إتباع الصدق و التحلي به قولا و فعلا. بل على الرجل الفطن الكيس أن يفتش و ينقب و يتحرى أثاره في كل زمان و مكان. لأن ذلك يهديه و يقوده إلى البر و الخير، فان فعل ذلك كان مع الصديقين و الشهداء و حسن أولئك رفيقا أي جوارا و صحبة في الجنة. و من معاني الصدق كذلك التصديق بكل ما أمر الله تعالى الإيمان به. و من معانيه كذلك الصدقة و الإنفاق في سبل و أوجه الخير الكثيرة، مصداقا لقوله تعالى " إنما الصدقات للفقراء و المساكين ".)

Voici le texte après découpage en mots :

Table 4.6. Découpage du Texte en Mots.

مما	إن	الله	النار	الحديث	به	في	ذلك	و	كذلك	و
صح	البر	صديقا	و	الشريف	قولا	كل	كان	من	الصدقة	المساكين
عنه	يهدي	و	لا	دعوة	و	زمان	مع	معاني	و	
عليه	إلى	إياكم	يزال	قوية	فعلا	و	الصديقين	الصدق	الإنفاق	
الصلاة	التقوى	و	الرجل	من	بل	مكان	و	كذلك	في	
و	و	الكذب	يكذب	النبي	على	لأن	الشهداء	التصديق	سبل	
السلام	لا	فان	و	صلى	الرجل	ذلك	و	بكل	و	
قوله	يزال	الكذب	يتحرى	الله	الفطن	يهديه	حسن	ما	أوجه	
عليكم	الرجل	يهدي	الكذب	عليه	الكيس	و	أولئك	أمر	الخير	
بالصدق	يصدق	إلى	حتى	و	أن	يقوده	رفيقا	الله	الكثيرة	
فان	و	الفجور	يكتب	سلم	يفتش	إلى	أي	تعالى	مصداقا	
الصدق	يتحرى	و	عند	إلى	و	البر	جوارا	الإيمان	لقوله	
يهدي	الصدق	إن	الله	إتباع	ينقب	و	و	به	تعالى	
إلى	حتى	الفجور	كذابا	الصدق	و	الخير	صحبة	و	إنما	
البر	يكتب	يهدي	في	و	يتحرى	فان	في	من	الصدقات	
و	عند	إلى	هذا	التحلي	أثاره	فعل	الجنة	معانيه	للفقراء	

Total: 162 mots

Pour chaque mot reconnu, on le compare avec les éléments de l'antidictionnaire ; si ce mot en fait partie, il ne sera pris en considération ni dans l'index, ni dans le calcul de fréquences.

Le tableau suivant présente l'index après l'élimination des mots vides:

Table 4.7. Suppression des Mots Vides.

بالصدق	يصدق	يهدي	الكذب	الصدق	أثاره	الصديقين	التصديق	الخير
الصدق	يتحرى	الفجور	يكتب	التحلي	زمان	الشهداء	الله	الكثيرة
يهدي	الصدق	الفجور	الله	قولا	مكان	حسن	الإيمان	الصدقات
البر	يكتب	يهدي	كذابا	فعلا	يهديه	رفيقا	معانيه	للفقراء
البر	الله	النار	دعوة	الرجل	يقوده	جوارا	الصدقة	المساكين
يهدي	صديقا	الرجل	النبي	يفتش	البر	صحبة	الإنفاق	قوية
التقوى	الكذب	يكذب	الله	ينقب	الخير	الجنة	أوجه	الفطن
الرجل	الكذب	يتحرى	إتباع	يتحرى	فعل	الصدق	سبل	الكيس

Total: 72 mots

Remarque

A la suppression des mots vides du domaine certains mots ne peuvent être éliminées qu'après l'intervention des règles syntaxiques contextuelles qui déterminent l'adjacence des constituants.

L'indexation par des pseudo-racines donne de meilleurs résultats pour le modèle vectoriel, ce que confirme [58].

Le tableau suivant présente l'index après l'opération de segmentation des mots – élimination des proclitiques/enclitiques et des préfixes/suffixes – qui permet de relier des termes qui ne se différencient que par une marque flexionnelle [179] :

Les mots (كتب،كتبت،أكتب،تكتب،كتبتما،يكتبون،كتبتن،يكتب،...) sont considérés comme des descripteurs différents alors qu'il s'agit de la même base, la *lemmatisation* cherche à résoudre cette difficulté.

Il est inutile qu'un document contienne les mêmes formes fléchies des termes que celles utilisées dans un autre document pour qu'ils soient reconnus similaires [18].

Table 4.8. Index Morphologique.

فطن	إنفاق	شهداء	أثار	حلي	جنة	صديق	رجل	صدق
كيس	أوجه	حسن	زمان	قول	إيمان	كذب	حرى	هدي
كثير	سبل	رفيق	مكان	فعل	دعو	فجور	كتب	بر
	فقراء	جوار	قاد	فتش	نبي	نار	اله	تقوى
	مساكين	صحب	خير	نقب	إتباع	كذاب	تصديق	قوي

Total: 43 mots

Le tableau suivant présente l'index allégé par des règles dérivationelles qui peuvent éventuellement être appliquées pour diminuer davantage le nombre de formes à traiter en rapprochant des mots qu' on suppose sémantiquement proches, étant dérivés d'une même source originale.

Table 4.9 Index Morphologique Poussé à l'Extraction de la Racine.

قوي	نفق	صحب	شهد	زمن	فعل	نبي	كذب	رجل	صدق
فطن	وجه	جنة	حسن	مكن	فتش	تبع	فجر	حرى	هدي
كيس	فقر	كثر	رفق	قاد	نقب	حلي	نار	كتب	بر
سبل	سكن	امن	جور	خير	أثر	قول	دعو	اله	تقو

Total: 40 mots

Remarque

La dernière étape diminue davantage la taille de l'index, puisque elle relie des termes supposés proches (étant dérivés d'une même racine), mais ceci entraîne une grande perte d'information et rapproche des documents différents dans la phase de classification.

Exemple

Les deux mots الحروقات،المحروقات sont regroupé sous la même racine حرق , et donc sous la même catégorie pourtant il est clair qu'il s'agit de deux sujets distincts :

Les *brûlures* ≠ le *pétrole*.

La solution est d'arrêter au niveau flexionnel et exploiter le résultat du niveau dérivationnel dans d'autres tâches.

A la sortie de ce niveau, le texte est représenté par un index de mots lemmatisés (chacun des mots a été ramené à sa forme canonique).

Dans le cas d'ambiguïté, soit le système fournit un seul lemme avec une seule catégorie, ou il fournit toutes les possibilités et la levée de l'ambiguïté est effectuée, soit à l'étape morphologique par des règles micro-syntaxiques, soit à l'étape syntaxique par des règles contextuelles syntaxiques.

Il ne s'agit pas ici de respecter au sens strict les modèles morphologiques proposés pour la langue : la forme généralement retenue pour représenter ces éléments est le nom, la grande majorité des travaux en intelligence artificielle est fondée sur l'extraction des groupes nominaux : "ce choix se justifie en pratique par le fait qu'en langue spécialisée, l'information pertinente est localisée dans les groupes nominaux et la fréquence de ces groupes nominaux demeure en langue spécialisée importante et élevée " [63] c'est pour cette raison qu'on a besoin de faire suivre l'analyse morphologique par une analyse syntaxique.

Remarque

La diminution de la taille de l'index est relative au texte ; si le texte contient beaucoup de termes dérivés d'une même racine, la réduction tend vers 100% ; et si le texte ne contient jamais deux mots communs à une racine ou même base, la réduction sera de l'ordre de 0%.

Texte1 : كتب الكاتب في مكتبه بالكاتبة كل الكتب غير المكتوبة في المكتبة

Index1 : كتب

Texte2 : تقدم الدولة حاليا تحفيزات للنهوض بالبحث العلمي في الجزائر

Index2 : قدم، دول، حفز، نهض، بحث، علم، الجزائر

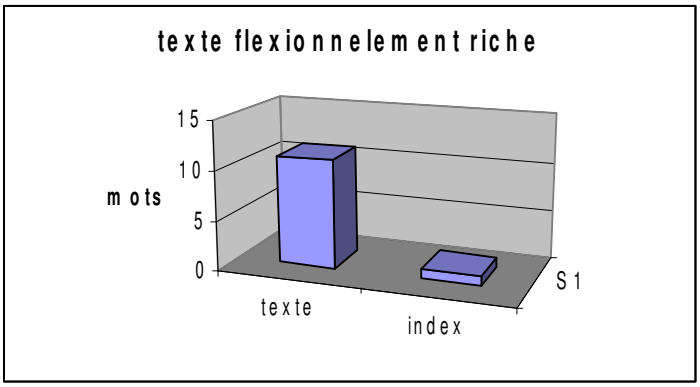


Figure 4.16 Texte1 vs Index1.

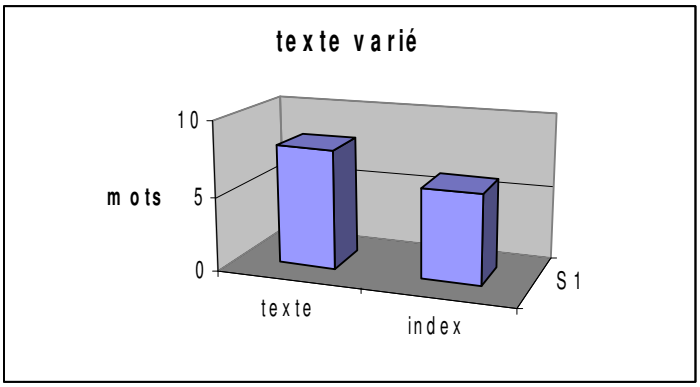


Figure 4.17 Texte2 vs Index2.

4.4. Analyse syntaxique

Cette phase de traitement est essentielle pour la suite du travail, car elle consiste à réduire la taille de l'index.

L'analyse syntaxique automatique des langues naturelles est une étape fondamentale dans le processus d'analyse automatique de la langue, puisque c'est à elle qu'incombe la tâche cruciale de déterminer les structures syntaxiques des phrases.

Ce sont ces structures, en effet, qui vont permettre de calculer les diverses interprétations sémantiques et pragmatiques ainsi que l'extraction des syntagmes.

La méthode d'analyse utilise les informations morphologiques hors contexte pour faire la catégorisation des différents constituants [46]. Nous nous intéressons aux syntagmes nominaux SN(s) au niveau syntagmatique de l'analyse linguistique sans prendre en considération les niveaux sémantiques et pragmatiques.

La syntaxe de la plupart des langues naturelles est engendrée par des règles de grammaire. Dans la langue arabe, les informations concernant la position du constituant dans la phrase (le préfixe, le suffixe et l'infixe) déterminent la catégorie du mot dans la phrase

4.4.1 Les composants de l'analyseur syntaxique

Un analyseur syntaxique peut être considéré comme un mécanisme qui assigne à un texte d'entrée un ensemble de représentations formelles, par exemple, des structures de constituants comportant toutes les informations grammaticales et lexicales relatives à la phrase d'entrée.

On distingue habituellement deux composantes dans un analyseur : une *déclarative* et l'autre *procédurale*.

La composante *déclarative* correspond aux connaissances linguistiques et la composante *procédurale*, est la stratégie d'analyse. Les connaissances linguistiques comprennent la grammaire de la langue, alors que la stratégie d'analyse est un algorithme qui spécifie en détail chacune des opérations impliquées dans le processus d'analyse

La composante déclarative

Elle comprend les connaissances linguistiques nécessaires au processus d'analyse, ces connaissances correspondent à un ensemble de règles de la grammaire qui détermine la structure des phrases.

La composante procédurale

Stratégie utilisée

La stratégie correspond à la façon dont l'analyseur utilise les connaissances linguistiques à sa disposition pour déterminer quelles structures peuvent être assignées aux constituants de la phrase qui lui est soumise.

Notre analyseur syntaxique va décider si deux termes successifs forment ou non un constituant d'un niveau donnée mais il ne vérifiera pas si la phrase proposée est acceptée ou non par la grammaire qui engendre le langage en question : sa stratégie très locale est donc ascendante.

4.4.2 Les fonctions de l'analyseur

Le premier objectif de ce niveau est la désambiguïsation syntaxique (l'étiquetage), et le deuxième est l'extraction des syntagmes.

Désambiguïsation syntaxique

Elle vise à associer à chaque mot, en contexte, une étiquette syntaxique. Cette étiquette indique la catégorie grammaticale (syntaxique) du mot.

Exemple

(لقد كاتب الزبون إدارة الشركة)

Le mot (كاتب) est ambiguë, il est soit le verbe (كَاتَبَ) (il a écrit à quel qu'un) soit le nom (كَاتِبٌ) (écrivain). Le module de désambiguïsation va intervenir pour décider de la catégorie correspondante dans le contexte : ce mot est en fait un verbe car il est précédé par une particule verbale.

Extraction des syntagmes

Le deuxième objectif de l'analyse syntaxique est l'extraction des syntagmes nominaux, pour les ajouter à l'index. Ainsi "الذكاء الاصطناعي" la phrase :

«يعتبر الذكاء الاصطناعي أهم ميدان للبحث في الإعلام الآلي »

pourra être indexée dans une seule entrée et pas séparément, cela augmente la précision des index.

Ainsi, à partir de la description des diverses structures possibles de GN (et il existe un nombre abondant d'études linguistiques portant sur ce type de syntagmes), et des structures impossibles, le système repère les séquences de constituants formant des Groupes Nominaux en éliminant les séquences impossibles.

Les recouvrements entre les groupes de mots peuvent, sous certaines conditions, améliorer la qualité des attributs terminaux pour la classification de documents. [105]. Les différentes compositions possibles pour les noms sont :

- un nom indéfini suivi d'un nom défini,
- un nom suivi d'un ou de plusieurs adjectifs qui sont en accord pour la définition / indéfinition,
- un nom défini suivi d'un nom indéfini ou phrase.

L'indexation se déroule alors de la manière suivante. Après la reconnaissance des mots et leur étiquetage, et le repérage des locutions indicatrices, les ambiguïtés dues à l'appartenance

polycatégorielle étaient levées, en fonction des cas, par application de règles locales et par le choix de l'appartenance la plus probable. Puis les formes syntaxiques indicatrices du figement étaient recherchées systématiquement.

Exemple رئيس الجامعة

«رئيس (Nom_I) الجامعة (Nom_D) » correspond à la forme « Nom_I Nom_D », il sera alors indexé en une seule entrée et les deux entrées (رئيس،الجامعة) sont supprimée de l'index.

Remarque

L'adjacence de deux constituants définies ou indéfinités ne signifie pas toujours un mot composé comme montre l'exemple suivant

Exemple شجع الأستاذ الطالب

Le système engendre le mot composé (الأستاذ الطالب) qui est erroné

La solution est de consulter l'attribut du verbe s'il s'agit d'un verbe transitif ou intransitif :

- pour la phrase شجع الأستاذ الطالب pas de mots composés.
- et pour la phrase دخل المدير الجديد le mot composé est (المدير الجديد) car le verbe (دخل) est intransitif.

4.4.3 Indexation

Au vu des nombreux travaux effectués, l'exploitation de ces structures semble intéressante pour une description plus riche du contenu. Cependant, une analyse syntaxique partielle apparaît suffisante pour l'extraction et la prise en compte des termes complexes et structurels [70]. Une fois que la collection est étiquetée le système extrait l'ensemble des syntagmes nominaux et l'utilise pour l'indexation [91].

Malgré la simplicité de l'utilisation de mots comme unité de représentation, certains auteurs proposent plutôt d'utiliser les phrases comme unité [92, 93, 94]. Les locutions nominales sont plus informatives que les mots seuls, par exemple : « *machine learning* » ou « *world wide web* » car les locutions ont l'avantage de conserver l'information relative à la position du mot dans la phrase (*the problem of compositional semantics* : homme pauvre ≠ pauvre homme).

Remarque

La lexicalisation d'une locution déroge à la composition sémantique. Le français peut utiliser le trait d'union en cas d'ambiguité : belle mère ≠ belle-mère ; œil de bœuf (œil d'un bœuf) ≠ œil-de-bœuf (fenêtre ovale). Le français utilise plus de locutions lexicalisées que l'allemand ou le turc pour compenser la faiblesse des mécanismes de composition de mots : ainsi, *machine à laver le linge* (devenu *lave-linge*) ou *chemin de fer métropolitain* devenu *métro*. Mais *Einfachlanggleitkommazahlen* pour « *nombre virgule flottante simple précision* » ou *Elektronischedatenverarbeitung* = *Informatik* pour *Traitement électronique des données* = *Informatique*. Ainsi, une locution longue peut marquer un glossème (au sens de Hjelmslev), et n'être qu'un mot dans une autre langue (ou un autre état de langue, selon le principe d'économie).

La prise en compte des mots composés dans l'index réduit davantage la taille de l'index. Continuant avec l'exemple précèdent, le tableau suivant présente l'index après l'analyse syntaxique.

Table 4.10 Index syntaxique.

تصديق	شهداء	أثار	حلي	حرى	صديق	**دعوة قوية**	هدي
أمر	حسن	زمان	قول	كتب	كذب	**الرجل الفطن الكيس**	بر
إيمان	رفيق	مكان	فعل	اله	فجور	**إتباع الصدق**	تقوى
	جوار	قاد	فتش	إنفاق	نار	**أوجه الخيرالكثيرة**	سبل
	صحب	جنة	نقب	مساكين	كذاب	فقراء	نبي

Total: 38 mots

Remarque

Les techniques linguistiques pour l'extraction des syntagmes, pour l'instant n'ont pas donné de résultats spectaculaires [106]

L'inclusion des locutions pendant l'indexation augmenterait la précision des systèmes de recherche d'informations mais pas celle de la catégorisation [107]

[108] a trouvé que le regroupement des mots n'a pas amélioré l'exactitude de la classification.

Une des difficultés d'intégration de l'analyse syntaxique dans les systèmes de classification est leur pondération. En effet les mesures traditionnelles (tf ou idf) sont inadaptées car les mots composés sont moins fréquents et donc sous-pondérés. Or une mauvaise pondération

de ces termes peut devenir néfaste au processus de classification. *Une solution consisterait à les intégrer dans le modèle vectoriel de manière indépendante par rapport au terme simple*. On obtiendrait alors deux sous-vecteurs permettant ainsi un meilleur classement des documents [70].

Logiquement, une telle représentation doit obtenir de meilleurs résultats que ceux obtenus via les mots. Mais les expériences ne sont pas concluantes car, si les qualités sémantiques sont conservées, les qualités statistiques sont largement dégradées : le grand nombre de combinaisons possibles entraîne des fréquences faibles et trop aléatoires [110].

La reconnaissance des « mots composés » est une opération qui peut améliorer les performances de l'indexation, mais ce n'est évidemment pas la seule. Rappelons qu'il ne s'agit pas d'utiliser une grammaire complète de l'arabe, mais plutôt d'identifier des structures locales, relatives à un certain nombre de phénomènes, fixant les conditions d'emploi et d'interprétation des unités lexicales dans un contexte donné, afin de repérer les éléments significatifs du contenu de documents.

4.5. Calcul de fréquences

Le domaine de la recherche d'informations depuis le début de l'informatisation des fonds documentaires jusqu'au développement actuel de la toile a vu apparaître de nombreuses techniques pour accéder à l'information. Une des premières idées, fondatrices du domaine et formulée par Luhn [111] en 1958, propose d'utiliser la fréquence des mots ainsi que leur position relative pour mesurer la pertinence d'un article par rapport à un besoin d'informations. A ce jour, beaucoup de modèles et de systèmes fondés sur l'idée de fréquence ont été développés, cependant, peu d'approches sont basées sur la proximité entre les termes pour détecter les documents pertinents [112].

Les algorithmes d'apprentissage ne sont pas capables de traiter directement les textes ni, plus généralement, les données non structurées comme les images, les sons et les séquences vidéos. C'est pourquoi une étape préliminaire dite de représentation est nécessaire. Cette étape consiste généralement en la représentation de chaque document par un vecteur, dont les composantes sont par exemple les mots contenus dans le texte, afin de le rendre exploitable par les algorithmes d'apprentissage [72]. Une collection de textes peut être ainsi représentée par une matrice dont les lignes sont les termes qui apparaissent au moins une fois et les colonnes sont les documents de cette collection. L'entrée w_{kj} est le poids du terme t_k dans le document d_j.

La pondération des termes vise à avantager les termes qui peuvent le mieux différencier les documents entre eux [81].

Elle permet d'une part de déterminer l'importance relative des termes et d'autre part de quantifier leur importance dans la description du contenu sémantique du document. [70] Elle est également utilisée pour filtrer l'index résultant du processus d'indexation (éliminer les index dont le poids est inférieur à un certain seuil). Il existe plusieurs techniques de pondération des termes dont les quatre les plus importantes sont décrites ci-après.

4.5.1 Fréquence d'occurrences

On admet généralement qu'un mot qui apparaît fréquemment dans un texte représente un concept important. La représentation est une extension naturelle de la représentation binaire (attribution de '1' pour l'existence et '0' pour l'absence) qui prend en compte le nombre d'apparitions d'un mot dans un document. Ainsi, un document est représenté par un vecteur ou suite indexée par les éléments du vocabulaire ordonné V, chaque composante correspondant au nombre d'apparitions dans le document du vocable de même rang dans V [66]. De manière formelle, le document d est représenté par un vecteur f_d tel que :

$$\forall v \in V,\ f_d[v] = \text{nombre d'apparitions du terme } v \text{ dans d.}$$

On répète certains mots naturellement pour décrire un phénomène, proposer une thèse, insister sur un point ou comme tic langagier. La fréquence de répétition indique-t-elle la signification du mot dans le document ?

G.K. Zipf a postulé que, par principe de plus petit effort, nous utilisons quelques mots très souvent et rarement la plupart des autres mots [77]. Ainsi, un petit pourcentage de mots du vocabulaire (<20%) compte pour une majorité de mots dans un texte. La loi de Zipf dit que, si on classe par ordre décroissant de fréquence les termes d'un long document, la fréquence du terme de n-ième rang varie en 1/n (loi affinée par Mandelbrot). Cette loi empirique se vérifie dans les textes comme dans les conversations téléphoniques.

Les termes les plus informatifs d'un corpus de documents ne sont pas les mots qui apparaissent le plus dans le corpus car ceux-ci sont pour la plupart des mots-outils, ni les moins fréquents du corpus qui peuvent par exemple être issus de fautes d'orthographe ou de l'utilisation d'un vocabulaire trop spécifique à quelques documents du corpus. Par contre, un mot qui, sans être banal, apparaît beaucoup dans un document, possède certainement une information forte sur la sémantique du document. C'est la conjecture de Luhn, relative à une relation entre la fréquence des termes et leur pouvoir évocateur, i.e. leur importance dans la

participation au sens d'un texte : un mot est important s'il n'est ni trop fréquent ni trop rare [70, 111].

On peut donc définir une fenêtre : on considère un mot comme représentatif si sa fréquence est entre un seuil minimal et un seuil maximal, ce qui correspond à l'informativité d'un mot, qui mesure la quantité du sens qu'un mot porte. La correspondance entre l'informativité et la fréquence d'apparition des mots est illustrée par la figure suivante

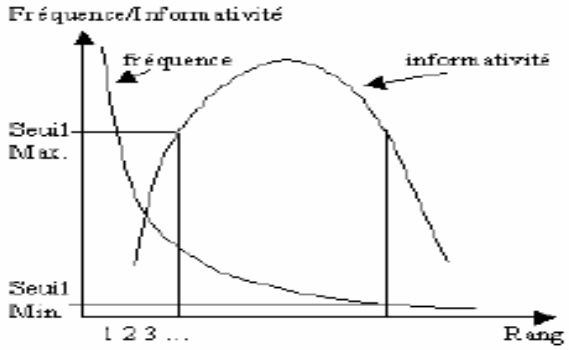


Figure 4.18 Correspondance entre l'informativité et la fréquence [67].

4.5.2 La valeur de discrimination

Par discrimination, on se réfère au fait qu'un terme distingue bien un document des autres documents. Un terme qui a une valeur de discrimination élevée doit apparaître seulement pour un petit nombre de documents. Un terme qui apparaît dans tous les documents n'est pas discriminant. Le pouvoir de discrimination d'un terme est important dans le choix des index qu'on veut garder [67]. En résumé, « un mot est informatif dans un document s'il y est présent souvent sans l'être trop dans les autres documents du corpus » [66].

Dans le modèle vectoriel, chaque document est représenté par un vecteur de poids P_{ij} (le poids du terme T_j dans le document D_i). Etant donné un corpus et un index, on a donc une matrice de correspondance. Pour calculer la valeur de discrimination d'un terme, on doit comparer une sorte d'uniformité au sein du corpus avec celle du corpus transformé dans lequel le terme en question a été uniformisé (mis au même poids). L'idée est que, si en uniformisant le poids d'un terme dans tous les documents, on obtient une grande amélioration dans l'uniformité du corpus alors ce terme est très différent dans différents documents : il a donc une grande valeur de discrimination. En revanche, si en uniformisant le poids du terme, on n'obtient pas beaucoup d'amélioration sur l'uniformité, ce terme est déjà distribué de façon uniforme, donc peu discriminant.

Le calcul de la valeur de discrimination d'un terme T_i se fait comme suit :

1) on calcule d'abord le vecteur centroïde du corpus : pour chaque terme, son poids dans le vecteur centroïde V est la moyenne de ses poids dans les documents. C'est-à-dire, si N est le nombre de documents dans le corpus on aura :

$$V_k = \frac{1}{N}\sum_{j=1}^{N} P_{kj}$$

2) on calcule l'uniformité du corpus comme la similarité moyenne des documents avec le vecteur centroïde :

$$U_1 = C * \sum \ RSV*(d_k,V)$$

où C est une constante de normalisation et RSV (d_i,V) est la similarité entre le document d_i et le vecteur centroïde V

3) on force le poids du terme T_i en question à 0, et on répète les deux étapes ci-dessus pour obtenir une nouvelle valeur d'uniformité U_2.

4) la valeur de discrimination du terme est :

$$\Delta_i = U_2 - U_1$$

Dans ce calcul de la discrimination, on ne se préoccupe pas beaucoup de la fréquence d'un terme dans un document particulier, mais plutôt à sa distribution dans le corpus. [69]

4.5.3 Tf*idf

L'objectif dans toute méthode d'indexation est de trouver les mots importants dans le document et aussi de distinguer le document de tous les autres documents. Dans la méthode par fréquence d'occurrence, la première condition est satisfaite en trouvant ces mots qui paraissent importants dans le document. La deuxième exigence à besoin de connaître comment les mots sont distribués à travers documents [25].

Les scores TFIDF (largement utilisé dans les problèmes de catégorisation [82]), qui représentent l'importance d'un concept dans un document pondérée par la présence de ce concept dans les autres documents, sont une bonne métrique pour déterminer les mots-clé d'un document [75].

Tf*idf désigne un ensemble de schémas de pondération de termes. Tf signifie « terme frequency » et idf « inverted document frequency » :

- Par tf, on désigne une mesure de l'importance d'un terme pour un document. En général, cette valeur est déterminée par la fréquence du terme dans le document.

- Par idf, on mesure si le terme est discriminant (ou non uniformément distribué) [68].

Ce score TFIDF donne une importance au concept en fonction de sa fréquence dans le document, pondérée par la fréquence d'apparition du concept dans le reste du corpus. C'est-à-dire on utilise le score TFIDF comme métrique de l'importance du concept pour la discrimination du document.

On présente ci-dessous quelques formules les plus utilisées de tf et idf.

$$Tf = f(t,d)/max[f(t,d)]$$
$$Tf = log(f(t,d))$$
$$Tf = log(f(t,d)+1)$$

où f(t,d) est la fréquence d'occurrence du terme t dans le document d,

$$Idf = log(N/n)$$

où N est le nombre de documents dans le corpus, et n ceux qui contient le terme t.

Une formule tf*idf est donc le produit d'une tf par une idf. Par exemple :

$$Tf*idf = [f(t,d)/max[f(t,d)]]*log(N/n)$$

Une formule tf*idf combine l'importance du terme pour un document (par tf), et le pouvoir de discrimination de ce terme (par idf). Ainsi, un terme qui a une valeur de tf*idf élevée doit être à la fois important dans ce document, et apparaître peu dans les autres documents. Ce terme est alors une caractéristique importante d'un document.

Mais l'ajout d'un nouveau document dans le système nécessite de recalculer tous les scores TFIDF. Néanmoins, lorsque le nombre de documents est déjà élevé, l'ajout d'un nouveau document ne modifie pas beaucoup les autres scores TFIDF, et la réévaluation peut donc être différée.

Variantes

Le codage TF × IDF ne corrige pas la longueur des documents. Pour ce faire, le codage **TFC** est similaire à celui de TF×IDF mais il corrige les longueurs des textes par la nor-

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

malisation en cosinus, afin de ne pas favoriser les documents les plus longs. [82]

D'autres codages sont également utilisés, comme par exemple le codage LTC [113] qui tente de réduire les effets des différences de fréquences, ou encore le codage à base d'entropie [114] qui donnerait de meilleurs résultats [100].

4.5.4 L'indexation sémantique latente

La méthode LSA est fondée sur le fait que des mots qui apparaissent dans le même contexte sont sémantiquement proches. Le corpus est représenté sous forme matricielle. Les lignes représentent les mots et les colonnes représentent les différents contextes choisis (un document, un paragraphe, une phrase, etc.). Chaque cellule de la matrice représente le nombre d'occurrences des mots dans chacun des contextes du corpus. Deux mots proches au niveau sémantique sont représentés par des vecteurs proches. La mesure de similarité est généralement définie par le cosinus de l'angle entre les deux vecteurs.

La première méthode d'indexation à base de fréquence n'a pas utilisé de rapports globaux dans la collection de documents. Dans la méthode TF*IDF on capte quelques-uns de ces rapports globaux pour améliorer la représentation d'un document dans l'index. LSI (proposée par [115]) est une méthode d'indexation basée sur la décomposition de la valeur singulière (SVD) [76] du mot dans la matrice du document.

L'étude effectuée par [116] a montré de bons résultats de la méthode LSI par rapport au modèle vectoriel, vu que la LSI se distingue par une décomposition en valeurs singulières.

Tandis que les méthodes antérieures ont identifié ces rapports en comptant le nombre de documents dans lesquels un mot se produit, le LSI construit des rapports basés sur les cooccurrences des mots qui se produisent dans de multiples documents. Ces rapports cachés sont appelés la *structure sémantique latente* dans la collection. L'avantage principal de LSI sur les autres méthodes est qu'il ne dépend pas des mots individuels pour localiser des documents, mais plutôt utilise un concept ou sujet pour trouver des documents pertinents [25].

Le document est transformé en espace LSI et comparé avec les autres documents dans le même espace. Les mots qui sont en rapport avec les mots-clé du document basés sur la cooccurrence sont comparés avec les documents qui ont été indexés de la même façon.

4.5.5 Evaluation des méthodes

Nous avons cité quatre méthodes différentes de pondération pour l'indexation des documents.

La première méthode basée sur la fréquence était la plus facile à calculer. On compte simplement le nombre d'occurrences des mots dans les documents. Cette méthode est encline à la manipulation et ne considère pas de rapports globaux.

La deuxième méthode ne se préoccupe pas beaucoup de la fréquence d'un terme dans un document particulier, mais beaucoup plus de sa distribution dans le corpus, c'est-à-dire de

trouver les termes discriminants, ceux qui distinguent bien un document des autres documents.

La méthode IDF ajoute des rapports globaux à la représentation d'un document. Les fréquences individuelles du mot dans un document sont inversement proportionnelles au nombre de documents dans lesquels le mot apparaît. Les mots qui apparaissent dans beaucoup de documents sont pondérés plus faiblement que les mots qui apparaissent dans quelques documents.

La méthode LSI suppose des calculs plus intensifs que les deux méthodes antérieures. LSI s'intéresse aux proximités lexicales en incluant des poids pour les mots qui, sans être inclus nécessairement dans un document, peuvent apparaître en cooccurrence avec d'autres mots dans quelque groupe de documents. Typiquement, ce groupe de documents partagera un sujet commun [25].

Pendant que le problème des champs lexicaux peut être résolu avec LSI, la polysémie brouille toute méthode d'indexation : par exemple un mot tel que (كتاب) a beaucoup de sens. Le mot (كتاب) peut signifier un livre ou une lettre. Tout document avec des mots qui apparaît avec le mot (كتاب) aura des mots dans les vecteurs du document avec les deux sens. Bien que ce soit rare pour un document d'utiliser plus qu'un sens d'un mot, une collection de documents peut avoir des documents qui utilisent de multiples sens du mot. L'usage de mots polysémiques dans les documents aura un effet nuisible sur un index calculé avec LSI.

4.5.6 Mesure de similarité

L'autre utilisation de l'indexation consiste à calculer, pour chaque document, un score de similarité par rapport à tous les autres, et ainsi de créer un réseau de proximité de documents.

La plupart des méthodes de classification utilisent des techniques statistiques proches de celles employées par les moteurs de recherche pour évaluer la proximité d'un texte avec une requête [18], les mêmes méthodes sont utilisées pour mesurer la ressemblance entre un texte et un autre texte.

De nombreuses mesures de similarité ou dissimilarité entre les mots ont été définies jusqu'à présent. Citons par exemple quelques mesures fondées sur les cooccurrences de mots dans les corpus : la mesure d'*Information Mutuelle* [117], le *Rapport d'Association* [118] ainsi que d'autres coefficients tels que le *coefficient de Dice* [119] ou la *mesure de Jaccard* [120]. Enfin, dans le cadre supervisé de la classification de documents, [121] s'appuient sur

l'approche distributionnelle afin de définir une mesure de désimilarité relativement aux classes cibles que l'on cherche à apprendre

L'analyse factorielle des correspondances [73] permet également de représenter synthétiquement les documents sur un graphique à deux dimensions[31], et donc d'évaluer les distances, selon certains axes qu'il faut interpréter. Cette méthode, utilisée le plus souvent sur une matrice documents-mots a été appliquée sur une matrice documents-concepts.

Cette méthode met en évidence très rapidement les *proximités sémantiques* des documents. L'analyse factorielle des correspondances est une méthode statistique qui commence à donner de bons résultats quand le corpus de textes est suffisamment étendu pour que les co-occurrences de mots soient suffisamment significatives ; cela fonctionne d'autant mieux que les unités textuelles sont courtes, comme en bibliographie [74]. Par contre, lorsque les textes sont de longueur importante, les cooccurrences de mots perdent de leur signification (ce qui impose de segmenter les textes). Par contre les cooccurrences de concepts sont moins sensibles à la taille des textes. Une autre utilisation de l'Analyse Factorielle des Correspondances sur les concepts détectés est de classifier automatiquement les textes [75].

Assurer que des documents similaires aient toujours la même indexation est un problème particulièrement délicat, que l'indexation soit manuelle ou automatique. Lorsque le processus d'indexation est manuel, la consistance est posée sous la forme : est ce que deux indexeurs différents indexent de la même façon le même document ? Lorsque le processus est automatique, le problème est vu sous un autre angle : est ce que deux documents au contenu sémantique identique sont indexés de la même façon ? On mesure la consistance en comparant les réponses entre deux indexeurs différents (consistance inter-indexeur). Pour un document d et un terme t, la consistance est définie par : le nombre de termes affectés à d par les indexeurs A et B sur le nombre de termes affectés à d par les indexeurs A ou B. On peut également mesurer les réponses d'un même indexeur (consistance intra indexeur) [69].

La phase d'indexation se fait en représentant le document sous forme vectorielle (modèle le plus courant) : les coordonnées représentent les poids dans le document des termes retenus. Lors d'une classification, la pertinence d'un document par rapport à un autre est évaluée par le degré de similarité entre le vecteur du premier document et celui du deuxième pour une classi-

---

[31] Ou 3. En fait, ces axes correspondent aux facteurs cf Atlas linguistique de l'ISC Lyon

Un mot polysémique pourra être vu dans différents plans correspondants plus ou moins à divers champs lexicaux.

fication non supervisée, ou le degré de similarité entre le vecteur du document et ceux de la base de référence pour une classification supervisée.

Pour chaque document du corpus, on calcule le score de similarité avec l'ensemble de concepts des autres documents. On peut utiliser comme mesure de similarité la formule *Cosine* qui calcule le cosinus de l'angle entre le vecteur représentant un document et chaque document du corpus [72].

$$COSINE(d,r) = \frac{\sum_{ced \cap r} TFIDF_{c,d} . TFIDF_{c,r}}{\sqrt{(\sum_{ced} TFIDF_{c,d}^{\ 2}).(\sum_{cer} TFIDF_{c,r}^{\ 2})}}$$

avec : *c* : un concept, *d* : le document, *r* : le deuxième document, $TFIDF_{c,d}$: le score TFIDF du concept *c* dans le document *d.*

Dans la classification supervisée on extrait de la base N des nouveaux documents, ceux qui sont les plus similaires aux documents de référence R. La similarité *s* entre un nouveau document d_j de N et un document de référence d_k de R est mesurée par le cosinus de l'angle formé par les vecteurs représentant ces documents (mesure couramment employée dans les systèmes de recherche d'information) [79].

D'autres mesures de similarité existent : nombre de concepts communs entre le nouveau document et les documents de référence, somme des produits TFIDF ou encore mesure *Okapi*. Parmi ces mesures *Cosine* est souvent celle qui donne les meilleurs résultats [71]. En effet, elle pondère la somme des produits TFIDF par la taille des documents.

Des approches utilisent directement la proximité des termes pour le calcul du score des documents [122,123], mais il y a des méthodes basées sur la similarité statistiques entre mots. Par exemple en [105] on choisit un couple de mots extraits du corpus Reuters-21578, dont la dissimilarité est parmi les plus faibles. Les mots *« graphic »* et *« mattress »* n'entretiennent pas, à première vue, de relation sémantique, cependant leur distribution respective par rapport aux classes cibles (cf. figure 4.19), indique que ces deux mots jouent un rôle commun, ce qui explique leur proximité statistique. De la même manière, les deux mots *« graphic »* et *« soybean »* disposent d'une forte dissimilarité, apparaît sur leur distribution (cf. figure 4.20).

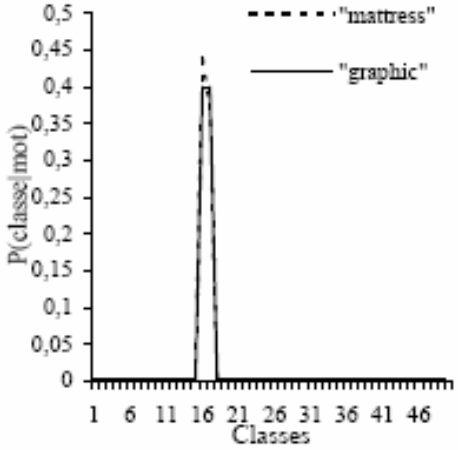


Figure 4.19 Distribution de deux mots fortement similaires sur 50 classes.

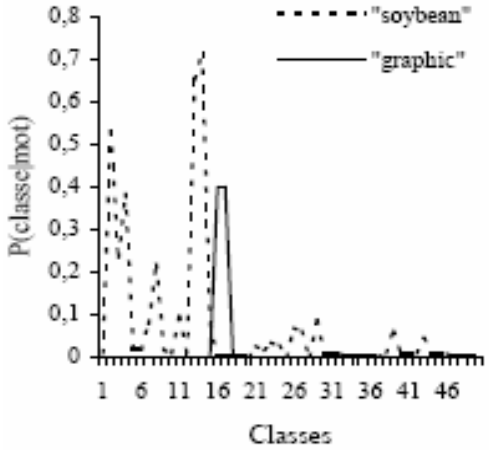


Figure 4.20 Distribution de deux mots fortement dissimilaires sur 50 classes.

4.6. Conclusion

Le traitement du document est devenu un enjeu. Les avancées sont dûes en grande partie au progrès effectués dans le traitement automatique de la langue naturelle. Des étapes de prétraitement ont lieu avant la fouille de données en tant que tel. Le prétraitement concerne la mise en forme des données entrées selon leur type, leur analyse, ainsi que le nettoyage des données, le traitement des données manquantes, la sélection d'attributs ou la sélection d'instances. Cette première phase est cruciale car du choix des descripteurs et de la connaissance précise de la population dépendra la mise au point des modèles de prédiction. L'information nécessaire à la construction d'un bon modèle de prévision peut être disponible dans les données mais un choix inapproprié de variables ou d'échantillon d'apprentissage peut faire échouer l'opération.

Chapitre 5. Classification des Documents Textuels

5.1. Introduction

La classification de documents est un domaine d'étude en plein essor en raison de la quantité d'information qui transite, notamment sur Internet, et de la valeur stratégique qu'elle revêt.

La classification automatique de documents devient nécessaire à cause du volume de documents échangés et stockés sur support électronique.

Le flux et la masse d'information disponible croissant de manière exponentielle, il est nécessaire de fournir aux lecteurs des mécanismes de filtrage de l'information qui lui permettront de prendre la décision de lire ou non un document [81].

Associer une classe à un texte libre est une opération coûteuse et longue, par conséquent, l'automatisation de cette opération est devenue un enjeu pour la communauté scientifique. [82]

Nous utiliserons, dans ce chapitre, les deux termes classement et classification[32] pour désigner le même concept à savoir la « catégorisation ».

5.2. Classification

5.2.1 Définition

La *classification* ou *catégorisation de textes* consiste à chercher une liaison fonctionnelle entre un ensemble de textes et un ensemble de catégories (étiquettes, classes). Cette liaison fonctionnelle, (modèle de prédiction), est estimée par un apprentissage automatique. Pour ce faire, il est nécessaire de disposer d'un ensemble de textes préalablement étiquetés, dit ensemble d'apprentissage, à partir duquel nous estimons les paramètres du modèle de prédiction le plus performant possible, c'est-à-dire le modèle qui produit le moins d'erreur en prédiction.

[32] En principe, la classification est la construction des classes, et le classement, le rattachement d'un objet à une classe définie.

5.2.2 Définition formelle

Formellement, la catégorisation de texte consiste à associer une valeur booléenne à chaque paire $(d_j, c_i) \in D \times C$, où D est l'ensemble des textes et C est l'ensemble des catégories selon que $d_j \in c_i$, ou non. Le but de la catégorisation de texte est de construire une procédure (modèle, classifieur) $\Phi : D \times C \rightarrow \mathbb{B}$ qui associe une ou plusieurs étiquettes (catégories) à un document d_j telle que la décision donnée par cette procédure « coïncide le plus possible » avec la fonction F: $D \rightarrow C$, la vraie fonction qui retourne pour chaque vecteur d_j une valeur c_i [82].

5.2.3 Les traits des documents

Afin d'apprendre à classer des objets par rapport à un ensemble cible de classes, ces derniers doivent être décrits selon des attributs (ou traits) pertinents ; or la construction d'un ensemble de tels attributs est une tâche difficile, notamment dans l'application à la classification de documents. En effet, lorsqu'il s'agit de classer automatiquement les documents par thématiques, l'analyse sémantique semble être, sinon l'unique, du moins la principale caractérisation possible des documents. Ainsi chaque document est perçu comme un "sac de mots" et l'ensemble du vocabulaire contenu dans les documents comme l'ensemble des attributs possibles. Cependant, la quantité de vocabulaire, son caractère redondant dans l'influence des mots pour la classification, la matrice résultante (documents × mots) très creuse sont autant de critères en faveur d'une réduction de la dimension de l'espace de description [100].

Plusieurs approches ont été proposées dans ce sens : la sélection des attributs pertinents en mesurant l'intérêt de chaque mot afin de supprimer ceux qui apportent peu d'information (gain d'information [125], test du χ^2 [126], etc.) ; le re-paramétrage des attributs pour en définir de nouveaux à partir de combinaisons et transformations des traits initiaux (LSI [115]) ; enfin le regroupement des attributs, permet de considérer les mots ayant un rôle similaire dans la classification comme un seul attribut [121].

5.2.4 Représentation

Un codage préalable du texte est nécessaire, comme pour l'image, le son, etc., [127], car il n'existe pas actuellement de méthode d'apprentissage capable de traiter directement des données semi-structurées, ni dans la phase de construction du modèle, ni lors de son utilisation en classement.

Comme les documents sont nombreux et/ou que leur nombre augmente sans cesse, il serait difficile de programmer à l'avance des règles de décision pour déterminer la classe d'un

nouveau document. Même si cela était possible, ces règles devraient être régulièrement modifiées par l'utilisateur pour qu'elles reflètent la réalité actuelle.

À partir de documents déjà classés, les méthodes d'apprentissage permettent de classer de nouveaux documents (Apprentissage supervisé),

De façon simple, le but de l'algorithme est de découvrir pourquoi chaque document d'exemple a été rangé dans telle ou telle classe, afin de prédire la classe de nouveaux documents à ranger dans le futur. Le but des algorithmes d'apprentissage supervisés donc est de trouver (par essais et erreurs) un modèle – une fonction mathématique – qui rende compte du lien entre des données d'entrée et les classes de sortie.

Le processus pour l'entraînement d'un classifieur comporte 4 étapes principales :

1. constituer un corpus d'apprentissage,
2. sélectionner les termes discriminants pour les catégories,
3. calculer le modèle d'apprentissage pour classer les documents,
4. évaluer le modèle d'apprentissage [81].

Remarque

Le texte à classer doit appartenir aux mêmes domaines que ceux utilisés lors de l'apprentissage. On ne saurait, par exemple, essayer de classer un article scientifique à partir d'un modèle construit sur un ensemble d'apprentissage constitué d'articles de journaux de mode [127].

5.3. Classification Supervisée Vs Classification non Supervisée

L'objectif de la catégorisation de texte est d'associer automatiquement une étiquette à tout nouveau texte à classer [82].

Le principe de base de la classification est de regrouper les documents similaires du point de vue de leur contenu. Deux approches sont possibles : [81]

- *La classification supervisée* ou classement automatique de documents dans des classes préexistantes (connues à l'avance) ;
- *La classification non supervisée* des documents ou clustering, c'est-à-dire la découverte de classes de documents sans a priori (on ne connaît pas les classes à l'avance).

La classification supervisée [124] consiste à identifier la classe d'appartenance d'un objet à partir de certains traits descriptifs. Cette approche permet le classement automatique de documents dans des classes préexistantes.

L'objectif est de trouver une liaison fonctionnelle, que l'on appelle également *modèle de prédiction*, entre les textes à classer et l'ensemble des catégories. Pour estimer le modèle de prédiction, il faut disposer d'un ensemble de textes préalablement étiquetés, dit ensemble d'apprentissage, à partir duquel on estime les paramètres du modèle de prédiction le plus performant possible, c'est-à-dire qui produit le moins d'erreurs en prédiction.

A la différence de la classification non supervisée où l'ordinateur doit découvrir lui-même des groupes de documents, la classification supervisée suppose qu'il existe déjà une classification de documents. C'est le cas par exemple d'une bibliothèque ou d'un moteur de recherche. Le but est alors de classer automatiquement un nouveau document. Il s'agit donc d'apprendre d'abord un modèle, ou classifieur, à partir d'un ensemble d'entraînement composé de couples (objet, classe) [105].

Contrairement à la classification non supervisée, la classification supervisée peut mesurer l'importance de chaque mot pour classer de nouveaux documents. Par exemple, une mesure (*information gain*) calcule la typicité d'un terme. Plus un mot est lié à une catégorie et pas aux autres, et plus il est important : si un nouveau document le contient, ce mot sera très discriminant. De nombreuses mesures semblables ont été mises au point.

Enfin, à l'inverse de la classification non supervisée, il est ici simple d'évaluer les résultats d'une classification. Parmi les N exemples de documents classés, on utilise une partie des documents pour l'entraînement, et le reste pour le test. Pendant la phase de test, on soumet chaque document à l'algorithme de classification et on regarde simplement si la machine trouve la bonne classe. Bien sûr, le résultat de ce test n'est en rien garanti lorsque la machine aura à classer de nouveaux documents ! (réussir le test est nécessaire, sans être suffisant).

5.4. Processus de catégorisation

5.4.1 Principe

Le processus de catégorisation présuppose la construction d'un modèle de prédiction qui, en entrée, reçoit un texte et, en sortie, lui associe une ou plusieurs étiquettes. Pour identifier la catégorie ou la classe à laquelle un texte est associé, un ensemble d'étapes est habituellement suivi. Ces étapes concernent principalement la manière dont un texte est représenté, le choix de l'algorithme d'apprentissage à utiliser et comment évaluer les résultats obtenus pour garantir une bonne généralisation du modèle appris.

Le processus de catégorisation, intégrant la phase de classement de nouveaux textes, est illustré par la figure 5.1.

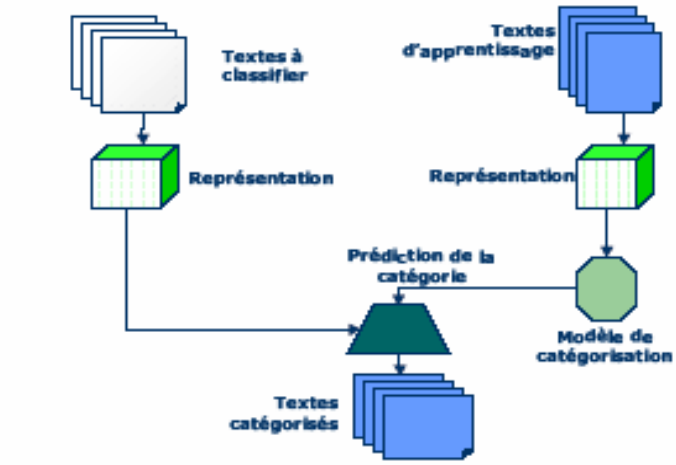


Figure 5.1 Processus de catégorisation des textes.

Il comporte deux phases que l'on peut distinguer comme suit :

1. **l'apprentissage** comprend plusieurs étapes et aboutit à un modèle de prédiction :

   a) formation d'un ensemble de textes étiquetés (catégories) ;

   b) extraction des k descripteurs (t_1; ..; t_k) les plus pertinents, formation du tableau « descripteurs × individus »,

   c) application d'un algorithme d'apprentissage sur ce tableau afin d'obtenir un modèle de prédiction Φ.

2. **le classement** d'un nouveau texte d_x, qui comprend deux étapes :

   a) recherche puis pondération des occurrences (t_1; ...; t_k) des termes dans le texte d_x à classer ;

   b) application du modèle Φ sur ces occurrences afin de prédire l'étiquette du texte d_x.

Les k descripteurs les plus pertinents (t_1; ...; t_k) sont extraits lors de la première phase par analyse des textes du corpus d'apprentissage. Dans la seconde phase, celle du classement d'un nouveau texte, nous cherchons simplement la fréquence de ces k descripteurs (t_1; ...; t_k) dans ce texte à classer [82].

5.4.2 Evaluation de la qualité d'un classifieur

La performance d'un classifieur dans la catégorisation de textes est souvent mesurée via la *précision* et le *rappel* [82].

5.5. Les applications de la catégorisation

Depuis [128], la catégorisation de textes est utilisée dans de nombreuses applications. Parmi ces domaines figurent : l'identification de la langue [104], la reconnaissance d'écrivains [129, 130,131]**,** la catégorisation de documents multimédia [132], le filtrage (la détection de spams pour ensuite les supprimer [133, 134]), le routage (la diffusion sélective d'informations [135]) et bien d'autres.

Difficultés de la catégorisation de textes

L'application des algorithmes d'apprentissage aux données textuelles introduit des difficultés supplémentaires : la grande dimensionnalité, la proximité lexicale, la polysémie, la subjectivité de l'attribution d'un texte à une telle ou telle catégorie…

- Dans la catégorisation de textes, la grande dimensionnalité peut réduire l'efficacité des algorithmes d'apprentissage. En effet, la plupart des algorithmes d'apprentissage sophistiqués, sont sensibles au |T|, nombre de variables utilisées pour coder les textes, car |T | est un paramètre de la complexité de l'algorithme. C'est pourquoi une méthode de réduction de dimension doit être utilisée avant d'estimer les paramètres d'un classifieur.
- Un même mot ou une même expression peuvent avoir plusieurs sens différents. Les ambiguïtés intrinsèques des mots ou des phrases apparaissent à deux niveaux : lexical et syntaxique. Il faut y rajouter les ambiguïtés pragmatiques de rapport au contexte ([136]).
- Il existe multiples manières d'exprimer la même réalité, avec des nuances diverses. (Problème des synonymes/hyponymes/hyperonymes/holonymes/méronymes).
- Contrairement, à d'autres situations où l'appartenance à une classe est objective (un client achète, ou non, un produit ; un patient a, ou n'a pas, tel microbe), l'attribution d'une catégorie à un texte est subjective [137]. En effet, la catégorie est attribuée en fonction du contenu sémantique de ce texte, qui est une notion subjective, et dépend du jugement d'un expert. Souvent les experts ne sont pas d'accord sur la classe d'appartenance d'un document [138].

5.6. Les algorithmes de classification des documents

Il existe un grand nombre d'algorithmes de catégorisation [124] : régression linéaire, machines à support vectoriel, réseaux de neurones, algorithmes génétiques, modèles de Markov cachés, RocChio, k plus proches voisins, Classifieur linéaire [139], Winnowl [140] et [141],…

Une revue d'ensemble des méthodes de classification de documents peut être trouvée, entre autres, dans [127].

La catégorisation de textes comporte un choix de techniques d'apprentissage disponibles. Parmi les méthodes d'apprentissage les plus souvent utilisées figurent l'analyse factorielle discriminante [142], la régression logistique [143], les réseaux de neurones [145, 93, 145], les plus proches voisins [146, 147], les arbres de décision [148, 149], les réseaux bayésiens [150, 151, 133, 152, 153], les machines à vecteurs supports [154, 155, 156, 108, 157] et, plus récemment, les méthodes dites de boosting [158, 159] et les algorithmes génétiques [160]…

5.6.1 K plus proches voisins

La méthode des k plus proches voisins est une méthode d'apprentissage à base d'instances. Elle ne comporte pas de phase d'entraînement, elle est basée sur l'hypothèse qu'un document devrait être classé dans la même classe que Ses k plus proches voisins dans le corpus d'apprentissage [81].

Elle a prouvé son efficacité en traitement de données textuelles [161].

5.6.2 Arbres de décision

Les arbres de décision sont utilisés fréquemment dans la catégorisation de textes [162]. [163] utilise la méthode ID3 (pour « Induction Decision Tree ») [164], la méthode C4.5 et la méthode C5 sont utilisées par [165].

5.6.3 Réseaux de neurones

Les réseaux de neurones sont actuellement plus utilisés dans plusieurs domaines du traitement de l'information textuelle et plus particulièrement celui de l'indexation [115], et la catégorisation textuelle [166].

Le modèle ART1 travaille avec des données binaires, ce qui le rend spécialement utile pour des tâches de classification textuelles [167].

5.6.4 Naïf de Bayes

Le classifieur naïf de Bayes est traditionnellement utilisé pour la classification de documents en raison de ses performances reconnues dans ce domaine [105]

5.6.5 Support Vector Machine

SVM est un des algorithmes les plus performants en classification textuelle [154]. Un nombre important d'auteurs étudient les SVM et les appliquent au texte, comme dans [154, 155, 156, 108, 157]…

5.7. Travaux de recherche sur la classification

[39] présente un système d'acquisition de familles morphologiques (des groupes de mots liés deux à deux par un lien morphologique d'affixation (préfixation ou suffixation) ou de composition) qui procède par apprentissage non supervisé à partir de listes de mots extraites de corpus de textes.

La question de l'acquisition de familles morphologiques est formulée comme un problème de classification.

L'approche consiste à prendre une liste de mots et former des familles par groupements successifs, similairement aux méthodes de classification ascendante hiérarchique. Les critères de regroupement reposent sur la similarité graphique des mots ainsi que sur des listes de préfixes et de paires de suffixes acquises automatiquement à partir des corpus traités. Les résultats obtenus pour des corpus de textes de spécialité en français et en anglais sont évalués à l'aide de la base CELEX et de listes de référence construites manuellement. L'évaluation démontre les bonnes performances du système, indépendamment de la langue.

[177] présente un mécanisme automatique pour classer les messages de clients, basé sur les techniques de fouille de textes et le support vector machine (SVM), Le mécanisme proposé peut filtrer les messages automatiquement dans le but d'extraire les réclamations et convenablement augmenter la productivité du département et augmenter la satisfaction des clients.

Cette étude emploie le *p-control chart* pour contrôler le taux des réclamations sur le niveau de la qualité du service attendue pour l'exécution du website. Les résultats expérimentaux ont démontré que la capacité des SVM à reconnaître correctement les messages défectueux a dépassé 83% avec une moyenne de 89% pour le mécanisme de classement, et le *p-control chart* était capable de refléter les changements exceptionnels de qualité de service.

[37] développe une méthodologie de classification et de recherche des documents basée sur les réseaux de neurones. Le processus de classification commence par l'extraction des expressions-clés du document avec le traitement automatique des textes et détermine la signification des expressions-clés d'après leur fréquence dans le texte. Pour maintenir un nombre d'expressions-clés indépendantes, une analyse de corrélation est appliquée pour calculer les ressemblances entre expressions-clés. Les expressions avec les plus hautes corrélations sont synthétisées dans un plus petit ensemble d'expressions. Finalement, le modèle de propagation est adopté comme classifieur. La sortie cible identifie la catégorie d'un document basé sur la classification hiérarchique

Pour améliorer l'exactitude de classification automatique, les modèles BPN sont soumis à un apprentissage continu et un mécanisme de rétroaction ajuste les poids du modèle BPN.

Pour la classification neuronale, [168] effectue un filtrage du lexique : sont supprimés les termes fonctionnels et ceux dont la fréquence a été jugée non pertinente pour la classification (c'est-à-dire les termes trop fréquents et les pas assez fréquents). De plus, le lexique est lemmatisé. Suite à ces opérations, le lexique épuré est composé de 306 termes. La fréquence des termes retenus varie entre 5 et 56 apparitions. De plus, est privilégiée une segmentation par mots, à raison de 150 mots par segment. La classification utilise le classifieur neuronal ART1.

Le processus de segmentation a permis de découper le corpus initial en 154 segments de 150 mots. En utilisant le classifieur ART1, ces 154 segments furent regroupés en 83 classes. D'un point de vue strictement technique, la qualité des résultats (moyenne de 1.86 segment par classe) est manifestement discutable.

L'approche *classphère* [167] permet d'appliquer les performances classificatoires dynamiques des réseaux de neurones à des corpus textuels et produit des regroupements susceptibles d'interprétations sémantiques.

Bien que puissant pour des applications textuelles [169, 170], ART1 présente les handicaps de ne pas pouvoir faire appartenir un même segment à plusieurs classes, et de dépendre du choix de p. par l'utilisateur. C'est pourquoi [167] développe l'algorithme incrémental Classphères, basé sur des hypersphères.

Dans le cadre du projet CONTERM du LANCI de l'UQAM a été développée une chaîne de traitement d'information textuelle. Elle comporte des processus de segmentation, de

filtrage du lexique (enlever les mots fonctionnels, les mots à haute fréquence d'apparition, etc.) et de lemmatisation au besoin de l'utilisateur.

Pour la segmentation ils n'ont utilisé que les mots. La segmentation transforme un texte initial en un ensemble de P vecteurs binaires. Chaque segment est alors représenté par un vecteur $\xi \in \mathbb{B}^N$, où N est la taille du lexique supposé ordonné.

Chaque composant du vecteur montre la présence ($\xi_i = 1$) ou l'absence $\xi_i = 0$ du mot i dans un segment. L'ensemble d'apprentissage est alors représenté par une matrice creuse binaire.

La classification s'effectue alors sur l'ensemble d'apprentissage $\xi_\mu \in \mathbb{B}^N$ où μ balaie l'ensemble des P segments. Le classifieur est un réseau de neurones à apprentissage non supervisé qui regroupe des segments similaires au sens d'une mesure de distance adéquate.

Les vecteurs, pouvant s'interpréter comme des points dans un espace, on utilise souvent des mesures de distance entre eux.

Pour la classification, ils ont utilisé les distances de Minkowsky de degré p=1 et 2.

- p=1 donne la *distance par blocs* ou *de Manhattan* en général, dite *distance de Hamming* quand les vecteurs sont à composantes binaires ;
- p=2 donne la *distance euclidienne ;*
- Si p tend vers l'infini, on obtient la *distance ultramétrique*, ou de Tchebytcheff (et un système de sphères de même rayon devient un système de classes d'équivalence)

Classphères travaille sur un ensemble d'apprentissage en créant des hypersphères regroupant les segments qui se ressemblent dans l'espace des mots.

Si l'on utilise la distance de Hamming, alors on cherche les voisins les plus proches de chaque segment, en les regroupant par rapport à une distance minimale. L'algorithme commence par créer la table triangulaire H des distances de Hamming entre segments, puis le vecteur V qui correspond à la distance minimale trouvée dans chaque colonne de H, c'est-à-dire, dans chaque segment. Une classe (ou hypersphère) de rayon r est ainsi formée en cherchant pour chaque segment μ, les segments ν voisins, donnés par $d_H(\mu, \nu) \leq r$. Plus la segmentation est petite, plus on obtient de classes pour le même texte. [167]

Classphères est un algorithme qui peut être spécialement utile pour la classification de textes dynamiques, par exemple sur le web.

[131] traite des classifieurs bayésiens appliqués à la reconnaissance de l'écrivain. Les phrases sont divisées en deux classes, les phrases de Chirac et les phrases de Mitterrand. Cha-

que phrase est divisé en groupes lexicaux, la présence d'un mot i dans une phrase, dont l'auteur est inconnu, permet au classifieur de déterminer une probabilité P_{ci} d'appartenir à la classe Chirac et une probabilité P_{mi} d'appartenir à la classe Mitterrand. Les probabilités individuelles des mots sont combinées pour obtenir une probabilité unique pour chaque phrase (par la formule de Fisher proposée par [171])

Alors les probabilités individuelles des mots sont obtenues par entraînement sur un corpus de phrases d'auteur connu.

Il y a un problème avec les probabilités calculées, si un mot est très rare. C'est le grand nombre de répétitions de l'événement qui indique qu'il y a de fortes chances que la prochaine fois l'apparition de ce mot indique une phrase de X ou Y.

En sortie du classifieur on souhaite obtenir une seule valeur de probabilité pour chacun des ensembles de phrases de Chirac et de Mitterrand après combinaison des probabilités

Le corpus d'entraînement est divisé en deux parties, l'une pour la création automatique du classifieur, l'autre pour tester l'apprentissage.

Une formule de combinaison été utilisé pour créer le classifieur bayesien et effectuer la classification.

[18] exploite des outils d'étiquetage syntaxique et de lemmatisation afin d'expérimenter des techniques de classification et de segmentation destinées à améliorer la pertinence des réponses fournies pour une requête.

Comme la plupart des systèmes de recherche documentaire, le logiciel SIAC (Segmentation et Indexation Automatiques de Corpus), développé au sein du Laboratoire d'Informatique d'Avignon (LIA), fournit en réponse à une requête une liste de documents classés par ordre de pertinence décroissante. Usuellement, cette liste est assez difficilement exploitable à cause de sa longueur : l'utilisateur ne peut lire tous les documents proposés et néglige certains d'entre eux qui, bien que pertinents, sont classés en fin de liste.

[18] présente deux algorithmes appliqués aux documents rapportés en réponse aux requêtes.

- La première méthode est destinée à fournir à l'utilisateur une vision plus claire des réponses en les classant de manière thématique. Cet algorithme, basé sur l'exploitation d'arbres de décision, est utilisé de manière originale pour la recherche documentaire. Les résultats montrent la capacité de la méthode à isoler les documents pertinents sans nécessiter des temps machine importants.

- La nouvelle méthode de segmentation proposée utilise directement les résultats donnés par la classification. Une segmentation des documents rapportés est en effet déduite à partir du contenu des feuilles de l'arbre de décision : si deux phrases contiguës d'un document n'appartiennent pas à la même feuille c'est parce qu'elles traitent de thématiques différentes. Une seconde recherche est effectuée non plus sur les documents mais sur les segments de documents. L'utilisateur a la possibilité d'accéder plus rapidement à l'information qu'il recherche. Alors la pertinence finale d'un document est égale à la plus grande valeur de pertinence trouvée entre la requête et chacun des segments du document.

[176] présente un *système de classification de textes en langue arabe* par l'utilisation de la théorie de *distance intertextuelle*. Le système vise la classification de textes en langue arabe dans un but de catégorisation et d'indexation. Les auteurs proposent de définir un processus de traitement recevant en entrée un texte brut pour donner en sortie la catégorisation de ce dernier. Cette catégorisation peut se faire par rapport à une référence existante ou par rapport à un autre texte en entrée.

L'utilisation de la théorie de la distance intertextuelle pour la mise en place d'une métrique de classification impose une étape de lemmatisation des textes. Cette étape nécessaire prépare les textes en les décomposant, ce qui permet l'exploitation des structures grammaticales dans la détection des classes d'équivalence entre segments de textes. Les auteurs ont exploité la grammaire de la langue arabe pour intégrer la notion de classes grammaticales au niveau du lemmatiseur ainsi que dans la métrique.

De cette manière le lemmatiseur fonctionne indépendamment du jeu de la structure adopté. Enfin, ils ont introduit la notion de poids associé aux classes grammaticales au niveau de la métrique.

Pour pouvoir décider du degré de ressemblance entre deux textes et ceci par rapport à l'utilisation d'un vocabulaire commun il faut utiliser un indice numérique significatif de cette distance, la distance intertextuelle D est la mesure de degré de ressemblance ou de dissemblance entre textes.

Une nouvelle formule est mise au point pour le calcul de la distance intertextuelle, intégrant la pondération, alors une intégration de la théorie de la distance intertextuelle en tant qu'outil de classification de textes en langue arabe.

5.8. Comparaison des algorithmes de classification

Beaucoup d'approches ont été utilisées pour la catégorisation de textes. Quelle est donc la meilleure méthode pour la catégorisation de textes ?

Pour comparer les performances de deux méthodes, soit on tente d'appliquer les deux méthodes aux mêmes données en utilisant les mêmes mesures de performance, soit on adopte une méthode d'évaluation contrôlée [101]. La première solution est difficile à réaliser :

- les méthodes n'ont pas été appliquées sur les mêmes jeux de données ;
- les auteurs n'utilisent pas les mêmes mesures de performance.

[101] propose deux méthodes pour comparer les classifieurs et les évaluer :

- *une comparaison directe* : il s'agit de l'utilisation de plusieurs méthodes par le même auteur ; de cette manière, le découpage et les mesures sont identiques pour toutes les méthodes. [147] compare les machines à vecteurs supports, les plus proches voisins, les réseaux de neurones, une combinaison linéaire, et des réseaux bayésiens. [108] proposent une série de comparaisons en mettant en compétition une variante de l'algorithme de Rocchio (appelée find similar), des arbres de décision, des réseaux bayésiens et des machines à vecteurs supports. Cette méthode de comparaison est la plus crédible de point de vue scientifique [172].
- *une comparaison indirecte* : soient Φ_1 et Φ_2 deux classifieurs. Ces deux classifieurs peuvent être comparés si deux conditions sont réunies :
  - les deux classifieurs Φ_1 et Φ_2 sont testés par différents groupes de chercheurs (même avec de conditions d'expérimentation différentes) sur deux collections Ω_1 et Ω_2 respectivement ;
  - un ou plusieurs classifieurs de « références » Φ_3, Φ_4, ..., Φ_m sont testés sur les deux collections Ω_1 et Ω_2 par des comparaisons directes ; ceci donne une idée du niveau de difficulté d'apprentissage sur chaque collection.

Dans ce qui suit nous présentons quelques résultats de comparaison des algorithmes de classification des textes.

Concernant les performances des classifieurs, [82] donne les résultats suivants :

- Les classifieurs par combinaison de décisions, les SVM, les méthodes à base d'exemples donnent les meilleurs résultats.

- Les réseaux de neurones, les classifieurs « en ligne » donnent des résultats inférieurs aux précédents.
- Les classifieurs linéaires tels que la méthode de Rocchio[33] et le classifieur naïf de Bayes, donnent souvent de mauvais résultats.

Concernant les dimensions, les machines à vecteurs supports sont capables de manipuler des vecteurs de grandes dimensions alors que, pour les réseaux de neurones, il est préférable de limiter la dimension [82] des vecteurs d'entrées. Rocchio [173] ou kNN sont généralement préférables [174], alors que pour des textes plus riches, Winnow [141] ou SVM [154] sont connus pour être robustes lorsque les termes sont très nombreux.

Plus précisément [32] trouve que Rocchio est moins performant que les k-PPV, eux-mêmes légèrement inférieurs aux SVMs.

[82] conclut que les RBFs avec noyau $\chi 2$ semblent très performants sur la catégorisation de texte. En fait la SVM multiclasse s'est avérée aussi puissante, mais les RBF sont beaucoup plus simples à implémenter et beaucoup plus rapides

[175] a trouvé de bons résultats avec l'algorithme SVM ;

[154] explique (partiellement) le bon comportement des SVMs pour la catégorisation de texte par sa capacité à traiter de nombreuses dimensions sans avoir à sélectionner des variables.

[147] conclut que les SVM sont meilleures que les k-plus proches voisins qui sont eux-mêmes meilleurs que LLSF, et que les réseaux de neurones multicouches basés sur la rétropropagation, eux-mêmes meilleurs que l'algorithme Bayésien Naïf.

[108] présente des comparaisons selon plusieurs critères

1. temps

Find Similar est la méthode d'apprentissage " la plus rapide " parce qu'il n'y a pas de minimisation explicite de l'erreur. SVM linéaire est le prochain le plus rapide. Les deux sont plus rapides que les réseaux bayésiens ou les arbres de décision.

2. vitesse de classification des nouvelles instances

[33] La méthode de Rocchio est un classifieur linéaire proposé dans [56] pour améliorer les systèmes de recherche documentaires.

Tous les classifieurs explorés sont très rapides dans cette considération, tous exigent moins de 2 ms pour déterminer si un nouveau document devrait être assigné à une catégorie particulière.

3. exactitude de la classification

[108] a utilisé la moyenne de précision et du rappel (breakeven point) pour comparer les algorithmes.

- Support Vector Machines été la méthode la plus exacte, avec une moyenne de 92% pour les 10 catégories les plus fréquentes et 87% sur toutes les 118 catégories.
- L'exactitude pour les arbres de décision été de 3.6% inférieur, avec une moyenne de 88.4% pour les 10 catégories les plus fréquentes.
- Les réseaux bayésiens donnent quelques faibles améliorations de performance par rapport Naïve Bayes.
- SVMs et arbres de décision classifient avec grande exactitude.

Conclusion

Dans ce dernier chapitre nous avons présenté la notion de catégorisation de textes, nous avons vu la différence entre les classifications supervisées et non supervisées et nous avons vu quelque applications de la catégorisation. Nous avons rapporté différents travaux effectués dans ce domaine, et comparé les différents algorithmes appliqués aux données textuelles dans la perspective de travailler sur la classification.

Conclusion

L'utilisation de la langue arabe comme moyen de communication à travers le support informatique a été longtemps appréhendé avec beaucoup d'hésitation par la communauté scientifique, notamment celle du monde arabe où cet outil trouvera beaucoup d'utilisations importantes.

Nous avons entrepris ce travail pour montrer que cela est faisable, et aussi pour relever le défi visant à offrir des outils informatiques pour l'analyse des documents en langue arabe. En effet nous avons montré que les études théoriques issues du Traitement Automatique des Langues Naturelles s'adaptent bien au traitement de la langue arabe – avec quelques spécificités.

Dans le souci de présenter un travail regroupant le maximum de concepts et d'approches de fouille de données et de classification nous avons présenté les différents algorithmes et montré leurs avantages et inconvénients pour la classification textuelle.

Nous estimons avoir atteint les objectifs que nous nous étions fixés, en présentant les techniques de fouille de données et de classification et En réalisant un analyseur qui prépare les données textuelles par une suite d'opérations linguistiques et statistiques pour être exploitable par les techniques de fouille de données.

Dans ce travail nous nous sommes intéressés à l'analyse des textes. En premier lieu une segmentation est effectuée, puis nous avons appliqué une analyse morphologique détaillée spécifique à la langue arabe avec comme résultats les tableaux lexicaux ; ces derniers sont affinés par une analyse syntaxique puis des calculs statistiques sont appliqués pour favoriser certains mots par rapports à d'autres. Les tableaux lexicaux sont exploités par les techniques de fouille de données.

Ce travail ouvre diverses perspectives.

- Un premier point concerne la prise en compte de toutes les formes des mots de la langue arabe dans l'analyse morphologique ;

- Un deuxième point concerne l'aspect statistique, pour comparer les textes par la distribution de leurs mots, et ajouter d'autres critères pour le calcul de la similarité ;
- Un troisième point concerne l'application des différents algorithmes de fouille de données pour la classification sur des textes en langue arabe.

Nous pensons être arrivés à nos objectifs, mais ceci n'est qu'un rayon de lumière parmi tant d'autres. Il faudrait maintenant aller au-delà, car il reste beaucoup à découvrir pour donner un plus grand élan à ce genre d'études et déboucher sur de bons outils au service de la langue arabe.

La langue arabe est aussi riche que vaste, et nous espérons que notre étude apporte quelques solutions exploitables dans l'immédiat, et tout au moins quelque satisfaction au courageux lecteur.

Médiagraphie

[01] Rachel Konrad, 2001, *Data Mining: Digging user info for gold*, *ZDNET News*, February 7, http://news.zdnet.com/2100-9595_22-528032.html?legacy=zdnn consulté le 15/04/2007

[02] The Technology Review Ten, *MIT Technology Review*, January/February 2001.

[03] The Gartner Group, *www.gartner.com.*

[04] David Hand, Heikki Mannila & Padhraic Smyth, 2001, *Principles of Data Mining*, MIT Press, Cambridge, MA.

[05] Peter Cabena, Pablo Hadjinian, Rolf Stadler, JaapVerhees & Alessandro Zanasi, 1998, *Discovering Data Mining : From Concept to Implementation*, Prentice Hall, Upper Saddle River, NJ.

[06] Stéphane Tufferry, 2002, *Fouille de données et scoring, bases de données et gestion de la relation client*, Dunod, Paris

[07] Bill Clinton, New York University speech, *Salon.com*, December 6, 2002, http://dir.salon.com/story/politics/feature/2002/12/06/clinton/index1.html, consulté le 15/4/2007

[08] John Naisbitt, 986, *Megatrends*, 6th ed., Warner Books, New York.

[09] Rémi Gilleron & Marc Tommasi, 2000, *Découverte de connaissances à partir de données*, http://www.grappa.univ-lille3.fr/polys/fouille/sortie005.html

[10] Peter Chapman, Julian Clinton, Randy Kerber, Thomas Khabaza, Thomas Reinart, Colin Shearer & Rudiger Wirth, 2000, *CRISP–DM Step-by-Step Data Mining Guide*, http://www.crisp-dm.org/.

[11] Fayyad, Piatesky – Shapiro, Smyth et Uthurusamy, 1996, *Advances in knowledge discovery and data mining* , AAAI Press/MIT Press,

[12] René Lefébure, Gilles Venturi, 2001, *Data mining. Gestion de la relation client. Personnalisation de sites web. Eyrolles*

[13] Daniel T. Larose, 2005, *Discovering knowledge in data an introduction to data mining*

[14] http://www.kdnuggets.com/polls/2002/methodology.htm

[15] NCR Systems Engineering Copenhagen (USA and Denmark), DaimlerChrysler AG (Germany), SPSS Inc, 2000, CRISP-*DM 1.0 Step-by-step data mining guide*, (USA) and OHRA Verzekeringen en Bank Groep B.V (NL).

[16] http://www.grappa.univ-lille3.fr/polys/fouille/sortie005.html

[17] J.B. MacQueen , 1967, *Some Methods for classification and Analysis of Multivariate Observations*, Proc. of 5-th Berkeley Symposium on Mathematical Statistics and Probability", Berkeley, University of California Press, 1:281-297

[18] Patrice Bellot, Marc El Bèze, 2001, *Classification et segmentation de textes par arbres de décision. Application à la recherche documentaire*, Technique et Science informatiques. Vol. 20, n° 1/2001, pp 107-134.

[19] http://culturetic.canalblog.com/archives/2005/10/10/876485.html. Consultée le 19/7/2007

[20] http://www.web-datamining.net/forum/faq_tm.asp Consultée le 19/7/2007

[21] http://www.datalgo.com Consultée le 19/7/2007

[22] http://www.atala.org/AtalaPedie/index.php?title=Fouille_de_textes Consultée le 19/7/2007

[23] http://www.definitions-marketing.com/ Consultée le 19/7/2007

[24] Frécon L, 2002, *Eléments de mathématiques discrètes*, PPUR, Lausanne (CH). 400 p.

[25] Manu Konchady, 2007, *Text Mining Application Programming*, Charles River Media Programming series,USA

[26] Ronen Feldman & James Sanger, 2007, *The text mining handbook, advanced approaches in analyzing unstructured data*, Cambridge University Press, New York, USA

[27] Baeza_yates R. & B Ribeiro-Neto, 1999, *Modern information retrieval.* ACM press books.

[28] Manning C.D. & H. Schutze, 1999, *Foundation of statistical natural language processing*. MIT press.

[29] Pierce, J.R, 1980, *An introduction to information theory symbols, signals and noise*, Dover publications.

[30] Hearst, M.A, 1999, *Untangling text data mining*. Proc. of the ACL'99, University of Maryland.

[31] Hearst, M. A et al, 2000 *The debate on automated essay grading*, IEEE Intelligent systems (September 2000): pp 22-37.

[32] Romain Vinot , Natalia Grabar & Mathieu Valette, 2003, *Application d'algorithmes de classification automatique pour la détection des contenus racistes sur l'Internet,* TALN 2003, Batz-sur-Mer

[33] Ricco Rakotomalala, 2005, *Arbres de Décision*, Revue MODULAD, N33.

[34] Dougherty J, Kohavi R., Sahami M., 1995, *Supervised and unsupervised discretization of continuous attributes*, in Proc. of the 12th International Conference on Machine Learning.

[35] *Introduction aux réseaux de neurones*, http://wwwobs.univbpclermont.fr/atmos/enseignement/cours-Master-2A/cours_RN_2006.pdf

[36] I.C. Lerman, 1970, *Les bases de la classification automatique*, Gauthier-Villars, Collection Programmation, Paris.

[37] Amy J.C. Trappey, Fu-Chiang Hsu Charles V. Trappey & Chia-I. Lin, 2006, *Development of a patent document classification and search, platform using a back-propagation network*, Expert Systems with Applications, Elsevier

[38] Gilles Bisson, sd, *Une nouvelle approche pour la classification conceptuelle ascendante*, CNRS, Projet SHERPA, Unité de recherche INRIA Rhône-Alpes

[39] Delphine Bernhard, 2007, *Apprentissage non supervisé de familles morphologiques par classification ascendante hiérarchique,* TALN 2007, Toulouse.

[40] Day W., Edelsbrunner H, 1984, *Efficient Algorithms for Agglomerative Hierarchical Clustering Methods*. Journal of Classification. Vol. 1. 1-24.

[41] Lehmam A. & Bouvet P, 2004, *Un résumé de textes, application à la langue arabe*. TALN (2004)

[42] Chalabi, Achraf, 2000, *MT-Based Transparent Arabization of the Internet TARJIM.COM*. In White, J.S.(Ed) AMTA (2000) Springer : Verlag Berlin Heidelberg.

[43] Daimi, Kevin, 2001, *Identifying Syntactic Ambiguities in Single-Parse Arabic Sentence.* In Computer and humanities.

[44] Ali, Nabil, 2003, *The Second Wave of Arabic Natural Language Processing.*

[45] Douzidia Fouad Soufiane, Guy Lapalme, 2005, *Un système de résumé de textes en arab,* 2ème Congrès International sur l'Ingénierie de l'Arabe et l'Ingénierie de la langue, Alger.

[46] Attia Mohamed, 2004, *A Report on the introduction of Arabic to ParGram*. The ParGram Fall Meeting, National Centre for language Technology, Ireland.

[47] Groblink M., Mladenic D, 2004, *Text mining tutorial* Slovenia.

[48] Attia Mohamed A, 2005, *Developing Robust Arabic Morphological Transducer Using Finite State Technology* In 8 th annual CLUK Research Colloquium.

[49] Mohamadi T, S. Mokhnache, 2002, *Design and development of Arabic speech synthesis*, WSEAS, Greece.

[50] Larkey L. S., Ballesteros L. AND Connell M, 2002, *Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis*, in Proc. of the 25th Annual Inter-

national Conference on Research and Development in Information Retrieval (SIGIR 2002), Tampere, Finland.

[51] Attia Mohamed A, 2000, *A large-scale computational processor of the Arabic morphology*, A Master's Thesis, Cairo University, Egypt.

[52] Bessou S, Louail M, Refoufi A, Kadem Z, Touahria M, 2007, *Un système de lemmatisation pour les applications de TALN*, Colloque international traitement automatique de la langue arabe CITALA 2007, Rabat. Maroc.

[53] Selmane S & Zergoug D, 1995, *Correcteur d'erreurs de frappe d'un éditeur de textes arabes non voyellés* Alger.

[54] Taibi Nacera, 1997, *Contribution a l'étude du traitement automatique des erreurs dans un texte écrit en arabe* Thèse de magister ENS (SH) Alger.

[55] Zemirli Zouhir, 1996, *Un analyseur destiné à l'aide à la construction d'une base de données lexicales de la langue arabe*. Colloque international "Langues situées, technologie et communication" IERA.

[56] Lasakri Mohamed Taib, 1994, *Sémantique du langage naturel à travers un système support de thésaurus*,Thèse d'état.

[57] Hassoun M.O, 1989, *Conception d'un dictionnaire pour le traitement automatique de l'arabe dans différents contextes d'applications*, Thèse de doctorat, Université Lyon 2.

[58] Aljlayl M., Frieder O., 2002, *On Arabic Search: Improving the Retrieval Effectiveness via a Light Stemming Approach,* In 11 the International Conference on Information and Knowledge Management (CIKM), Virginia (USA), p. 340-347.

[59] Darwish, K. & D. Oard. *CLIR Experiments at Maryland for TREC, 2002: Evidence Combination for Arabic-English Retrieval in TREC*, Gaithersburg, MD.

[60] Darwish, K, 2003, *Probabilistic Methods for Searching OCR-Degraded Arabic Text*, Doctoral dissertation, University of Maryland.

[61] Chen A. & Gey F, 2002, *Building an Arabic Stemmer for Information Retrieval*. Proc of the Eleventh Text REtrieval Conference (TREC 2002). National Institute of Stadards and Technology.

[62] Fluhr C, 2000, *Indexation et recherche d'information textuelle*, Ingénierie des langues, Hermes.

[63] Aït Taleb .s & Benjelloun .f, sd, *Automatisation de la chaîne de production terminologique reconnaissance extraction des termes* Article IERA.

[64] Calabretto S, 2003, *Recherche d'information*, LIRIS, INSA Lyon.

[65] Jalem R,Chauchat JH, 2002, *Pour quoi les n-grammes permettent de classer des textes? Recherche de mots clefs pertinents à l'aide des n-grammes caractéristiques. 6es journées internationales d'analyse statistique des données textuelles*, Laboratoire ERIC. Université Lyon 2 (F).

[66] Denoyer L, 2004, *Apprentissage et inférence statistique dans les bases de documents structurés : application aux corpus de documents textuels*. Thèse de doctorat de l'Université Paris 6.

[67] V.Rijsbergen C.J, 1980, *Information Retrieval*, Dept of Computing Science, University of Glasgow.

[68] Jaillet S Teisseire M Dray G, 2003, *Adéquation des modèles de représentation aux méthodes de catégorisation* .LIRMM-CNRS-ISIM-Université Montpellier.

[69] Fouad Dahak, 2006, *Indexation des documents semi-structurés*, Mémoire de Magister, INI, oued Smar.Alger.

[70] Louise QUOIRIN, 2007, *La vengeance de Roger* extrait n°10, Master 2 LID parcours image

[71] Bellot P, 2000, *Méthodes de classification et de segmentation locales non supervisées pour la recherche documentaire*, Thèse de doctorat, Université d'Avignon.

[72] Salton G, 1983, *Introduction to Modern Information Retrieval*, McGraw-Hill book company.

[73] Benzécri J.-P. & al, 1973, *La taxinomie, Vol. (1); L'analyse des correspondances, Vol. (2)*, Dunod, Paris.

[74] Kerbaol M. et Bansard J.-Y, 1999, *Pratique de l'analyse des données textuelles en bibliographie*; Ecole MODULAD SFdS, INRIA, Bases de données et statistiques.

[75] Pouliquen Bruno, Delamarre Denis, Le Beux Pierre, 2002, *Indexation de textes médicaux par extraction de concepts, et ses utilisations,* JADT 2002 : 6es Journées internationales d'Analyse statistique des Données Textuelles.

[76] Berry, M. W,S.T.Dumais & T.A.Letsche, 1995, *Computational methods for intelligent information access.* Proc. of supercomputing '95, Sandiego.

[77] Zipf, G.K, 1949, *Human behavior and the principle of least effort*. Addison Wesley.

[78] Adeline Nazarenko, 2007, *Fouille de textes Méthodes et enjeux* Déjeuner de la technologie Laboratoire d'Informatique de Paris-Nord, Université Paris 13 & CNRS.

[79] François Jacquenet et al, 2003, *Veille Technologique assistée par la Fouille de Textes*, Université Jean Monnet, Saint-Etienne EURISE.

[80] Fatma Nasser al Shamsi, 2007, *Statistical arabic information extraction system*, Department of computer science college of arts and sciences, University of Sharjah.

[81] Luc Grivel, 2006, *Outils de classification et de catégorisation pour la fouille de textes*, Equipe ISIS, Université de Marne-La-Vallée.

[82] Radwan JALAM, 2003, *Apprentissage automatique et catégorisation de textes multilingues*, Thèse de doctorat, Université Lumière Lyon2.

[83] Zighed, D.A. and Rakotomalala R., 2000, *Graphes d'induction*. Apprentissage et Data Mining. Hermes Science Publication, Paris.

[84] Grivel L, 2006, *Comment faire face à l'explosion des volumes d'information ? le text mining et ses applications à l'intelligence économique, la gestion de la relation client et la gestion de connaissances*, in Recherche, Technologie et société n 62, Revue trimestrielle du réseau ECRIN pp 12-14.

[85] Grivel L, Guillemin-Lanne S, Lautier C, Mari A, 2001, *La construction de composants de connaissance pour l'extraction et le filtrage de l'information sur les réseaux* , ISKO, université Paris X Nanterre, p 197- 208

[86] Grivel L, Guillemin-Lanne S, Coupet P, Huot C, 2001, *Analyse en ligne de l'information : une approche permettant l'extraction d'informations stratégiques basés sur la construction de composants de connaissance*, VSST , Barcelone, p 149-161.

[87] Grivel L , *Customer feedback and opinion surveys analysis in the automotive industry*, ch. 13 in Text mining and its applications to intelligence, CRM and knowledge management, series : management information systems volume 9, p 327, WIT press

[88] Lebart, L, 2001, *Classification of textual data*. Séminaire de recherche à the School of Computer Science and Information Systems, Birkbeck College.

[89] Kodratoff, Y, 2001, *Machine Learning and Its Application*s, in Comparing Machine Learning and Knowledge Discovery in DataBases : An Application to Knowledge Discovery in Texts, pages 1–21. Springer Verlag LNAI 2049.

[90] Mohamed Abdel Fattah, Fuji Ren & Shingo Kuroiwa, 2005, *Stemming to improve translation lexicon creation form bitexts,* Information Processing and Management 42 (2006) 1003–1016.

[91] Siham Boulaknadel, 2005, *Utilisation des syntagmes nominaux dans un système de recherche d'information en langue arabe*, LINA FRE CNRS 2729- Université de Nantes.

[92] Fuhr, N. and Buckley, C, 1991, *A probabilistic learning approach for document indexing*. In ACM Transactions on Information Systems, volume 9, pages 223–248

[93] Schütze, H., Hull, D. A. & Pedersen, J. O, 1995, *A comparison of classifiers and document representations for the routing problem.* In Fox, E. A., Ingwersen, P. & Fidel, R., editors, Proc. of

SIGIR-95, 18th ACM International Conf. on Research and Development in Information Retrieval, pages 229–237, Seattle, USA. ACM Press, New York, US.

[94] Tzeras, K. and Hartmann, S, 1993, *Automatic indexing based on Bayesian inference networks*. In Korfhage, R., Rasmussen, E. & Willett, P., editors, Proc. of SIGIR-93, 16th ACM International Conference on Research and Development in Information Retrieval, pages 22–34, Pittsburgh, US. ACM Press, New York, US.

[95] Gaizauskas R, Wilks Y, 1998, *Information extraction : beyond document retrieval*, Journal of documentation 54(1) : 70-105

[96] Grishman R, 1997, *Information extraction : techniques and challenges*. In Pazienza, M. T.editor, information extraction : a multidisciplinary approach to an emerging information technology, Frascati, Italie, LNAI tutorial, Springer

[97] Jacquemin C & Bourigault D, 2003, *Term extraction and automatic indexing*. In R.Mitkov, editor hand book of computational linguistics, pages 599-615, Oxford university press, Oxford.

[98] Pazienza M,T. *Information extraction and surroundings*, ch. 2 in Text mining and its applications to intelligence, CRM and knowledge management, series : management information systems volume 9, p 327, WIT press.

[99] Wilks Y, 1997, *Information extraction as a core language technology*. In information extraction : a multidisciplinary approach to an emerging information technology, ed M.T Pazienza, Frascati, Italie, LNAI tutorial, Springer verlag, p 14-18

[100] Aas, K. and Eikvil, L, 1999, *Text categorization : a survey*. Technical report, Norwegian Computing Center.

[101] Yang, Y, 1999, *An evaluation of statistical approaches to text categorization.* Information Retrieval, 1(1/2) :69–90.

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

[103] de Loupy, C, 2001, *L'apport de connaissances linguistiques en recherche documentaire*. In TALN'01.

[104] Cavnar W. B. and Trenkle, J. M, 1994, *N-gram-based text categorization*. In Proc. of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, pages 161–175, Las Vegas, US

[105] Guillaume Cleuziou, sd, *Regroupements non-disjoints de mots pour la classification de documents,* LIFO, Laboratoire d'Informatique Fondamentale d'Orléans Université d'Orléans

[106] C.H.A. Koster and M. Seutter, 2003, *Taming Wild Phrases*, in Proc. ECIR.

[107] Larkey, L. S, 1998, *Some issues in the automatic classification of US patents*. AAAi-98 working notes.

[108] Dumais, S., Platt, J., Heckerman, D. & Sahami, M, 1998, *Inductive learning algorithms and representations for text categorization.* In Gardarin G., French J. C., Pissinou N., Makki K. & Bouganim L., editors, Proc. of CIKM-98, 7th ACM International Conference on Information and Knowledge Management, pp 148–155, Bethesda, US. ACM Press, New York, US.

[109] Louwerse M. et van Peer W, 2002, *Thematics: Interdisciplinary Studies*. John Benjamins Publishing Company.

[110] Lewis, D. D., 1992 b, *Representation and learning in information retrieval.* PhD thesis, Dept of Computer Science, University of Massachusetts, Amherst, US.

[111] Luhn H. P., 1958, *The automatic creation of literature abstracts*, IBM Journal of Research and Development, vol. 2, p. 159-168.

[112] Annabelle Mercier, Amélie Imafouo, Michel Beigbeder, sd, *Modèle de proximité : Conception et comparaison à une méthode de recherche de passages*, Ecole Nationale Supérieure des Mines de Saint-Etienne.

[113] Buckley, C., Salton, G., Allan, J. & Singhal, A, 1994, *Automatic query expansion using SMART : TREC 3*. In Text REtrieval Conference.

[114] Dumais, S, 1991, *Improving the retrieval of information from external sources. Behavior Research Methods,* Instruments & Computers, 23(2) :229–236.

[115] Deerwester, S., Dumais, S., Landauer, T., Furnas, G. & Harshman, R, 1990, *Indexing by latent semantic anlysis*. Journal of the American Society of Information science, 416(6) :391–407.

[116] Dumais S.T., 1992, *Enhancing Performance in Latent Semantic Indexing (LSI) Retrieval*, Technical Memorandum Tm-ARH-017527, Bellcore.

[117] Fano R., 1961, *Transmission of Information : A Statistical Theory of Communication,* MIT Press and John Willey & Sons.

[118] Church K., Hanks P,1989, *Word association norms, mutual information and lexicography*, Actes de la 27e Annual Conference of the Association of Computational Linguistics ACL'89, pp.76-82.

[119] Sneath P., Sokal R., 1973, Numerical taxonomy. *The principles and practice of numerical classification*, San Francisco, W.H. Freeman.

[120] Greffenstette G., 1994, *Exploration in Automatic Thesaurus Discovery,* Londres, Kluwer Academic Publishers.

[121] Baker L., McCallum A, 1998, *Distributional clustering of words for text classification*, Actes de la 21e International Conference on Research and Development in Information Retrieval SIGIR'98, p. 96-103.

[122] Clarke C. L. A., Cormack G. V., Tudhope E. A, 2000, *Relevance ranking for one to three term queries*, Information Processing and Management, vol. 36, p. 291- 311.

[123] Rasolofo Y., Savoy J, 2003, *Term Proximity Scoring for Keyword-based Retrieval Systems*, ECIR 2003 Proc., p. 207–218.

[124] Sebastiani F, 2005, *Text categoriZation*, ch. 4 in Text mining and its applications to intelligence, CRM and knowledge management, series : management information systems, vol. 9, p 327, WIT press.

[125] Yang Y., Pedersen J., 1997, *A Comparative Study on Feature Selection in Text Categorization*, Actes de la 14e International Conference on Machine Learning ICML'97, p. 412-420.

[126] Liu H., Setiono R., 1995, *Chi2: Feature selection and discretization of numeric attributes*, Actes de la 7e International Conference on Tools with Artificial Intelligence ICTAI'95, USA, pp 388-391.

[127] Sebastiani F., 2002, *Machine learning in automated text categorization*, ACM Computing Surveys, Vol 34 no 1, 2002, pp. 1-47.

[128] Maron, M, 1961, *Automatic indexing : an experimental inquiry.* Journal of the Association for Computing Machinery, 8(3) :404–417.

[129] Forsyth, R. S, 1999, *New directions in text categorization*. In Gammerman, A., editor, Causal models and intelligent data management, pages 151–185. Springer Verlag, Heidelberg, DE.

[130] Teytaud, O. and Jalam, R, 2001, *Kernel based text categorization*. In Proceeding of IJCNN-01, 12th International Joint Conference on Neural Networks, Washington, US. IEEE Computer Society Press, Los Alamitos, US.

[131] Laurent Pierron, Coskun Durkal et Jean-Baptiste Chevalier, 2005, *Classification, combinaison et regroupements pour séparer les discours de Mitterrand de ceux de Chirac*, Atelier DEFT'05, TALN 2005, Dourdan.

[132] Sable, C. L. and Hatzivassiloglou, V, 2000, *Text-based approaches for non-topical image categorization.* International Journal of Digital Libraries, 3(3) :261–275.

[133] Androutsopoulos, I., Koutsias, J., Chandrinos, K. V. & Spyropoulos, C. D, 2000, *An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages.* In Belkin, N. J., Ingwersen, P. & Leong, M.-K., editors, Proc. of SIGIR-00, 23rd

ACM International Conference on Research and Development in Information Retrieval, pages 160–167, Athens, GR. ACM Press, New York, US.

[134] Cohen, W. W., 1996, *Learning rules that classify e-mail*. In The 1996 AAAI Spring Symposium on Machine Learning in Information Access, pages 18–25.

[135] Liddy, E. D., Paik, W. & Yu, E. S, 1994, *Text categorization for multiple users based on semantic features from a machine-readable dictionary*. ACM Transactions on Information Systems, 12(3) :278–295.

[136] Lefèvre, P, 2000, *La recherche d'information - du texte intégral au thésaurus*. Hermès Science, Paris.

[137] Uren, V, 2000, *An evaluation of text categorisation errors*. In Proc. of the One-day Workshop on Evaluation of Information Management Systems, pages 79–87, London, UK. Queen Mary and Westfield College.

[138] Sebastiani, F, 1999, *A tutorial on automated text categorisation.* In Amandi, A. and Zunino, R., editors, Proc. of ASAI-99, 1st Argentinian Symposium on Artificial Intelligence, pages 7–35, Buenos Aires, AR

[139] Lewis, D. D., Schapire, R. E., Callan, J. P., & Papka, R, 1996, *Training algorithms for linear text classifiers.* In Proc. of the 19th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR'96) (pp. 298–306).

[140] Littlestone N., 1988, *Learning quickly when irrelevant attributes abound : A new linear threshold algorithm*. Machine Learning, 2, pp. 285-318.

[141] I. Dagan, Y. Karov, D. Roth, 1997, *Mistake-Driven Learning in Text Categorization.* Proc. of the Second Conference on Empirical Methods in NLP, pp. 5563.

[142] Lebart, L. & Salem, A. (1994). *Statisque textuelle*. Dunod, Paris.

[143] Hull, D. A, 1994, *Improving text retrieval for the routing problem using latent semantic indexing*. In Croft, W. B. and van Rijsbergen, C. J., editors, Proc. of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval, pages 282–289, Dublin, IE. Springer Verlag, Heidelberg, DE.

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

[145] Stricker, M, 2000, *Réseaux de neurones pour le traitement automatique du langage : conception et réalisation de filtres d'information*. PhD thesis, Université Pierre et Marie Curie - Paris VI, Paris.

[146] Yang, Y. and Chute, C. G, 1994, *An example-based mapping method for text categorization and retrieval*. ACM Transactions on Information Systems, 12(3) :252–277.

[147] Yang, Y. and Liu, X, 1999, *A re-examination of text categorization methods*. In Hearst, M. A., Gey, F. & Tong, R., editors, Proc. of SIGIR-99, 22nd ACM International Conference on Research and Development in Information Retrieval, pages 42–49, Berkeley, US. ACM Press, New York, US.

[148] Lewis, D. D. and Ringuette, M, 1994, *A comparison of two learning algorithms for text categorization*. In Proc. of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, pages 81–93, Las Vegas, US.

[149] Apté, C., Damerau, F. J. & Weiss, S. M, 1994, *Automated learning of decision rules for text categorization*. ACM Transactions on Information Systems, 12(3) :233–251.

[150] Borko, H. and Bernick, M, 1964, *Automatic document classification. part ii : additional experiments.* Journal of the Association for Computing Machinery, 11(2) :138–151.

[151] Lewis, D. D, 1998, Naive (Bayes) at forty : The independence assumption in information retrieval. In Nédellec, C. & Rouveirol, C., editors, Proc. of ECML-98, 10th European Conference on Machine Learning, pages 4–15, Chemnitz, (D). Springer Verlag, Heidelberg (D).

[152] Chai, K. M., Ng, H. T. & Chieu, H. L, 2002, *Bayesian online classifiers for text classification and filtering*. In Beaulieu, M., Baeza-Yates, R., Myaeng, S. H. & Järvelin, K., editors, Proc. of SIGIR-02, 25th ACM Intl Conference on Research and Development in Information Retrieval, pages 97–104, Tampere, FI. ACM Press, New York, US.

[153] Adam, C. K., Ng, H. T. & Chieu, H. L, 2002, *Bayesian online classifiers for text classification and filtering*. In Beaulieu, M., Baeza-Yates, R., Myaeng, S. H. & Järvelin, K., editors, Proc. of SIGIR-02, 25th ACM International Conference on Research and Development in Information Retrieval, pages 97–104, Tampere, FI. ACM Press, New York, US.

[154] T. Joachims, 1998, *Text categorization with support vector machines : Learning with many relevant features*. European Conference on Machine Learning (ECML-98), Springer Verlag, pp. 137-142.

[155] Joachims, T, 1999, *Transductive inference for text classification using support vector machines*. In Bratko, I. and Dzeroski, S., editors, Proc. of ICML-99, 16th International Conference on Machine Learning, pages 200–209, Bled, SL. Morgan Kaufmann Publishers, San Francisco, US.

[156] Joachims, T, 2000, *Estimating the generalization performance of a SVM efficiently*. In Langley, P., editor, Proc. of ICML- 00, 17th International Conference on Machine Learning, pages 431–438, Stanford, US. Morgan Kaufmann Publishers, San Francisco, US.

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

[158] Schapire, R. E., Singer, Y. & Singhal, A, 1998, *Boosting and Rocchio applied to text filtering*. In Croft, W. B., Moffat, A., van Rijsbergen, C. J., Wilkinson, R. & Zobel, J., editors, Proc. of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval, pages 215–223, Melbourne, AU. ACM Press, New York, US.

[159] Schapire, R. E. & Singer, Y, 2000, *BOOSTEXTER : a boosting-based system for text categorization*. Machine Learning, 39(2/3) :135– 168.

[160] Carreras X. & Márquez L, 2001, *Boosting trees for anti-spam email filtering*. In Proc. of RANLP-2001, 4th Intl Conference on Recent Advances in Natural Language Processing.

[161] Yang Y, 1997, A*n evaluation of statistical approach to text categorization*. Technichal Report CMU-CS-97-127, Carnegie Mellon University.

[162] Mitchell, T. M, 1997, *Machine Learning*. Computer Science. McGraw-Hill, New York.

[163] Fuhr, N., Hartmann, S., Knorz, G., Lustig, G., Schwantner, M. & Tzeras, K, 1991, *AIR/X – a rule-based multistage indexing system for large subject fields*. In Lichnerowicz, A., editor, Proc. of RIAO-91, 3rd International Conference "Recherche d'Information Assistee par Ordinateur", pages 606–623, Barcelona, ES. Elsevier Science Publishers, Amsterdam, NL.

[164] Quinlan, J, 1986, *Induction of decision trees*. Machine Learning, 1(1) :81–106.

[165] Li, Y. H. and Jain, A. K, 1998, *Classification of text documents*. The Computer Journal, 41(8) :537–546.

[166] Balpe, J. P., Lelu, A., Papy, F. & Saleh I., 1996, *Techniques avancées pour l'hypertexte*. Hermès, Paris.

[167] Juan-Manuel Torres-Moreno, Patricia Veláquez-Morales & Jean-Guy Meunier,1999, *Classphères : un réseau incrémental pour l'apprentissage non supervisé appliqué à la classification de textes*, Université du Québec à Montréal.

[168] Dominic Forest, Jean-Guy Meunier, 2004, *Classification et catégorisation automatiques: application à l'analyse thématique des données textuelles*, JADT 2004 : 7es Journées internationales d'Analyse statistique des Données Textuelles.

[169] Seffah, A. & J. G. Aladin, 1996, *Un atelier de génie logiciel orienté objets pour l' analyse cognitive de textes*. Rapport technique, LANCI-UQAM.

[170] Gabi, K, 1997, *Extraction Dynamique de Connaissances à partir de Textes par Réseaux Neuronaux.* Rapport de DEA en Sciences Cognitives, INPG, Grenoble (F).

[171] Robinson G, 2003, *A statistical approach to the spam problem*, Linux journal

[172] Sebastiani, F., Sperduti, A. & Valdambrini, N, 2000, *An improved boosting algorithm and its application to automated text categorization*. In Agah, A., Callan, J. & Rundensteiner, E., editors, Proc. of CIKM-00, 9th ACM International Conference on Information and Knowledge Management, pages 78–85, McLean, US. ACM Press, New York, US.

[173] Rocchio J.J., 1971, *Relevance feedback in Information Retrieval*, In : Salton, G. (ed.), The Smart Retrieval system - experiments in automatic document processing, Prentice-Hall, Englewood Cliffs, NJ, pp 313-323.

[174] Thomat Ault & Yiming Yang, 2001, *kNN, Rocchio and Metrics for Information Filtering at TREC-10*, TREC-10 Notes.

[175] Fall C.J., A. Törcsvarib A., P. Fiévet P. & Karetkac G., 2004, *Automated categorization of German-language patent documents*, Expert Systems with Applications, vol.26, p 269–277.

[176] Jouadi W, Benghezala H, Zrigui M, 2007, *La distance intertextuelle pour la classification de textes en langue arabe*, CITALA 2007, Rabat, Maroc.

[177] Shuchuan Lo, 2006, *Web service quality control based on text mining using support vector machine*, Expert Systems with Applications, vol 34, n°1, pp 603-610, Elsevier.

[178] F. Taroni, A. Biedermann, C. Aitken, P. Garbolino, 2004, *A general approach to Bayesian networks for the interpretation of evidence*. Forensic Science International. Vol. 139 No 1, pp. 5-16.

[179] Bessou S, Saadi A, Touahria M, 2007, Un Système d'Indexation et de Recheche des Textes en Arabe (SIRTA), , Séminaire national sur le langage naturel et l'intelligence artificielle, LANIA'2007Chlef/Algérie.