Chapitre III. La Fouille de textes appliquée aux corpus des traditions prophétiques "*Hadith*"*

1. Introduction

L'utilisation généralisée du web a causé une demande sans précédent de systèmes capables de sélectionner, de structurer, et d'extraire les informations textuelles disponibles. Les nouveaux flux d'information et les grandes bases de données qui commencent à être disponibles créent également des besoins nouveaux auxquels les différentes communautés du texte essaient de répondre en adaptant les approches qu'elles ont développées depuis de nombreuses années. Ces communautés sont actuellement en pleine évolution et les frontières traditionnelles qui avaient été dessinées au cours des années sont largement bouleversées. On assiste à l'émergence d'un domaine qui est au confluent de leurs préoccupations et que l'on appellera «l'accès à l'information textuelle». Nos expérimentations de ce chapitre, visent à évaluer notre démarche méthodologique de fouille de textes basée sur le modèle vectoriel pour la réorganisation des résultats retournés par un système de recherche d'information.

Ce chapitre est organisé comme suit: la Section 2 présente une description textuelle du corpus « Sahîh Al-Boukhârî »; Dans la section 3 nous nous intéressons au problème d'extraction d'information dans le corpus de «Sahîh Al-Boukhârî»; la Section 4 présente une vue d'ensemble de l'approche de fouille de textes appliquée à notre corpus prophétique; les expériences et les résultats d'évaluation sont rapportés dans la Section 5 ; finalement la Section 6 conclue ce chapitre.

2. Le Corpus Prophétique « *Al-Jâmi'us-Sahih* » de L'imam *Al-Boukhârî*

L'imam Al-Boukhârî est l'auteur de nombreux livres, mais le plus connu est le Sahîh Al-Boukhârî « *Al-Jâmi'us-Sahih* » qui est un recueil de hadîth. Son livre contient 7275 Hadîth avec répétition et environ 2230 sans répétition. Beaucoup de savants musulmans ont essayé de trouver une faille dans cette grande et remarquable collection, mais sans succès. C'est pour cette raison qu'il est établi chez les savants musulmans à l'unanimité que le livre de hadith le plus authentique est *Al-Jâmi'us-Sahih*. Al-Jâmi'us-Sahih a été transmis par une voie double, écrite et orale. *Bedruddin Aynî* et *ibn Hajar* comptent parmi les plus illustres savants qui ont fait l'exégèse de cette œuvre magistrale; ils rapportent trois chaînes de transmissions différentes. Il n'existe entre les sources de l'un et l'autre exégète que d'infimes variations. Al-Jâmi'us-Sahih est, après le Coran, l'œuvre la plus fiable au sujet de l'Islam originel [Wikipedia, 2010].

2.1. Description textuelle du « Sahîh Al-Boukhârî »

En tant que compilation de Hadiths, le corpus « *Sahîh Al-Boukhârî* » est structuré textuellement de la façon suivante:

- Numérotation séquentielle des parties, la première partie (1) c'est **« بدء الـــوحي »** et la dernière partie (100) c'est **« كتاب التوحيد »**.
- Numérotation séquentielle ascendante des chapitres de chaque partie.

* Une grande partie de ce chapitre a été publiée dans les articles suivants :

1. F.Harrag, A. Hamdi-Cherif, et E. El-Qawameh, Information Retrieval Architecture for "Hadith" Text Mining, *Journal of Digital Information Management*, ISSN 0972-7272, Vol.6, N.6, pp. 449-455, 2008.
2. F.Harrag, A. Hamdi-Cherif, A. S. Al-Salamn et E. El-Qawameh, Evaluating the Effectiveness of VSM model and Topic segmentation in Retrieving Arabic Documents, *International Journal of Computer Systems Science and Engineering*, ISSN 0267-6192, Vol.26, N.1, pp. 55-68, 2011.

- Numérotation séquentielle des Hadiths de chaque chapitre, même si le Hadith est répété, il a un nouveau numéro.
- Les paroles du prophète «paix et salut sur lui» sont mises entre parenthèses ( ).
- les versets du Coran sont mises entre accolades {}.
- La forme [:ر] est mise à la fin s'il s'agit d'un Hadith répété.

La figure 3.1 illustre un exemple typique d'un Hadith du *Sahîh Al-Boukhârî* tel que c'est décrit dans le paragraphe précédent.

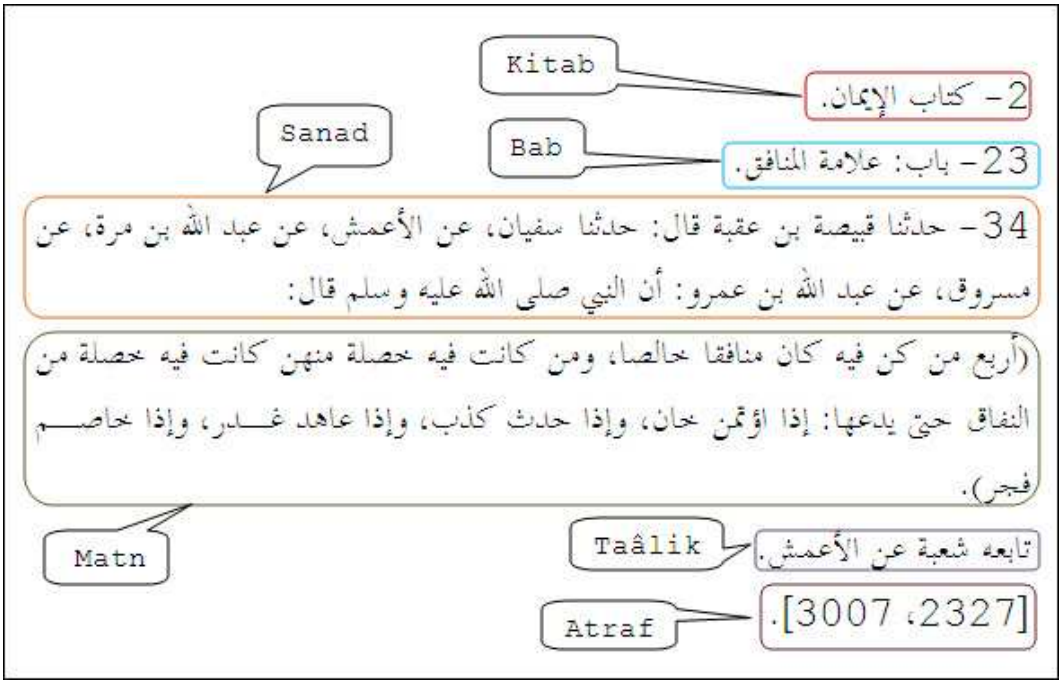

Fig. 3.1. Structure typique d'un Hadith du « Sahîh Al-Boukhârî ».

3. Extraction d'information dans le corpus « Sahîh Al-Boukhârî »

Les domaines de l'Extraction et de la Recherche de l'Information textuelle (respectivement EI et RI) sont l'objet de recherches actives depuis plusieurs années dans la communauté Intelligence Artificielle et fouille de textes. Ce n'est pourtant que récemment et avec l'apparition de grands corpus de données que l'on a ressenti la nécessité d'intégrer dans les systèmes de recherche d'informations existant, des modules d'extraction d'information. Le traitement de grands corpus de données induit des besoins qui se situent à la frontière des domaines d'extraction et de recherche d'information. C'est dans cette optique que se situe le travail que nous présentons dans cette section.

Nous nous intéressons à l'extraction d'informations de surface, i.e. d'informations qui ne demandent pas un traitement linguistique complexe pour être catégorisées. Notre but est de détecter et d'extraire dans des textes, des passages ou des séquences de mots, contenant des informations pertinentes concernant un ensemble de traitements. Nous proposons un système hiérarchique basé sur des techniques d'apprentissage numérique qui résout successivement un problème de compréhension de textes (problématique *EI*). Ce système est testé sur un problème typique d'extraction d'information, il s'agit de la tâche Scenario Templates (patrons d'événements) de MUC[1] qui est devenu une des références dans le domaine.

[1] *Message Understanding Conference*

3.1. Corpus et tâche

Le corpus sur lequel nous avons réalisé des tests est constitué du texte de la collection « *Sahîh Al-Boukhârî* ». La tâche consiste à extraire de ce fichier global, les informations pertinentes pour la structuration conceptuelle de notre corpus (Parties « كتب », Chapitres « أبواب », Chaîne de narration « أسانيد »…etc.). Plus spécifiquement, le système doit détecter les zones de textes pertinentes et leur affecter une étiquette conceptuelle parmi un ensemble fini d'étiquettes: pour chacun des concepts détectés, il s'agit de remplir un patron qui contient une description en plusieurs champs de l'événement (voir Figure 3.2).

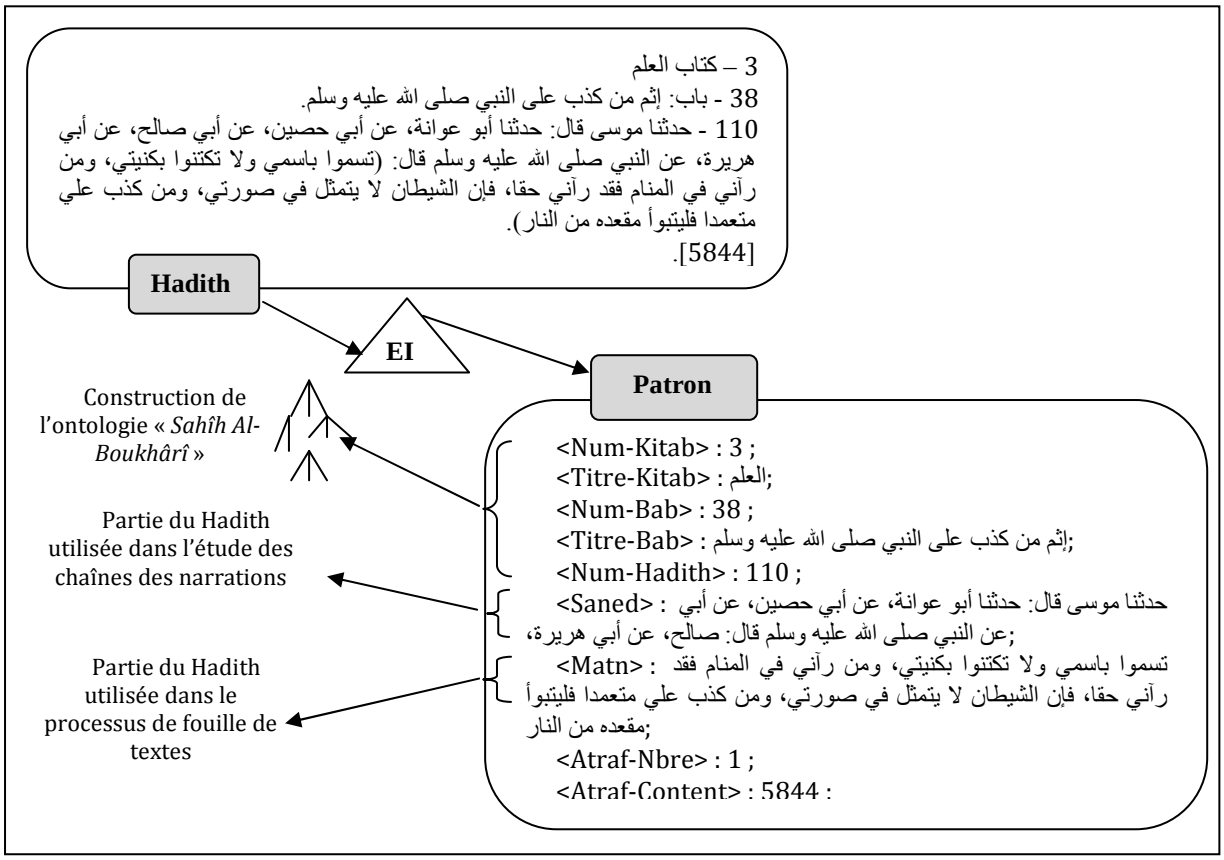

Fig. 3.2. Modèle d'extraction d'informations pertinentes dans le corpus « *Sahîh Al-Boukhârî* ».

Nous avons traité une partie de cette tâche qui consiste à détecter pour chaque hadith les informations du type « *Matn* ». La même démarche a été appliquée pour détecter les informations concernant la plupart des autres champs («*Kitab*», «*Bab* », « *Num-Hadith* », « *Saned* », « *Matn* », « *Taâlik* » et « *Atraf* »). L'approche hiérarchique que nous adoptons pour cette sélection nous permet, d'une part, d'éliminer dès que possible les informations non pertinentes, et d'autre part de raffiner successivement la sélection de variables (ou termes) nécessaires pour détecter l'information pertinente.

3.2. Modèle d'Extraction d'information

L'extraction d'information de type sémantique nécessite un traitement dynamique des mots dans un passage et la prise en compte du contexte. Dans ce but, nous avons utilisé un modèle de production des séquences sous forme d'un automate comme il est mentionné dans la Figure 3.3. Cet automate permettra de passer d'une séquence de vecteurs (ici les mots codés) à une séquence de symboles (ici les concepts que l'on désire extraire) qui correspondra par exemple à la suite d'étiquettes la plus probable pour la séquence. L'automate du modèle d'extraction d'information est définie par un ensemble d'états et de transitions entre ces états, des textes de codage sont associées à chaque état. Dans notre modélisation, les différents états de l'automate

codent les concepts, la structure des transitions code «l'automate de concepts» c'est à dire les transitions entre concepts acceptables.

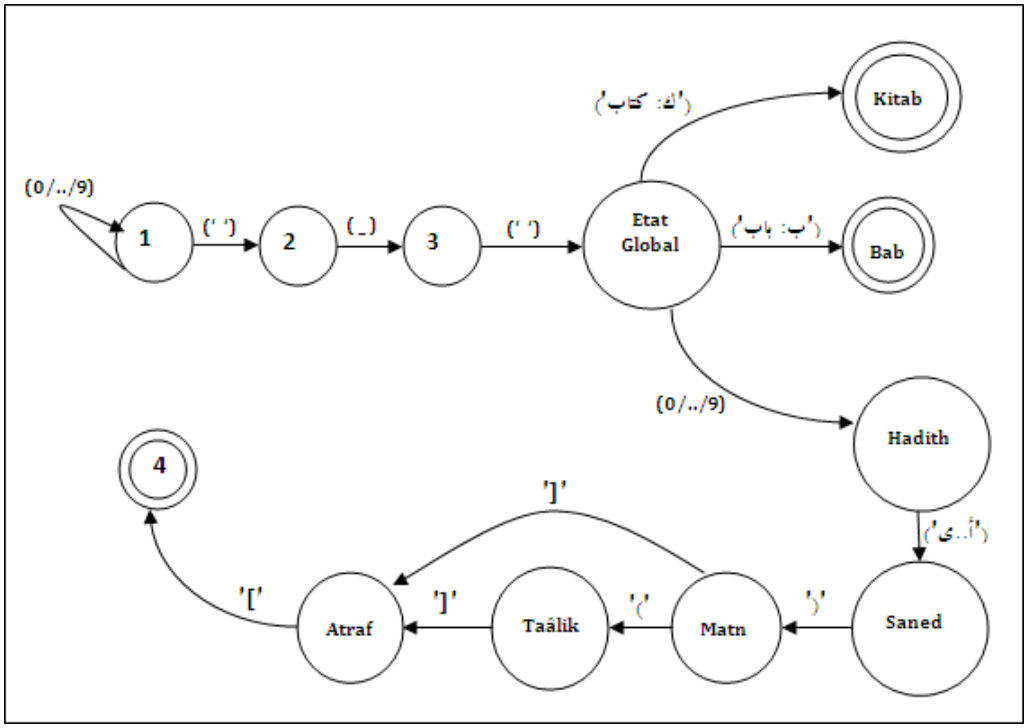

Fig. 3.3. Automate de concepts pour la production des séquences dans le modèle d'extraction d'information dans le corpus « Sahîh Al-Boukhârî ».

L'extraction d'information représente la première étape de notre chaîne de traitement automatique des traditions prophétiques. Elle permettra de passer d'un corpus représenté sous forme d'un fichier texte brut non-structuré à un corpus annoté représenté sous forme d'un fichier texte semi-structuré. Le fichier annoté sera utilisé comme une source de donnés pour alimenter le processus de fouille de textes. L'extraction d'informations de surface permettra de distinguer les différentes parties (champs) du hadith (Kitab, Bab, Sanad, Matn…etc.) sur lequel seront appliqués les différents traitements informatisés. Les grandes étapes de l'algorithme d'extraction d'information sont détaillées ci-dessous :

1. **Etat initial :**

   Si caractère = chiffre ('0..9') *alors* reconnaissance du numéro de chapitre « Kitab » ; Section « Bab » ou Hadith ;
2. **Extraction de l'entité Chapitre « Kitab » :**

   Si caractère = « ك » *alors* reconnaissance de l'entité "*Kitab*" ;
3. **Extraction de l'entité Chapitre « Kitab » :**

   Si caractère = « ب » *alors* reconnaissance de l'entité "*Bab*" ;
4. **Extraction de l'entité Hadith :**

   Si caractère≠ «ك» et caractère ≠ «ب» *alors* reconnaissance de l'entité "*Hadith*" ;
   « Sanad » : *Si* caractère = « (» *alors* reconnaissance de l'entité "*Saned*" ;
   « Matn » : *Si* caractère = «) » *alors* reconnaissance de l'entité "*Matn*";
   « Taâlik » : *Si* caractère = « [» *alors* reconnaissance de l'entité "*Taâlik*" ;
   « Atraf » : *Si* caractère = «] » *alors* reconnaissance de l'entité "*Atraf*";

Algorithme 3.1. Extraction d'information dans le corpus « *Sahîh Al-Boukhârî* ».

4. La fouille de textes appliquée au corpus prophétique

Un processus de fouille de textes ou de recherche documentaire en texte intégral se décompose en trois étapes principales [Bellot, 2000]:

- L'indexation des textes sur lesquels portent les interrogations;
- La recherche proprement dite à partir d'une requête;
- La présentation des résultats de la recherche.

Tout d'abord, le système doit être capable d'indexer la base documentaire dans notre cas c'est le corpus annoté résultant de l'opération précédente et plus spécifiquement les textes des parties "Matn" de l'ensemble des hadiths. Durant cette étape, le logiciel parcourt les textes de la base de prophétique afin d'en relever les termes (noms et verbes) les plus importants. Cela donne lieu à la création d'un index: une liste de termes auxquels sont associés les documents contenant ainsi leur poids dans chacun d'eux. L'étape de recherche se fait ensuite à partir d'une requête exprimée en langage naturel.

Il s'agit de mesurer, grâce aux informations enregistrées dans l'index, la ressemblance sémantique (le score) entre chaque texte du corpus et la requête au moyen de mesure de similarité: *Cosine* qui est décrite plus loin. Pour terminer, les documents trouvés sont ordonnés en fonction de leur score et proposés à l'utilisateur qui est dans ce cas l'expert de hadiths d'où il peut par la suite continuer sa tache de la recherche du degré de véracité du hadith en se basant sur les informations collectées par notre moteur de recherche. Afin d'obtenir des listes de documents de bonne qualité, nous avons choisi d'utiliser une méthode de recherche qui a déjà fait sa preuve dans ce domaine. Pour cette raison, le modèle vectoriel est choisi comme base dans notre système de recherche d'information.

4.1. Le modèle de l'espace vectoriel

Les systèmes de fouille de textes représentent généralement les textes du corpus cible (ainsi que les requêtes) par des mots clés. Ces mots clés sont eux-mêmes habituellement extraits des textes (documents ou requêtes) lors d'une phase d'indexation (Voir Chapitre II). Pour chaque texte, un poids est attribué à chacun des mots clé qu'il contient. Une matrice « documents/mots clé » représente l'ensemble des documents (un vecteur est associé à chaque document, les composantes des vecteurs sont les poids des mots clé) [Salton, 1983]. Le processus habituel de recherche documentaire dans le modèle vectoriel représente la requête par un vecteur dans le même espace que les documents et compare ce vecteur à tous ceux de la matrice. Cette comparaison équivaut au calcul d'une fonction de similarité (ou de distance) entre les vecteurs représentant les documents et le vecteur correspondant à la requête. Elle permet d'ordonner les documents en fonction de leur ressemblance avec la requête comme il est mentionné dans la figure 3.4.

Ce modèle a notamment été critiqué à cause de l'hypothèse d'indépendance des mot-clé (la dimension de l'espace correspond au nombre de mots-clés) [Raghavan et Wong, 1986]. Cependant, malgré sa simplicité apparente, le modèle vectoriel a montré être au moins aussi bon (autant pour la qualité des résultats que pour la rapidité avec laquelle ils sont obtenus) que les autres modèles, lors de la plupart des campagnes d'évaluation. Aussi, c'est le modèle vectoriel que nous avons choisi comme base pour notre système de recherche d'information.

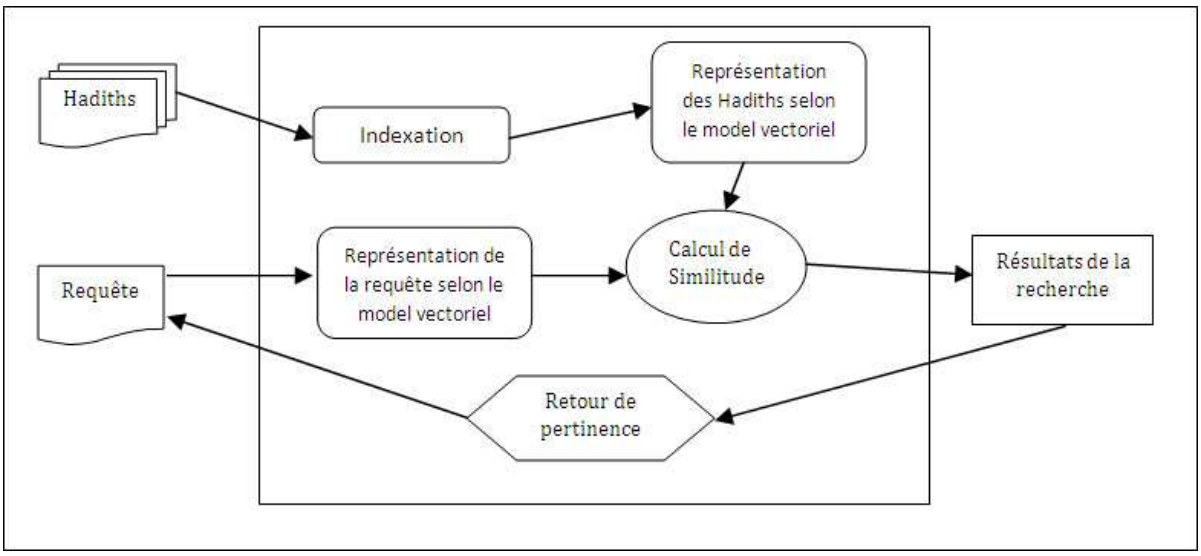

Fig. 3.4. Indexation et recherche de Hadiths dans le modèle vectoriel.

4.2. Un processus de fouille des textes

Les différentes étapes de la fouille des textes dans notre système de recherche d'information sont résumées dans l'Algorithme 3.2. Elles sont décrites dans les sections suivantes.

Indexation du Corpus Prophétique

1. Reconnaissance lexicale des termes arabes ;
2. Elimination des mots vides arabes ;
3. Lemmatisation ;
4. Assignation des poids des termes dans chaque document (Création de la matrice *D-T*) ;

Recherche des Traditions prophétiques (*Hadith*)

1. Appliquer les points 1 à 4 de l'étape indexation aux termes de la requête (sans création de matrice) ;
2. Calcul de la similarité Requête-Hadiths ;
3. Présentation des résultats de recherche.

Algorithme 3.2. Indexation et recherche de *Hadiths.*

Il faut distinguer l'indexation de la base (prétraitement des textes "*Matns*" et détermination de l'importance des termes de chaque texte) de la recherche proprement dite (analyse de la requête et constitution d'une liste de réponses).

4.3. Prétraitement des textes prophétiques

La représentation de textes la plus simple a été introduite dans le cadre du modèle vectoriel présenté ci-dessus, et porte le nom de « sac de mots ». L'idée est de transformer les textes en vecteurs dont chaque composante représente un mot. Ceci nécessite un pré-traitement linguistique. Les composantes du vecteur sont une fonction de l'occurrence des mots dans le texte. Cette représentation des textes exclut toute analyse grammaticale et toute notion de distance entre les mots : c'est pourquoi cette représentation est appelée « sac de mots » [Jalam 2003]. S'attaquer au problème de la recherche d'information dans les corpus prophétiques signifie aussi s'attaquer à des difficultés qui sont propres au traitement automatique de la langue arabe.

4.3.1. Langue arabe et morphologie

L'arabe est la langue mère d'environ 300 millions de personnes dans le monde arabe. Elle est aussi la langue du Coran pour plus d'un milliard deux cents millions de musulmans dans le monde entier. Si nous intéressons aux propriétés morphologiques et syntaxiques, la langue arabe est considérée comme une langue difficile à maîtriser dans le domaine du traitement automatique de la langue, [Larkey et al. 2002]. Elle a une morphologie beaucoup plus riche que l'anglais. La plupart des mots arabes sont dérivée du modèle *fa'ala* (فعل). Tous les mots qui suivent le même modèle ont des propriétés communes, par exemples le modèle *faa'el* (فاعل) indique le sujet du verbe, le modèle *maf'ool* (مفعول) représente l'objet du verbe. Le tableau 3.1 représente les différentes dérivations du mot racine kataba (كتب), leurs modèles, leurs prononciations et leurs traductions en français pour montrer l'effet de ces dérivations sur la signification. Les lettres qui ont été ajoutées à la racine principale du mot (كتب) sont soulignées [Syiam et al. 2006].

Mot arabe	Modèle	prononciation	Sens français
كتب	Fa'ala (فعل)	kataba	Ecrire
كتابة	Fa'ala (فعالة)	kitaba	Ecriture
كاتب	Fa'el (فاعل)	katib	Auteur
مكتوب	Maf'ool (مفعول)	maktoob	Ecrit
كتاب	F'aal (فعال)	kitab	Ouvrage
مكتبة	Maf'ala (مفعلة)	maktaba	Bibliothèque
مكتب	Maf'al (مفعل)	maktab	Bureau

Tableau 3.1. Les différentes dérivations du mot (kataba, كتب)

4.3.2. Normalisation des lettres

Pour gérer les variations dans la représentation des scriptes arabes, nous appliquons plusieurs types de normalisation sur les textes des hadiths et des requêtes. Les requêtes seront normalisées avant qu'elles ne soient soumises au moteur de recherche. Dans notre système, la normalisation est représentée par les étapes suivantes:

- Conversion des textes dans le codage arabe (CP1256), si nécessaire.
- Elimination des ponctuations.
- Elimination des voyelles.
- Elimination des non-lettres (chiffres, caractères spéciaux,…etc.).

4.3.3. Elimination des mots vides

Une bonne partie des chercheurs débutent le processus d'indexation par la suppression de mots vides de sens. Cette méthode nécessite l'utilisation d'une liste de ces mots qui ne contiennent aucune information sémantique et qui ne modifient pas le sens des mots qui les accompagnent. Par exemple, en arabe, on peut compter des mots comme «من», «إلى», «عن», «على». Il s'agit souvent de mots fonctionnels comme les prépositions. Évidemment, le contenu de cette liste dépend de la langue et possiblement du domaine des textes à rechercher. Cette étape, habituellement préalable à d'autres méthodes de sélection d'attributs, permet d'éliminer efficacement plusieurs mots inutiles à la tâche de recherche d'information.

4.3.4. Stemming et Lemmatisation

La recherche de radical «Stemming» et la lemmatisation sont d'autres processus utilisés par certains chercheurs pour créer un vocabulaire réduit. Leur but est de regrouper en un seul attribut les multiples formes morphologiques de mots qui ont une sémantique commune. Par exemple, «يدرس», «مدّرس» et «دراسة» pourraient être regroupés, et un seul attribut serait ajouté à l'espace vectoriel plutôt que trois. Ce faisant, la dimension du vocabulaire s'en trouverait réduite

et on aurait capté la sémantique commune d'une famille de mots. Plusieurs études dans le domaine de la recherche d'information arabe ont montré que l'usage des racines et des lemmes arabes comme termes d'indexation peut substantiellement améliorer l'efficacité de la recherche par rapport à l'usage des mots simple [Al-Kharashi et Evens, 1994] [Hmeidi et al., 1997] [Abu-Salem et al., 1999]. Les analyseurs morphologiques à grande échelle fournissent plus d'information que la racine seule d'un mot. Ils peuvent fournir des informations telles que la signification des préfixes et des suffixes et la désambiguïsation des racines [Beesley, 1996] [Attia, 2000].

a. Processus de lemmatisation

Notre processus de lemmatisation est basé sur la troncature d'un ensemble restreint d'affixes, il ne s'appuie pas sur des règles de dépendances syntaxiques mais plutôt sur des règles morphologiques. Nous disposons de deux ressources: une base des affixes et un dictionnaire de lemmes. Nous testons si un mot appartient au dictionnaire sinon nous opérons à une troncature d'affixe et nous ajoutons le mot et son lemme au dictionnaire. L'idée de cette lemmatisation est de tronquer quelques préfixes qui ne sont rien d'autres que des prépositions attachées aux mots, et quelques suffixes, étant généralement des pronoms accordés à la fin des mots. Pour ce faire, nous avons regroupé ces affixes dans 2 grandes classes : préfixes et suffixes.

Les préfixes sont: (و, أو, ال, ب, ل, ول, لل, بال, وب, ول, فال, كال, ولل, فل, وبال, فب).

Les suffixes sont: (ا, ه, ن, ي, ت, ها, ين, و, ان, ات, ون, هم, نا, ك, وا, هما, تي).

Le dictionnaire de lemmes est un objet qui mappe une table de la base de données dont chaque entrée correspond à une ligne du dictionnaire de la langue. Il contient environ 227.994 mots différents de la langue arabe reliés chacun à un des 3359 lemmes c'est-à-dire entrées uniques d'un dictionnaire classique, Chaque ligne est constituée d'un mot et d'un lemme. Les différentes étapes de notre processus de lemmatisation sont décrites dans l'algorithme suivant:

1. Recherche du mot dans le dictionnaire Mot-Lemme:

Appliquer une routine de recherche dans le dictionnaire Mot-Lemme ;
Si le mot existe alors aller à **3** ;
Sinon aller à **2** ;

2. Appliquer le processus de lemmatisation :

Pour chaque mot **faire**

- Normaliser le mot.
- **Si** le mot est un mot vide alors aller à **3** ;
- **Sinon** Enlever les préfixes et les suffixes.

3. Fin.

Algorithme 3.3. Processus de Lemmatisation.

b. Exemple applicatif

Le texte à lemmatiser est le matn du Hadith N=° 16 du chapitre N=° 8 «La douceur de la foi » de la partie N=° 2 « La Foi » du livre « Sahîh Al-Boukhârî »:

﴿ ثلاث من كن فيه وجد حلاوة الإيمان: أن يكون الله ورسوله أحب إليه مما سواهما، وأن يحب المرء لا يحبه إلا لله، وأن يكره أن يعود في الكفر كما يكره أن يقذف في النار ﴾

La liste des mots vides à éliminer est la suivante: {من، في، أن، إلى،ما، سوى، لا، إلا، كما}

La liste des préfixes à tronquer est la suivante: { م، ل، ال، و }

La liste des suffixes à tronquer est la suivante: { ه، هما }

La lemmatisation du Matn du hadith a donné le texte suivant:

﴿ ثلث، كان، وجد، حلا، أمن، الله، رسول، أحب، مرء، كره، عاد، كفر، قذف، نور ﴾

4.4. Indexation des textes prophétiques

Une fois ces pré-traitements effectués sur l'intégralité des textes "Matns" du corpus hadiths, l'indexation proprement dite peut commencer. Cette étape consiste à relever les termes les plus significatifs à partir des textes « Matns » des hadiths afin d'établir un lien entre chaque terme (les entrées de l'index) et les Matns qui les contiennent (la structure de données utilisée pour conserver ces liens est décrite la section suivante). Une fois les termes relevés, un poids doit être attribué à chacun d'eux pour sa présence dans chaque Matn. La pondération des termes utilisée est décrite en 4.4.2. La figure 3.5 présente le schéma général du processus d'indexation.

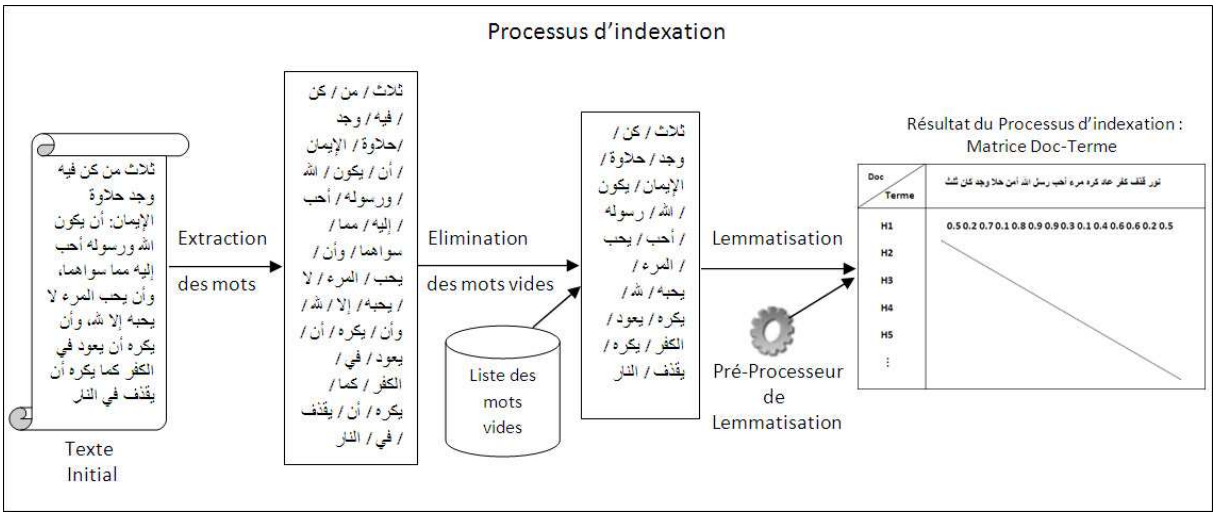

Fig. 3.5. Schéma du processus d'indexation.

4.4.1. Matrice d'indexation (Document-Terme)

La représentation des documents dans le modèle vectoriel conduit à autant de vecteurs que de documents. La taille de ces vecteurs est égale au nombre de termes (mots clé) différents contenus dans la base à indexer. Conserver la matrice des termes telle quelle obligerait à considérer séparément chaque vecteur afin de déterminer quels sont ces documents. En fait, par rapport à l'utilisation d'une matrice inversée (Figure 3.6), cela reviendrait à effectuer, pour chaque nouvelle requête, autant de recherches que ce qu'il y a de documents dans le corpus. La matrice inversée (Doc-Terme) décrit la fréquence des termes qui appartiennent au corpus hadith. Dans cette matrice, chaque terme utilisé dans un texte « matn » de la collection est représenté par une colonne et chaque ligne correspond à un Hadith de la collection. Les éléments a_{ij} de la matrice (Doc-Terme) donnent les fréquences de termes utilisés dans les textes calculées selon la formule TF-IDF décrite dans la section suivante. Notre système de recherche d'information crée la matrice (Doc-Terme) du corpus hadith comme indiqué dans l'Algorithme générale 3.2.

	قول	حدث	عبد	امم	ملك	ورب	وقل	...	مقتول	Class	
2	5.83	4.12	5.36	4.97	20.47	11.15	9.23	...	0	الإيمان	
3	0	0	0	0	0	0	0		0	الإيمان	
4	0	0	0	0	0	0	0		0	العلم	
5	0	0	0	4.97	0	0	0		0	الأحوال-الشخصية	
6	0	0	0	0	0	0	0		0	الأحوال-الشخصية	
7	0	0	0	0	0	0	0		0	الأحوال-الشخصية	
8	0	0	0	0	0	0	0		0	الأحوال-الشخصية	
9	0	0	0	0	0	0	0		0	الأحوال-الشخصية	
10	0	0	0	0	0	0	0	...	0	الأحوال-الشخصية	
11	26.25	0	0	0	0	0	13.84		0	الأحوال-الشخصية	
12	0	0	0	0	0	0	0		0	القضية-الأحكام	
13	0	0	0	0	0	0	4.61		0	القضية-الأحكام	
14	0	0	0	0	0	0	0		0	القضية-الأحكام	
15	2.92	0	0	0	0	0	0		0	العلم	
16	2.92	0	0	0	0	0	0	...	0	القضية-الأحكام	
17	0	4.12	0	0	0	0	0		0	القضية-الأحكام	
	⋮					⋮				⋮	

Fig. 3.6. Création de la matrice Doc-Terme.

4.4.2. Pondération des termes

Différentes formulations de la pondération des termes sont proposées dans la littérature (voir par exemple [Salton et Buckley, 1988][Kowalski, 1997]). Seule celle retenue pour notre système, en raison de la qualité dont elle a fait preuve dans de nombreux systèmes, est présentée dans cette section.

La formulation du poids des termes choisie tient compte du nombre d'apparitions du terme dans les hadiths du corpus (ou dans la requête) et du nombre des hadiths qui contiennent ce terme dans tout le corpus (elle est appelée TFIDF ; Term Frequency, Inverse Document Frequency en anglais). Cette formulation fait l'hypothèse qu'un terme est important pour un hadith donné, s'il apparaît souvent dans ce hadith et que peu de hadiths le contiennent [Salton et Buckley, 1988]. Cette pondération est définie de la manière suivante:

$$TFIDF(w,d) = TF_{w,d} \cdot IDF_{w,d} = TF_{w,d} \cdot \left(\left(\log_2 \frac{N}{DF_w} \right) + 1 \right) \qquad (3.1)$$

Avec : w un terme, d un document, $TF_{w,d}$ le nombre d'apparitions de w dans d, DF_w le nombre de hadiths du corpus qui contiennent w et N le nombre total des hadiths du corpus.

Différentes variantes de cette mesure ont été proposées, [Sparck-Jones, 1972] utilise la seule composante IDF tandis que [Croft, 1983], [Harman, 1986] et [Salton et Allan, 1994] combinent IDF avec TF suivant différents facteurs de normalisation. Ils proposent en effet de normaliser TFIDF pour donner une chance identique aux documents quelle que soit leur taille. L'influence des facteurs TF et IDF sur la recherche documentaire est évaluée par exemple dans [Lee et al., 1997].

4.5. Traitement des requêtes

Le traitement et la formulation des requêtes ont fait l'objet de très nombreuses études. Ces études concernent par exemple l'enrichissement des requêtes (query expansion) par l'ajout automatique de termes [Attar et Fraenkel, 1977] grâce à l'utilisation d'un thesaurus [Qiu et Frei, 1993]. L'enrichissement peut également être réalisé par l'utilisateur (manual relevance feedback) ou automatiquement (automatic relevance feedback) en fonction des documents rapportés lors d'une première recherche (cf. [Robertson et Sparck-Jones, 1976]).

Dans notre système, les requêtes sont traitées de la même manière que les hadiths corpus: reconnaissance des unités lexicales, élimination des mots vides et lemmatisation permettent d'aboutir à l'écriture finale des requêtes avant l'interrogation.

4.5.1. Pondération des termes de la requête

Les termes des requêtes sont pondérés de la même manière que les termes d'un document ou d'un hadith indexé (cf. Section 4.4), dans ce cas : TF_w est le nombre d'apparitions de w dans la requête (ce nombre est égale à 1 dans la plupart du temps ce qui conduit à ne considérer que la composante IDF). Cette pondération est motivée par le fait que les requêtes sont exprimées en langue naturelle et que tous les mots qu'elles contiennent ne sont pas d'importance égale.

4.6. Calcul des similarités

Comme cela a déjà été dit, la recherche proprement dite s'effectue en calculant une mesure de similarité entre chaque hadith de la collection et la requête. L'index permet d'obtenir la liste des hadiths qui contiennent les termes de la requête. Les similarités sont calculées en fonction de chaque terme de la requête par la prise en compte du poids des termes dans les hadiths et dans la requête. Dans cette section seule la mesure *Cosine*, qui est implémentée dans notre système, est décrite. D'autres mesures sont proposées et discutées notamment dans [Salton, 1983], [Harman, 1992] ou [Baeza-Yates et Ribeiro-Neto, 1999]. Les propriétés géométriques des principales mesures sont analysées dans [Jones et Furnas, 1987].

4.6.1. La mesure Cosine

Cosine est l'une des mesures de similarité les plus fréquemment utilisées grâce à son bon fonctionnement sur des corpus variés. Elle consiste à calculer les valeurs des cosinus des angles séparant les vecteurs des hadiths et le vecteur de la requête (voir figure 3.7 ci-dessous) [Salton, 1983]. Selon le modèle vectoriel, les hadiths et la requête sont représentés dans le même espace. Par rapport à un simple produit scalaire, cette mesure présente l'avantage de normaliser les scores de chaque hadith en fonction de leur taille, elle-même pondérée par le poids des termes. La mesure cosine est définie comme suit :

$$Cosine(h,r) = \frac{\sum_{w \in h \cap r} TFIDF_{w,h} \cdot TFIDF_{w,r}}{\sqrt{\left(\sum_{w \in h} TFIDF_{w,h}^2\right) \cdot \left(\sum_{w \in r} TFIDF_{w,r}^2\right)}} \qquad (3.2)$$

Avec : w un terme, h un hadith, r la requête, $TFIDF_{w,h}$ le poids de w dans d et $TFIDF_{w,r}$ celui de w dans la requête r.

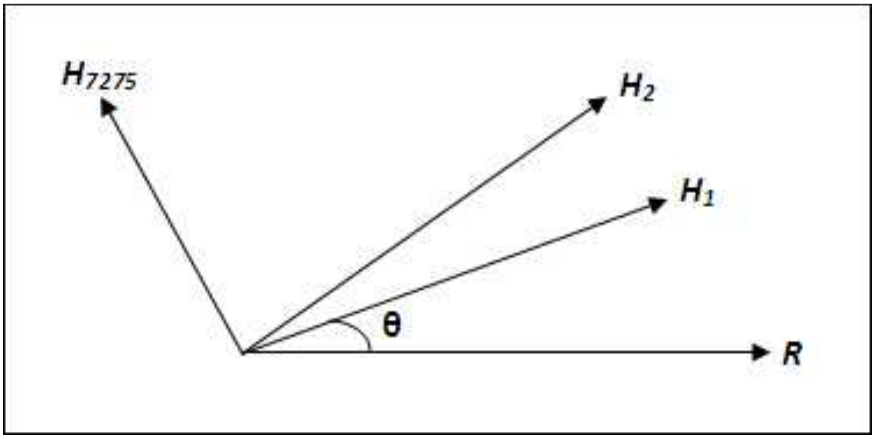

Fig. 3.7. Le cosinus comme mesure de similarité (H_i et R sont respectivement les vecteurs représentant les hadiths et la requête).

4.6.2. Algorithme de recherche utilisant Cosine

L'Algorithme 3.4 est utilisé pour calculer les scores des hadiths du corpus en fonction d'une requête après étiquetage et lemmatisation de cette dernière.

// Algorithme de recherche en utilisant Cosine //

- Pour chaque terme *T* de la requête *R*:
  - Obtenir depuis la matrice (*D-T*), la liste des hadiths contenant T ;
  - Pour chaque hadith *H* de cette liste:
  - Mettre à jour les scores de H en fonction des poids de T dans le hadith et dans la requête:
    - $Score\ (H)\ =\ score\ (H)\ +\ (poids\ (T,H)\ *\ poids\ (T,R))$
  - Mettre à jour les sommes des carrés des poids utilisés dans le calcul du score pour *H* et pour *R* (normalisation des scores) ;
- Normaliser les scores de chaque hadith;
- Ordonner les hadiths en fonction des scores normalisés ;

Algorithme 3.4. Calcul des scores des hadiths selon Cosine en fonction d'une requête

5. Expériences et résultats

L'expérimentation et la validation de notre démarche méthodologique ont été réalisées sur un échantillon du corpus «Sahîh Al-Boukhârî ». Dans les domaines de recherche d'information et du traitement automatique du langage, plusieurs corpus de référence ont été développés durant les dernières années. La collection de corpus la plus fréquemment citée est celle du projet TREC2 [Voorhees et Harman, 2005]. Les corpus de ce projet sont caractérisés par les différents traitements qu'ils permettent d'évaluer. Dans le domaine de recherche d'information dans les collections prophétiques, aucun corpus de référence n'a été développé et ce, malgré l'importance de ces corpus comme une source de législation islamique.

5.1. Etude statistique du corpus « Sahîh Al-Boukhârî»

Ne disposant pas de corpus de référence adapté aux spécificités de notre projet, nous avons donc décidé de construire notre propre corpus d'expérimentation. Nous avons opté pour un corpus composé de textes prophétiques du «Sahîh Al-Boukhârî ». Ce choix est motivé par un facteur très important pour l'évaluation de notre démarche. En effet, chaque texte du « Sahîh » a été manuellement catégorisé par « l'imam Al-Boukhârî ». Nous avons utilisé cette information à des fins d'évaluation des résultats obtenus par le processus de classification thématique. La collection complète du «Sahîh Al-Boukhârî » est disponible sur Internet sous forme d'un seul fichier texte non-structuré organisé selon une taxinomie propre à « l'imam Al-Boukhârî ». Cette taxinomie est constituée de trois niveaux hiérarchiques. Le premier niveau est celui des parties (Koutob ; كتب), le deuxième niveau est celui des chapitres (Abouab ; أبواب) et le troisième niveau est celui des traditions prophétiques (Ahadith ; أحاديث).

Le corpus «Sahîh Al-Boukhârî » est composée de 7275 hadiths et fait plus de 3 Mo en format texte brut. À titre indicatif, mentionnons que le corpus «Sahîh Al-Boukhârî » comporte 596123 occurrences et 38246 formes dont la fréquence varie entre 1 et 31811 occurrences. On y retrouve, entre autres, 16566 hapax (mots dont la fréquence est de 1). Le ratio formes/occurrences est donc de 6,41%; alors que le ratio hapax/formes est de 43,31%.

Après avoir supprimé les marques de ponctuation, les caractères numériques, et toutes les informations qui concernent les parties «الكتب», les chapitres «الأبواب», les chaînes de narrations

[2] trec.nist.gov

«الأسانيد», les commentaires «التعليقات» et les bouts «الأطراف», et après l'application des processus d'élimination des mots vides et de lemmatisation sur le reste de corpus (les textes *Matns* du corpus «متون الأحاديث»), la taille du corpus a été substantiellement réduite. Ainsi, le corpus, nettoyé, est composé de 307127 occurrences et de 1918 formes dont la fréquence varie entre 1 et 32867 occurrences. On y retrouve désormais 204 hapax. Le ratio formes/occurrences est diminué à 0,62%; alors que le ratio hapax/formes est maintenant de 10,63%.

5.1.1. Etude de la fréquence des mots sur le corpus de «Sahîh Al-Boukhari»

a. Enoncé de la loi de Zipf

La distribution de l'occurrence des mots dans un corpus de texte donné n'est pas uniforme, certains mots apparaissent très fréquemment, tandis que d'autres apparaissent très rarement. Les mots les plus fréquents en arabe sont les mots grammaticaux comme من, إلى, عن, على, … . Sur le corpus «Sahîh Al-Boukhari», les dix (10) mots qui apparaissent le plus fréquemment sont: **عن, الله, ان, من, وسلّم, صلّى, عليه, حدّثنا, بن, قال**.

La distribution de fréquence des mots dans un corpus a été étudiée empiriquement par Zipf [Zipf, 1949] et les résultats de cette analyse sont connus sous le nom de loi de Zipf. Pour énoncer cette loi, Zipf est parti d'un principe général qu'il a ensuite énoncé mathématiquement. Si l'on considère un corpus contenant N textes et que l'on note $TF_{w,d}$ l'occurrence d'un mot w dans un texte d, on peut définir TTF_w, l'occurrence totale du mot w sur le corpus C :

$$TTF_w = \sum_{d \in C} TF_{w,d} \qquad (3.3)$$

Si l'on classe ensuite l'ensemble des mots du corpus par ordre décroissant d'occurrence totale, on obtient pour chaque mot un rang r_w. La loi formulée par Zipf s'écrit alors :

$$TTF_w \cdot r_w = K_c \qquad (3.4)$$

K_c est une constante qui dépend du corpus. Cette relation peut s'écrire également :

$$\log r_w = \log r K_c - \log TTF_w \qquad (3.5)$$

Cette dernière relation, montre que si l'on trace le logarithme de l'occurrence en fonction du logarithme du rang, on doit obtenir une droite de pente (–1).

La Figure 3.8 montre la vérification expérimentale de la loi de Zipf sur le corpus «Sahîh Al-Boukhari». On compte 38246 termes différents sur ce corpus ; à partir du rang 1027, les termes ont une fréquence totale sur l'ensemble du corpus inférieure à cinquante (50).

En fait, il est fréquent, comme le montre la figure 3.8, que la loi ne soit pas très bien vérifiée pour les hautes fréquences et les basses fréquences. Cependant, cette loi reflète bien le comportement général de la distribution des occurrences : il existe un petit nombre de mots très fréquents, il existe un grand nombre de mots très rares n'apparaissant qu'une fois ou deux sur le corpus et il existe tout un ensemble de mots dont la fréquence d'apparition se situe entre ces deux domaines.

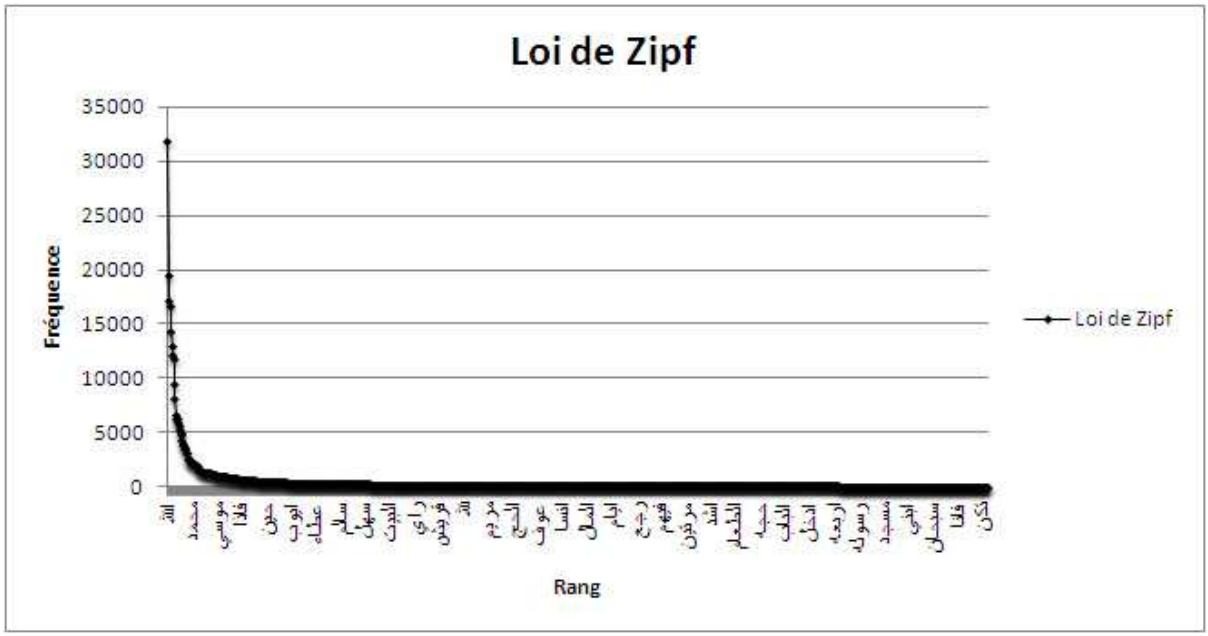

Fig. 3.8. Vérification expérimentale de la loi de Zipf sur le corpus «Sahîh Al-Boukhari».

b. Conséquences de la loi de Zipf

Comme le nombre de mots présents dans un corpus peut être très grand, les méthodes statistiques cherchent, en général, à réduire le nombre de mots utilisés pour représenter les textes. Nous verrons dans la suite comment s'effectue cette opération, mais les observations du paragraphe précédent permettent d'effectuer une première réduction de la dimension de l'espace de descripteurs.

Il s'agit de supprimer les mots dont on sait a priori qu'ils ne seront pas utiles pour les algorithmes d'apprentissage. Cette étape est critique, car les mots supprimés lors de cette étape le sont définitivement et il ne faut donc pas supprimer de mots importants.

La distribution de fréquences décrite par la loi précédente a deux conséquences importantes pour la représentation des textes.

- **Suppression des mots fréquents**

Les mots qui apparaissent le plus souvent dans un corpus sont, comme on l'a vu précédemment, les mots grammaticaux ou les mots de liaisons. Ces mots doivent être supprimés de la représentation des textes pour deux raisons :

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

Comme le nombre de mots concernés est faible, il est possible de définir une liste de mots qui sont automatiquement supprimés de la représentation. Par exemple [Sahami, 1998] définit une liste de 570 mots courant en anglais, plus une liste de 100 mots très fréquents sur le web, pour supprimer les mots les plus courants. Cependant, l'établissement d'une telle liste peut poser des problèmes. D'une part, il n'est pas facile de déterminer le nombre de mots exacts qu'il faut inclure dans cette liste. D'autre part, cette liste est intimement liée à la langue utilisée et n'est donc pas transposable directement à une autre langue.

- **Suppression des mots rares**

En général, les auteurs cherchent également à supprimer les mots rares d'un corpus afin de réduire de façon appréciable la dimension des vecteurs utilisés pour représenter les textes, puisque, d'après la loi de Zipf, ces mots rares sont très nombreux. D'un point de vue linguistique, la suppression de ces mots n'est pas nécessairement justifiée : certains mots peuvent être très rares, mais très informatifs. Néanmoins, ces mots ne peuvent pas être utilisés par des méthodes à bases d'apprentissage du fait de leur très faible occurrence.

Une des méthodes communément retenues pour supprimer ces mots consiste à ne considérer que les mots l'occurrence totale est supérieure à un seuil fixé préalablement.

Nous exposons dans le cinquième chapitre une méthode de détermination du vocabulaire a base de seuil permettant de d'écarter automatiquement les mots rares sans utiliser une liste de mots prédéfinis. Cette méthode présente en outre l'avantage d'être adaptée à la classification.

5.2. Requêtes et corpus de test

Il aurait été pratiquement impossible d'analyser rigoureusement un corpus d'une telle ampleur dans le cadre de notre projet. Nous avons donc limité notre corpus d'expérimentation à 453 hadiths regroupés en 14 catégories de taille différente. Pour constituer notre corpus, nous avons donc effectué 3 requêtes thématiques {$R_1=H_{33}$ et $R_3=H_{129}$ et $R_3=H_{5834}$} afin de récupérer 453 hadith de chaque catégorie. Les catégories retenues, choisies de manière aléatoire à partir des 100 catégories de niveau 1, sont résumés dans le tableau 3.2 ci-dessous :

La Catégorie	Nombre de Documents
La Foi (الإيمان)	33
Le Coran (القرآن)	34
Le Savoir (العلم)	32
Les Crimes (الجنايات)	32
Al Djihad (الجهاد)	34
La Morale (الأخلاق والآداب)	35
Les Générations antérieures (الأمم السابقة)	22
La Bibliographie (السيرة)	21
Les Jugements (الأقضية والأحكام)	34
Les Adorations (العبادات)	33
Les Comportements (المعاملات)	35
L'Alimentation (الأشربة و الأطعمة)	35
Les Vêtements (اللباس والزينة)	39
Les Etats personnelles (الأحوال الشخصية)	34

Tableau 3.2. Descriptif des catégories du corpus Hadith.

La requête R_1 se représente par le « *Matn* » du hadith n° 33 du « Sahîh Al-Boukhârî », les informations du hadith H_{33} sont les suivantes :

2- كتاب الإيمان.
23- باب: علامة المنافق.
33- حدثنا سليمان أبو الربيع قال: حدثنا إسماعيل بن جعفر قال: حدثنا نافع بن مالك بن أبي عامر أبو سهيل، عن أبيه، عن أبي هريرة، عن النبي صلى الله عليه وسلم قال: (**آية المنافق ثلاث: إذا حدث كذب، وإذا وعد أخلف، وإذا اؤتمن خان**).
[2536]، [2598]، [5744].

R_1

Fig. 3.9. Informations du Hadith H_{33} (Requête R_1).

La requête R_2 se représente par la partie « *Matn* » du hadith n=° 129 du « Sahîh Al-Boukhârî », les informations du hadith H_{129} sont les suivantes :

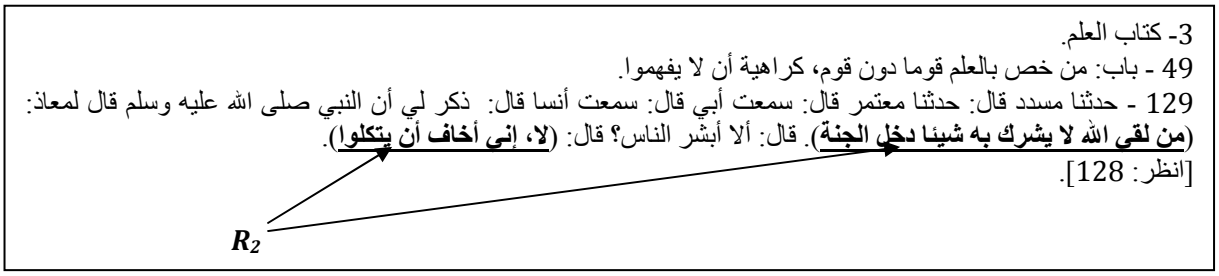

Fig. 3.10. Informations du Hadith H_{129} (Requête R_2).

La requête R_3 se représente par le « *Matn* » du hadith n° 5834 du « Sahîh Al-Boukhârî », les informations du hadith H_{5834} sont les suivantes :

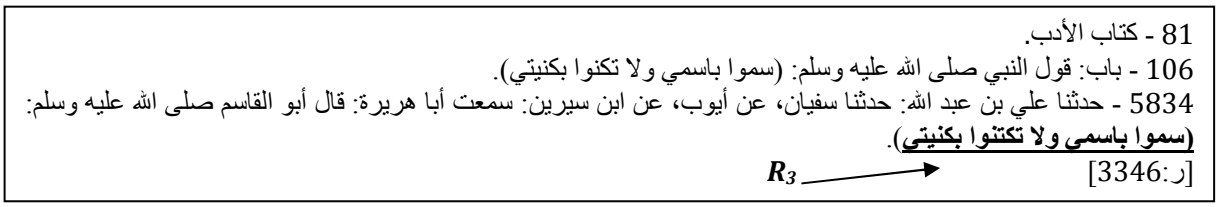

Fig. 3.11. Informations du Hadith H_{5834} (Requête R_3).

5.3. Calcul de poids des termes des requêtes

Le texte ("Matn") du hadith H_{5834} est considéré dans notre cas, comme une requête de test pour le processus de recherche d'information. Les deux racines extraites du texte de la requête R_3 sont : "سمّى" pour "سمّوا ,باسمي " et "كنّى" pour "تكنّوا ,بكنيتي ", ces deux racines n'apparaissent dans la requête que deux fois pour chacun d'eux. Le calcul du poids selon la pondération TFIDF est fait sur un ensemble de 7275 hadiths, la racine "سمّى" apparaisse dans 119 hadiths alors que la racine "كنّى" apparaisse dans 16 hadiths. La racine "كنّى" à un poids de 19.65 malgré qu'elle n'apparaisse que 28 fois dans les 16 hadiths par rapport à la racine "سمّى" qui apparaisse 642 fois dans les 119 hadiths, ceci s'explique par le fait que la mesure TFIDF donne un grand pouvoir discriminatoire aux termes très rares dans la base des hadiths. Le texte ("Matn") du hadith n° 5834 du « Sahîh Al-Boukhârî » est le suivant: ﴿ سموا باسمي ولا تكتنوا بكنيتي ﴾.

$$TFIDF(\text{"كنّى"}, 5834) = TF_{\text{كنّى},5834} \cdot TIDFF_{\text{كنّى},5834} = 2 \cdot \left(\left(\log_2 \frac{7275}{16}\right) + 1\right) = 18.04 \qquad (3.6)$$

$$TFIDF(\text{"سمّى"}, 5834) = TF_{\text{سمّى},5834} \cdot TIDFF_{\text{سمّى},5834} = 2 \cdot \left(\left(\log_2 \frac{7275}{119}\right) + 1\right) = 9.00 \qquad (3.7)$$

Terme de la requête	Poids (TFIDF)
كنّى	18.04
سمّى	9.00

Tableau 3.3. Indexation de la requête R_3 (H_{5834}).

5.4. Calcul de similarité

Les hadiths pertinents pour la requête précédente sont présentés dans le tableau 3.4 ci-dessous. Cette liste représente les hadiths rapportés par le système de recherche d'information comme réponses à la requête R_3 représentée par le hadith 5834 du « Sahîh Al-Boukhârî ». Le tableau 3.5 indique les résultats de calcul des poids des termes et des scores des hadiths pertinents.

Kitab	Bab	Num Hadith	Matn Hadith
الخمس	قول الله تعالى: {فأن لله خمسه وللرسول} /الأنفال: 41/. يعني: للرسول قسم ذلك، قال رسول الله صلى الله عليه وسلم: (إنما أنا قاسم وخازن، والله يعطي)	2946	سموا باسمي، ولا تكنوا بكنيتي، فإني إنما جعلت قاسما أقسم بينكم
المناقب	كنية النبي صلى الله عليه وسلم	3344	سموا باسمي، ولا تكنوا بكنيتي
الأدب	قول النبي صلى الله عليه وسلم: (سموا باسمي ولا تكنوا بكنيتي). قاله أنس، عن النبي صلى الله عليه وسلم. [ر:2014]	5833	سموا باسمي ولا تكنوا بكنيتي
الأدب	من سمى بأسماء الأنبياء. وقال أنس: قبل النبي صلى الله عليه وسلم إبراهيم، يعني ابنه. [ر:1241]	5843	سموا باسمي ولا تكتنوا بكنيتي، فإنما أنا قاسم أقسم بينكم

Tableau 3.4. Liste des hadiths pertinents pour la requête R_3 (H_{5834}).

Terme de l'index	Poids pour le hadith 2946	Poids pour le hadith 3344	Poids pour le hadith 5833	Poids pour le hadith 5843
كنّى	18.04	18.04	18.04	18.04
سمّى	9.00	9.00	9.00	9.00
جعل	4.21	00.00	00.00	0.000
قسم	9.85	00.00	00.00	9.85
Cos (H, R_3)	**0.88**	**1.00**	**1.00**	**0.89**

Tableau 3.5. Scores des hadiths pertinents pour la requête R_3.

5.5. Evaluation du système de recherche

Puisque l'objectif d'un système de recherche est de minimiser l'effort fourni par un utilisateur pour accéder à ce qu'il cherche, alors les critères d'évaluation s'intéressent à la rapidité de la recherche et de la qualité des résultats. Tout en étant conscient des limites de toute évaluation, certains critères tentent tout de même de mesurer la qualité d'une recherche. Après chaque recherche, le système propose une liste de documents à l'utilisateur à partir de l'ensemble des documents du corpus cible. Cette liste contient des documents pertinents et d'autres non pertinents. D'autre part, certains documents pertinents ont été « oubliés » et ne font pas partie de la liste. Toute recherche subdivise donc le corpus en 4 sous-ensembles : celui des documents pertinents trouvés, celui des documents pertinents non trouvés, celui des documents non pertinents trouvés à tort et celui des documents non pertinents non rapportés. Parmi les objectifs d'un système de recherche figurent la diminution du nombre de documents non pertinents trouvés à tort et la diminution du nombre de documents pertinents non rapportés. Les critères correspondant à ces deux objectifs se nomment « précision » et « rappel ».

5.5.1. Courbes de rappel, précision et rappel/précision

Une des manières de tenir compte à la fois du rappel et de la précision d'un système est d'interpoler les valeurs de précision correspondant à différents niveaux de rappel. Les niveaux standards de rappel varient entre 0 et 1 suivant un pas de 10%. La règle d'interpolation généralement utilisée définit la précision pour un niveau de rappel i comme étant la valeur maximale de précision pour tout niveau de rappel supérieur ou égal à i. Cette définition permet d'obtenir une valeur de précision pour un rappel nul (elle correspond au niveau maximal de précision obtenu pour un rappel quelconque). Par exemple, s'il existe 200 documents pertinents pour une requête, la valeur interpolée de la précision pour un rappel de 10% correspond à la meilleure précision obtenue avec au moins 20 documents pertinents [Bellot 2000].

Nous utilisons les trois requêtes {R_1, R_2 et R_3} pour évaluer l'exactitude de notre système de recherche. Ces trois requêtes correspond aux hadiths 33, 129 et 5834 du corpus « *Sahîh Al-Boukhârî* ». Les Tableaux 3.6, 3.7 et 3.8 indiquent les résultats d'évaluation obtenus sur notre corpus prophétique selon les trois requêtes R_1, R_2 *et* R_3:

Doc	Similarité	Pertinence	Précision	Rappel
210.txt	0.7015	1	1.00	0.20
260.txt	0.2549		0.50	0.20
211.txt	0.2390	1	0.66	0.40
136.txt	0.1852	1	0.50	0.60
266.txt	0.1698		0.30	0.60
225.txt	0.1663	1	0.36	0.80
140.txt	0.1551		0.33	0.80
3.txt	0.1548	1	0.38	1.00
140.txt	0.1551		0.33	0.80

Tableau 3.6. Résultats d'évaluation pour la requête R_1.

Doc	Similarité	Pertinence	Précision	Rappel
21.txt	0.4335	1	1.00	0.25
135.tx	0.3812	1	1.00	0.50
247.tx	0.3738		0.66	0.50
307.tx	0.3475		0.50	0.50
15.txt	0.3448		0.40	0.50
81.txt	0.3126		0.33	0.50
14.txt	0.2689		0.18	0.50
11.txt	0.2678	1	0.25	0.75
90.txt	0.2675	1	0.30	1.00

Tableau 3.7. Résultats d'évaluation pour la requête R_2.

Doc	Similarité	Pertinence	Précision	Rappel
3344.txt	1.00	1	1.00	0.16
3345.txt	1.00	1	1.00	0.33
3346.txt	1.00	1	1.00	0.50
5833.txt	1.00	1	1.00	0.66
2014.txt	1.00		0.80	0.66
2015.txt	1.00		0.66	0.66
5843.txt	0.89	1	0.71	0.83
2946.txt	0.88		0.62	0.83
5844.txt	0.70	1	0.66	1.00

Tableau 3.8. Résultats d'évaluation pour la requête R_3.

La Figure 3.12 représente la comparaison des courbes de précision pour les trois requêtes R_1, R_2 *et* R_3:

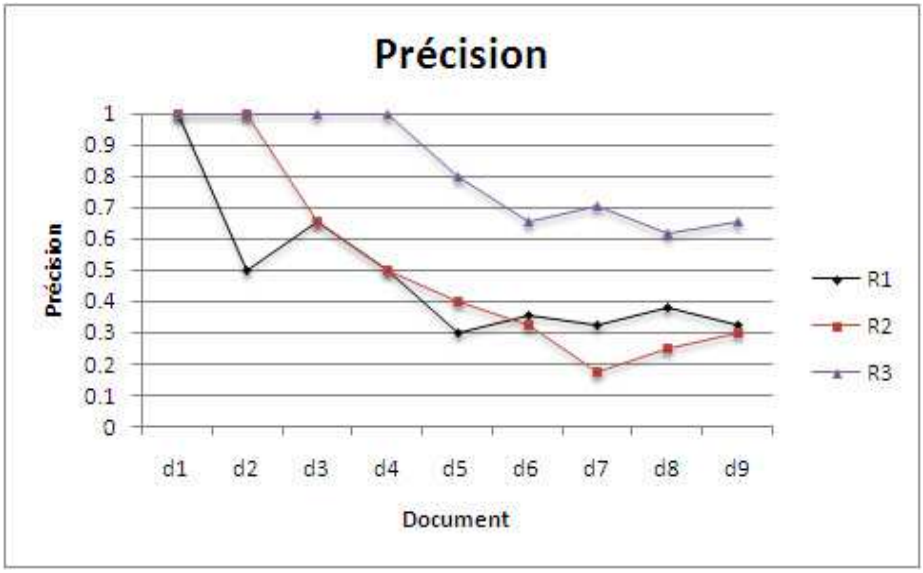

Fig. 3.12. Courbe Précision pour les trois requêtes R_1, R_2 *et* R_3.

La Figure 8.13 représente la comparaison des courbes de rappel pour les trois requêtes R_1, R_2 *et* R_3:

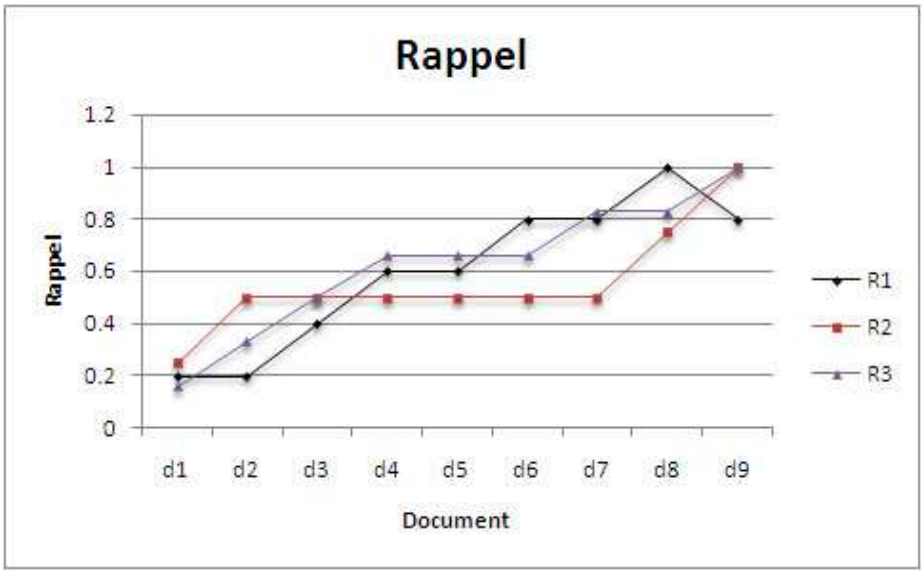

Fig. 3.13. Courbe Rappel pour les trois requêtes R_1, R_2 *et* R_3.

La précision moyenne est de 0.48 pour la requête R_1, de 0.51 pour la requête R_2 et de 0.82 pour la requête R_3. Le rappel moyen est de 0.60 pour la requête R_1, de 0.55 pour la requête R_2 et de 0.62 pour la requête R_3.

La Figure 3.14 représente la comparaison des courbes de rappel/Précision pour les trois requêtes R_1, R_2 *et* R_3:

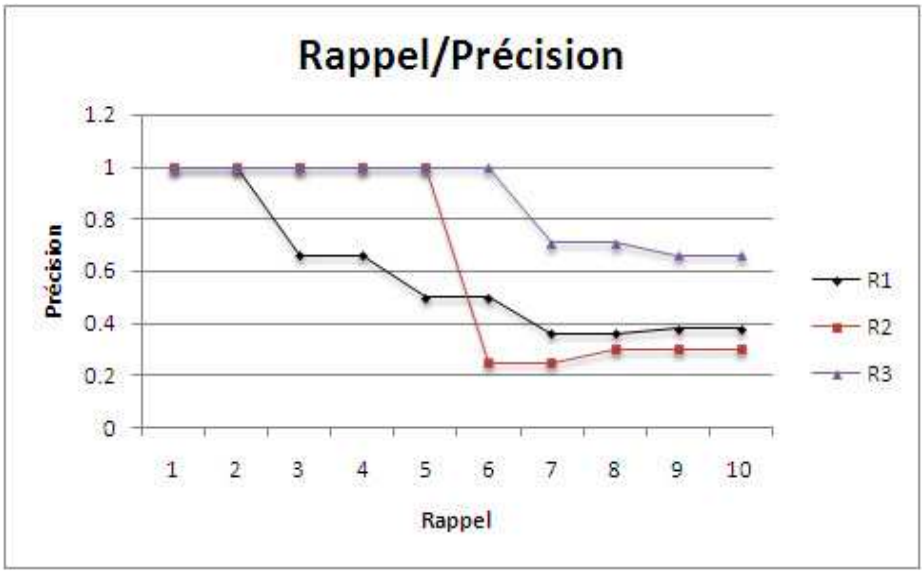

Fig. 3.14. Courbe Rappel/Précision pour les trois requêtes R_1, R_2 *et* R_3.

Les pré-traitements appliqués sur les trois requêtes (élimination des mots vides, lemmatisation et pondération TFIDF) sont identiques. Le niveau de précision pour un rappel de 0,5 est de 50 % supérieur pour les requêtes R_2 et R_3 par rapport à la requête R_1. Des hausses comparables sont constatées pour les autres critères d'évaluation : amélioration de 50 % de la précision pour le deuxième document en ce qui concerne les requêtes R_2 et R_3 par rapport à la requête R_1 ; amélioration de 30 % du rappel pour le deuxième document en ce qui concerne la requête R_2 par rapport à la requête R_1 et amélioration de 10 % du rappel pour le deuxième document en ce qui concerne la requête R_3 par rapport à la requête R_1.

Les écarts entre les résultats mentionnés montrent que les méthodes testées sont mieux adaptées aux requêtes R_2 et R_3. Dans ce cas, un tel écart démontre que les requêtes ne sont pas assez nombreuses pour pouvoir aboutir à un jugement réel sur le fonctionnement du système de recherche d'information.

6. Conclusion

Dans ce chapitre, La mise en œuvre d'un système de fouille de texte basée sur le modèle vectoriel a été discutée. Ce système a été créé dans le but de fournir une liste de hadiths classés selon leurs degrés de similarité avec une requête donnée. Ce système sera aussi utilisé pour l'évaluation des méthodes de classification et de segmentation présentées dans les chapitres suivants de ce mémoire. Les bases théoriques de la recherche d'information, la fouille de texte et les méthodes essentiellement statistiques, implémentées dans ce système, ont été exposées (modèle vectoriel, pondération des termes, mesures de similarités). Une place particulière a été accordée à la description des algorithmes utilisés pour l'indexation et la recherche. La question primordiale de l'évaluation et de la comparaison qualitative des systèmes et des méthodes de recherche est également abordée. Dans les chapitres suivants, des méthodes de classification et de segmentation des documents sont proposées et évaluées. Leur objectif commun est d'offrir à l'utilisateur une structuration des réponses fournies par un système de recherche d'information et d'en améliorer la précision.