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

UNIVERSITE MOHAMED BOUDIAF - M'SILA
FACULTE DES MATHEMATIQUES ET DE L'INFORMATIQUE

DEPARTEMENT D'INFORMATIQUE

MEMOIRE de fin d'étude

Présenté pour l'obtention du diplôme de MASTER

Domaine : Mathématiques et Informatique

Filière : Informatique

Spécialité : Technologie de l'Information et de Communication

Par : LAHRACHE Fatma

SUJET

Classification des textes prophétiques

Soutenu publiquement le : / /2016 devant le jury composé de :

Nom et prénom Enseignant	**Université de M'sila**	**Président**
Mr : Brahimi Belkacem	**Université de M'sila**	**Rapporteur**
……	**Université de M'sila**	**Examinateur**
……	**Université de M'sila**	**Examinateur**

Promotion : 2015 /2016

Remerciements

Avant de présenter ce travail, je tiens à remercier Dieu tout puissant, de m'avoir permis d'arriver à ce niveau d'étude, et aussi pour m'avoir donné beaucoup de patience et de courage.

Je remercier mes parents qui n'ont pas lésiné sur aucun problème et mon apporter toute aide nécessaire pour atteindre ce niveau qui me permettra d'assurer mon avenir إن شاء الله.

A travers ce mémoire je tiens à présenter mes sincères remerciements et notre profonde reconnaissance à notre aimable encadreur : Mr Brahimi Belkacem pour sons aide sons faille et sans oublier Mr Yaakoubi Rachad, qui nous ont aidé dans notre travail

Je tiens à vous dire que vous conseils et vos recommandations ont largement contribuées au succès dont je vous serez loyalement redevable.

J'adresser aussi mon sincère reconnaissance à tous les enseignants du département de Mathématique et informatiques de l'université de MSILA pour leur aides, soutiens et leurs conseils ainsi que tout le staff administratif du département.

Merci à tous ceux et celles qui ont contribués de près ou de loin dans l'accomplissement de ce travail.

LAHRACHE FATMA

Dédicace

Au nom de dieu misércordieux et à son prophète Mohammed « صلى الله عليه وسلم »,

Je dédie à mes parents qu'Allah protège comme dieu a dit :

{وَقَضَى رَبُّكَ أَلاَّ تَعْبُدُواْ إِلاَّ إِيَّاهُ وَبِالْوَالِدَيْنِ إِحْسَانًا} {الاسراء/ 23-25 }

Mon Pére « Bachir » qui a allumer la premier bougie de ma vie depuis mon enfance, refuge et mon refuge, qui a supporter ma douleur de chaque instant de ma vie, à celui qui s'est sacrifier pour nous tous il est notre fierté.

A ma Mère mon ange « Fatiha » la plus belle mère dans le monde, de son câlins chaleureux qui m'a toujours supporté et qui m'a souhaité beaucoup de bonheur.

A mes Frères « F.Brahim » l'homme qui a bon cœur et qui est prêt à nous aider dans les moments difficiles à sa Femme « Assia », à mon Frère « Sofien » source de tendresse et sa Femme « Nawal » , à mon Frère « Aissa » jumeau de mon âme, à mes chères Oncles « Noureddine, Yousef », à mes Tantes , à ma chère Mère « Aicha » lui souhaitons une longue vie ,à mes petits Frères et sœurs « Aymen, Bachir, Moncef , Aya, Malek », à mon esprit Akram et à tous mes amis et mes Collègues.

Aujourd'hui un nouveau bonheur prend forme dans ma vie, c'est grâce à vous on a su entretenir notre merveilleuse relation, notre belle complicité. Vous qui été toujours là pour moi.je vous remercie du fond de mon âme.

LAHRACHE FATMA.

Table des matières :

CHAPITRE 3 : LA CLASSIFICATION DU TEXTE

CHAPITRE 4 : LES TACHES DE PRETRAITEMENT D'UN TEXTE

CHAPITRE 5 : RESULTATS ET ANALYSES

Liste de figure :

Liste des tables :

Introduction générale :

La disponibilité croissantes de la quantité énorme de l'information et la cause de l'inflation du volume des données, lorsqu'on parle des données massives nous sources, du volume qui arrive à des centaines de téraoctets ou béta octets.

A partir de là le Data Mining aperçus comme une technique conçu parlons des quantités que nous ne peuvent pas imaginer, des données des différents types et des différentes pour extraire des connaissances et à partir de la quantité énorme des informations, cette technique basée sur des algorithmes sont basés sur l'exploration de données, il est dérivé de nombreuses des sciences telles que les statistiques, la logique, l'intelligence artificielle, systèmes experts, etc.

La fouille de donnée (DM) et la fouille de donnée textuelle (TM) sont des technologies modernes qui sont utilisées dans le système d'information, le Text mining (TM) à savoir de l'extraction de l'information utile à partir de gros volumes de contenus textes.

Les différentes recherches précédentes de la classification des textes conçus beaucoup plus sur des textes française et des textes anglaise, la classification des textes arabes sont moins nombreux que les autres langues.

Le Coran et la Sunna sont les deux principales sources de la théologie islamique, El hadith est l'ensemble des paroles et des actes de l'envoyer de Dieu prophète MOHAMMED.

Lorsqu'on a fait des recherches sur les classifications des textes prophétiques on a remarqué que peu de recherches sur la classification des textes prophétiques.

Notre objectif de travail est de mener une étude sur la classification des textes prophétiques, faire une comparaison entre les algorithmes de classification supervisée, entre les modèles de représentations de texte (stem arabic, stemlight et n-gram).

Pour réaliser cet objectif, nous avons créé une collection des textes prophétique de Sahih Elboukhari qui représente deux corpus ayant un nombre de classe différent pour étudier l'effet du nombre de classes sur les performances des classifieurs.

1. On a appliqué des méthodes de classification supervisé (Naive bayes, KNN et SVM), le but est de faire une étude comparative pour répondre à la question suivante :
   - Quel est le meilleur algorithme adapté à la classification des textes prophétique ?
2. pour chaque méthode de classification on a étudié l'effet des différents étapes du prétraitement (tokenization,stopwords,stem arabic,stemlight et n-gram) sur la classification des textes prophétiques.

3. L’objectif et de comparer les différents techniques de représentation du textes prophétiques (stem arabic,stemlight et n-gram) sur la base des critères de mesures de classification (F-mesure et Accurancy).

Ce travail contient 5 chapitres :

Le premier intitulé « Data mining » dans lequel nous avons parlé principalement des notions sur la fouille de donnée et la fouille de donnée textuelle (TM).

Le deuxième intitulé « Le texte prophétique Sahih Elboukhari » dans ce chapitre, nous avons parlé sur Sahih Elboukhari et sa description, nous avons également présenté sur corpus qui nous avons collecté et étudié dans ce travail, aussi parlé sur les recherches faites dans les domaines de classification sur les textes prophétiques et leur résultats.

Le troisième intitulé « La classification de texte » ce chapitre présente les tâches de la classification, on a décrit les algorithmes les plus utilisés dans l’apprentissage automatique et fouille de donnée textuelle.

Le quatrième chapitre « les tâches de prétraitement d’un texte » dans lequel on a présenté les différents techniques et opérations de prétraitement sur le texte afin de le préparé pour la tâche de fouille de donnée textuelle (classification).

Cinquième chapitre intitulé « Résultats et analyse » concerne la partie pratique dans lequel nous avons appliqué les différents algorithmes d’apprentissage supervisé sur notre corpus, et définie les résultats obtenus et les résultats de l’étude comparative.

CHAPITRE 1 :

DATA MINING

1 Introduction :

La fouille de donnée (souvent appelée « data mining ») est l'exploration et l'analyse de grandes quantités de données afin d'y découvrir de l'information implicite. Cette information peut être de différente nature, par exemple on recherchera des règles d'association, une classification ou une segmentation de population

Dans ce chapitre on va résumer en brève la définition de la fouille de données, leur processus on expliquant aussi les différant taches de la fouille de données.

2 Fouille de donnée :

2.1 Définition du Fouille de donnée :

La fouille de données, qui est définie en anglais par (Data Mining DM) est un ensemble de méthodes, de techniques et d'outils pour exploiter les documents non structurés se sont des textes écrits. Pour extraire le sens des documents non structurés, de déterminer le sens d'un texte sans nécessairement en lire tout le contenu dans le but de découvrir des informations cachées ou prendre automatiquement la bonne décision. D'une manière plus précise, Il s'agit aussi d'organiser et de les structurer afin d'en dégager des thématiques, des relations dans une perspective d'analyse non littéraire rapide.

La fouille de données textuelles utilisé pour classer des documents, réaliser des résumés de synthèse automatique ou encore pour assister la veille stratégique ou technologique selon des pistes de recherches prédéfinies.

La fouille de donnée textuelle est bien motivé, en raison du fait qu'une grande partie des données du monde peuvent être trouvés sous forme de texte (articles de journaux, des e-mails, de la littérature, des pages Web, etc.).

2.2 Les tâches de la fouille de données :

Beaucoup de problèmes intellectuels, économiques ou même commerciaux peuvent être exprimés en termes des six tâches suivantes : [21]

1. La classification.
2. L'estimation.
3. La prédiction.
4. Le groupement par similitude.
5. L'analyse des clusters.
6. La description.

Les trois premières tâches sont des exemples du Data Mining supervisé dont le but est d'utiliser les données disponibles pour créer un modèle décrivant une variable particulière prise comme but en termes de ces données. Le groupement par similitude et l'analyse des clusters sont des tâches non-supervisées où le but est d'établir un certain rapport entre toutes les variables. [20]

La description appartient à ces deux catégories de tâche, elle est vue comme une tâche supervisée et non-supervisée en même temps. [21]

On expliquant ces différentes tâches.

2.2.1 Classification :

La classification permet de créer des classes d'individus (terme à prendre dans son acception statistique).celles-ci sont discrètes : homme/femme, oui/non, rouge/vert/bleu.[6]

2.2.2 L'estimation :

Contrairement à la classification, le résultat d'une estimation permet d'obtenir une variable contenue. Celle-ci est obtenue par une ou plusieurs fonctions combinant les données en entrée.

Le résultat d'une estimation permet de procéder aux classifications grâce à un barème. Par exemple, on peut estimer le revenu d'un ménage selon divers critères (type de véhicule et nombre profession ou catégorie socioprofessionnelle, type d'habitation ... etc.).

Il sera ensuite possible de définir des tranches de revenus pour classifier les individus.

La technique la plus appropriée à l'estimation est le réseau de neurones. [6]

2.2.3 La prédiction :

La prédiction ressemble à la classification et à l'estimation mais dans une échelle temporelle différente. [6]

2.2.4 Le regroupement par similitude :

Le regroupement par similitude consiste à grouper les éléments qui vont naturellement ensembles.la technique la plus appropriée au regroupement par similitude est l'analyse du panier de la ménagère. [6]

2.2.5 L'analyse de clusters :

L'analyse de clusters consiste à segmenter une population hétérogène en sous population homogène. Contrairement à la classification, les sous populations ne sont pas préétablis. La technique la plus appropriée à la classification est l'analyse des clusters. [6]

2.2.6 La description :

C'est souvent l'une des premières tâches demandées à un outil de Data mining. On lui demande de décrire les données d'une base complexe. Cela engendre souvent une exploitation

supplémentaire en vue de fournir des exlications.la technique la plus appropriée à la description est l'analyse du panier à la ménagère. [6]

2.3 Les étapes du processus de data mining :

L'image suivante représente les différentes étapes du processus d'extraction de connaissance.

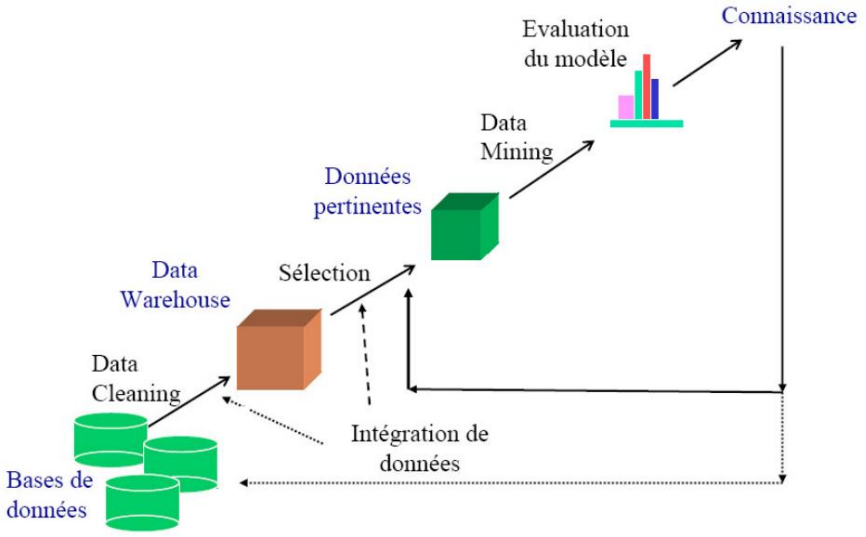

Figure1.1 Processus d'extraction de connaissance à partir d'une base de données [37]

2.3.1 Collecte des données : la combinaison de plusieurs sources de données, souvent hétérogènes, dans une base de données. [25] [27].

2.3.2 Nettoyage des données : la normalisation des données : l'élimination du bruit (les attributs ayant des valeurs invalides et les attributs sans valeurs) [25] [27].

2.3.3 Sélection des données : Sélectionner de la base de données les attributs utiles pour une tâche particulière du data mining [3].

2.3.4 Transformation des données : le processus de transformation des structures des attributs pour être adéquates à la procédure d'extraction des informations. [29]

2.3.5 Extraction des informations (Data mining) : l'application de quelques algorithmes du data mining sur les données produites par l'étape précédente (Knowledge Discovery in Databases, u KDD) [27] [5].

2.3.6 Visualisation des données : l'utilisation des techniques de visualisation (histogramme, camembert, arbre, visualisation 3D) pour exploration interactive de données (la découverte des modèles de données) [29] [27].

2.3.7 Evaluation des modèles : l'identification des modèles strictement intéressants en se basant sur des mesures données. [25]

3 Fouille de données textuelle :

Le Fouille de donnée textuelle, (en anglais appelé Text Mining) 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 ainsi découvrir des connaissances et des relations à partir des documents disponibles.

L'outil de Text Mining va générer de l'information sur le contenu du document. Cette information n'était pas présente, ou implicite, dans le document sous sa forme initiale, elle va être rajoutée, et donc enrichir le document. [19]

3.1 Objectifs de la fouille de données textuelles :

La fouille de données textuelle peut être utilisée en particulier dans les cas suivants :

- Pour mieux comprendre le positionnement d'un discours, d'une thèse, d'un communiqué.
- Pour appréhender les thèmes récurrents qui sont associés à une activité, une entreprise ou des concurrents.
- Pour mesurer les points faibles et les points forts dans une revue de presse.
- Pour comparer des textes sur un même thème afin d'en déterminer les points communs ou au contraire de distinguer les différences stylistiques.
- Pour créer automatiquement des répertoires de sites Web ou emails associés à des thématiques. Pour quantifier un texte ou les parties d'un texte pour en extraire les structures signifiantes les plus fortes telles que le résumé automatique et la segmentation thématique.
- Pour établir des liens entre les termes et les documents utilisés dans l'indexation.
- Pour établir des règles de classification automatique de documents (classification supervisée ou non supervisée).

4 Classification de texte (TC) :

Classification de texte (TC) également connu sous le nom de catégorisation de texte, est la tâche d'attribuer automatiquement un ensemble de documents en catégories ou classes ou sujets à partir d'un ensemble prédéfini. Cette tâche, qui tombe au carrefour de la recherche d'informations (IR) et l'apprentissage machine (ML), a été témoin d'un intérêt en plein essor au cours des dix dernières années, des chercheurs et les développeurs. [28]

5 Application de Classification de texte :

Peut fournir des vues conceptuelles de collections de documents et des applications importantes dans le monde réel par exemple [8] :

- Les reportages sont généralement organisés par catégories de sujets ou par des codes géographiques.
- Les Documents universitaires sont souvent classés par domaines techniques et sous-domaines.
- Les rapports des patients dans les organismes de santé sont souvent indexés à partir de plusieurs aspects : Le tri des fichiers dans les hiérarchies de dossiers, sujet des identifications, des intérêts dynamiques basés sur les tâches, organisation automatique des métadonnées, filtrage du texte organisation du document des bases de données et les pages Web.
- Analyse d'opinion (opinion mining) c'est classer automatiquement les revues (textes d'opinions) en positive, négative, neutre.
- Une autre application généralisée de la catégorisation de textes est le filtrage de spam, où les messages électroniques sont classés en deux catégories spam et non-spam
- Identification du l'auteur : dans ce cas, le système de classification doit identifier l'auteur du texte.

6 Problème de Classification de texte :

Le problème de la classification de texte se compose de plusieurs sous-problèmes qui ont été étudiés de manière intensive dans la littérature tels que l'indexation de documents, l'attribution de la pondération, regroupement de documents, la réduction de dimensionnalité, de détermination de seuil et le type de classificateurs...

Plusieurs méthodes ont été utilisées pour la classification de texte tel que :

- Support Vector Machines (SVM).
- K voisin le plus proche (KNN).
- Les réseaux de neurones (NN).
- Naïf Bayes (NB).
- Les arbres de décision (DT).
- Entropie maximum (ME).
- Règles d'association.

7 Langue arabe :

La langue arabe est la langue des populations arabes qui firent leur entrée dans l'histoire depuis 3 millénaires environ et qui occupaient les zones septentrionales de l'Arabie.

La langue arabe considéré la 5éme langue courantes utilisées dans le monde. Elle est parlée par plus de 422 millions de personnes en tant que première langue et de 250 millions en tant que langue secondaire, La langue arabe fait partie de la grande famille des langues sémitique. [33]

Le système archaïque d'écriture arabe était consonantique. Chaque lettre de l'alphabet arabe représente une consonne unique depuis les temps anciens. Cependant, la fin du VIIe siècle,

Les diacritiques arabes qui sont des symboles graphiques qui discriminent entre la variété des prononciations des consonnes, ont été inventés par "Abou Al-Aswad Al-Du'ali".

Néanmoins, ils sont très souvent éliminés du texte écrit d'aujourd'hui. Lecteurs arabes pouvaient discerner les mots avec la même forme d'écriture par l'intermédiaire de son contexte.

Diviser le texte d'entrée en fractions désirées est généralement la phase initiale dans la plupart des tâches de traitement de texte.

Ces fractions pourraient être des phrases, des chiffres, des mots, des caractères ou toute autre fraction utile. Chaque fraction est appelé un **« Token »** et le processus est appelé **« Tokenization ».**

En arabe **Token** peut spécifier toute une phrase grammaticale par exemple «و سنساعدهم » (ce qui signifie : "et nous allons les aider»).

L'un des éléments les plus efficaces dans les phrases distinctives ou limites symboliques est des signes de ponctuation.

Ils ont émergé dans le système d'écriture arabe en 1912. En fait, l'utilisation de la ponctuation ne persiste pas dans la langue arabe.

L'alphabet arabe se compose de vingt-huit (28) lettres fondamentales

(أ ب ت ث ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف ق ك ل م ن ه و ي), (vingt-neuf (29) lettres si on n'a pas exclu la hamza (الهمزة), qui se comporte soit comme une lettre à part entière soit comme un diacritique).Il s'écrit de droite à gauche.

Dans la langue arabe Il n'y a pas de majuscules ou minuscules pour les lettres comme les lettres anglaises.

Les signes diacritiques (الحركات) dans la langue arabe permettent d'exprimer les voyelles brèves et d'apporter différentes modulations aux voyelles longues ainsi qu'aux consonnes.

Le but de signes diacritiques pour apprendre à les reconnaitre et à les prononcer correctement en contexte, pour distinguer des lettres ambigües et pour faciliter la lecture.

La majeure partie de l'écriture arabe est écrit sans Harakat. Cependant, ils sont couramment utilisés dans certains textes religieux qui exigent le strict respect des règles de prononciation telles que Qur'an (القرآن). Il est fréquent d'ajouter Harakat à hadiths (الحديث), ainsi une autre utilisation dans la littérature pour enfants pour connaitre le sens des mots arabe. Harakat sont également utilisés dans les textes ordinaires quand une ambiguïté de la prononciation pourrait se poser. [7]

Les Diacritiques arabes comprennent :

Fatha (ـَ) (الفتحة), Kasra (ـِ) (الكسرة), Damma (ـُ) (الضمة), Soukoune (ـْ) (السكون), Shadda (ـّ) (الشدة) et Tanwin (ـً) (التنوين).

Fatha : permet la réalisation de la voyelle brève [a]. Il se présente sous la forme d'un accent aigu placé juste au-dessus de la lettre. [12]

Damma : permet la réalisation de la voyelle brève [u].Il se présente sous la forme d'un mini waw (و) placé juste au-dessus de la lettre. [12]

Kasra : permet la réalisation de la voyelle brève [i]. Il se présente sous la forme d'un accent aigu placé juste en-dessous de la lettre. [12]

Soukoune : Les syllabes peuvent être ouvertes ou fermées. C'est-à-dire Si la syllabe se termine par une consonne, elle est fermée. Si la syllabe se termine par une voyelle elle est ouverte. Pour indiquer qu'une syllabe est fermée (à la prononciation) on ajoute simplement un soukoune (petit cercle) au-dessus de la lettre. [12]

Tanouine :

- Tanouine fatha ou fathatan : permet la réalisation du son [an]. Il se présente sous la forme d'un double fatha.
- Tanouine damma ou dammatan : permet la réalisation du son [on].Il se présente sous la forme d'un double damma.
- Tanouine kasra ou kasratan : permet la réalisation du son [en] ou [an]. Il se présente sous la forme d'un double kasra. [12]

Les prononciations de diacritiques précités pour la lettre arabe (ب) sont présentées dans le tableau suivant :

Voyelle			**Nunnation**			**Pas Voyelle**	**Deux constantes**
b + a (fatha) = ba	b + u (damma) = bu	b + i (kasra) = bi	b + an (tanouine fatha) = ban	b + un (tanouine damma) = bun	b + in (tanouine kasra) = bin	b + (soukoun) = b	b + shadda = bb
بَ	بُ	بِ	بً	بٌ	بٍ	بْ	بّ

Table 1.1 : diacritiques précités pour la lettre arabe (ب)

8 Complexité de la langue arabe :

L'arabe est une langue difficile pour un certain nombre de raisons :

- Orthographe avec diacritiques est moins ambiguë et plus phonétique en arabe, certaines combinaisons de caractères peuvent être écrites de différentes manières. [32]
- Langue arabe a voyelles courtes qui donnent la prononciation différente. Grammaticalement ils sont nécessaires, mais omis dans les textes arabes écrits. [30]
- La langue arabe a une morphologie très complexe que comparer à la langue anglaise.
- Les synonymes sont très répandus. [15]
- La Classification automatique du texte dépend du contenu des documents, un grand nombre de fonctionnalités ou des mots-clés peut être trouvée dans le texte arabe tel que les morphèmes qui peuvent générées à partir d'une racine qui peut conduire à une mauvaise performance en termes de précision et de temps. [15]

9 Conclusion :

Ce chapitre présente la fouille de données DM, les différents processus du DM, et leur tâches, la fouille de données textuelle la classification de texte TC, aussi on a focalisé sur la langue arabe et sa complexité.

CHAPITRE 2 :

LE TEXTE PROPHETIQUE « Sahih Alboukhari »

1 Introduction :

Le Coran et la Sunna sont les deux principales sources de la théologie islamique. Dieu à envoyer le Coran à travers DJEBREEL pour le prophète MOHAMMED, qui est définie en arabe قرآن « la parole de Dieu » .ce que signifié message et non pas une loi.

Qui veut dire une écriture universelle adressé à toute l'humanité. Dieu a expliqué les règles de l'islam et les grandes lignes de cette religion par le coran. Mais le coran doit montrer en détail que la deuxième référence de l'Islam est la Sunna. Elle représente l'application pratique de ce que révélé Dieu tout-puissant dans le coran.

Sunna est l'ensemble des paroles et des actes de l'envoyer de Dieu le prophète MOHAMMED, elle est dicter par ses compagnons « SAHABA » afin de nous faire connaitre tous les actes du prophète et les appliques.

Ce chapitre traite des concepts liés aux corpus textuel corpus de (SAHIH AL BOUKHARI)

2 Le corpus prophétique de L'imam Al-Boukhârî :

L'imam Al-Boukhârî rédige des différents ouvrages de hadiths, mais le plus connu est le Sahih alboukhari « AL-jami'us-Sahih », après un examen minutieux et rigoureux, il enregistra des paroles du prophète paix et bénédictions sur lui dont la chaine de transmission ne se compose que de transmetteurs justes et fiable, sans défaut ni brisure. Pendant seize ans, il scruta 600 000 hadiths et retint 7 275 hadiths avec répétition et environ 2230 sans répétition dont l'authenticité est au-delà du moindre doute. 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.

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 après le Coran l'œuvre la plus fiable au sujet de l'Islam original. [9]

3 Description textuelle du « Sahîh Al-Boukhârî » :

Le corpus « Sahîh Al-Boukhârî » est structuré textuellement de la façon suivante :

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

Al hadith se compose de plusieurs éléments sont :

- Kitab : des livres qui rassemblent Al hadiths dans tel est raconté par chaque compagnon des prophètes.
- Bab : les textes prophétiques distribués par Abwab (أبواب), chaque Bab (باب) contient un ou plusieurs textes prophétiques dans un sujet, et chaque Bab (باب)à un titre qui indique le thème de Hadith.
- Sanad : est la chaîne des personnes qui ont transmises le hadith (chaîne de transmission).
- Matn : le texte, la parole en elle même.
- Atraf : est la partie ou la phrase commençante du texte qui fait référence à la parole et à l'action ou à la caractéristique du prophète صلى الله عليه وسلم, ou de son accord donné à d'autres actions. **الأطراف جمع طرف ، وطرف الحديث ، الجزء الدال على الحديث ، أو العبارة الدالة عليه ، مثل حديث الأعمال بالنيات ، وحديث الخازن الأمين ، وحديث سؤال جبريل .**

La figure suivant représente un exemple typique d'un Hadith du Sahîh Al-Boukhârî et ses structures :

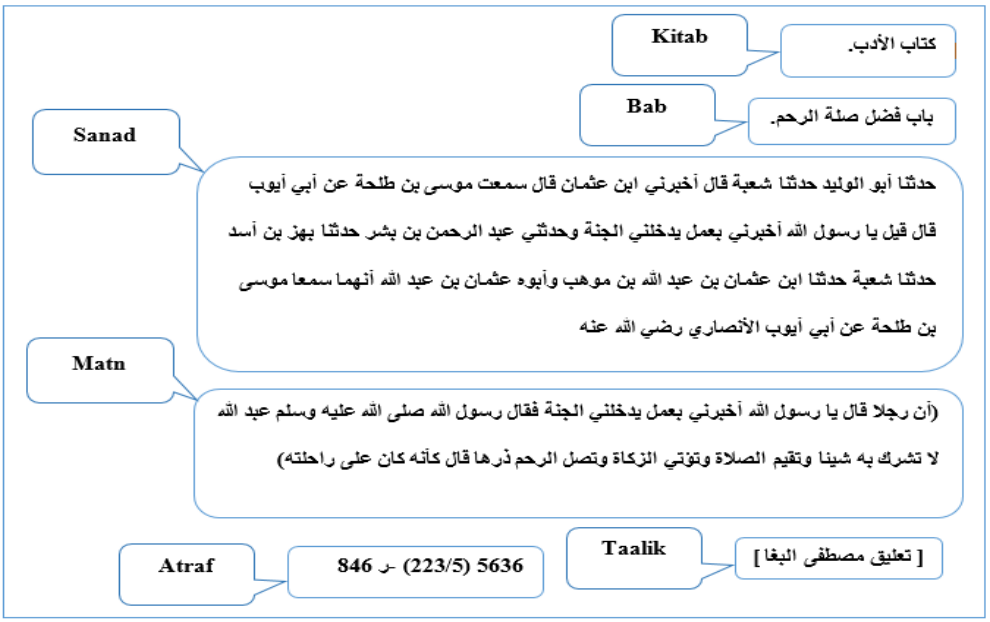

Figure 2.1 : Structure typique d'un Hadith du « Sahîh Al-Boukhârî ».

Mais dans notre travail nous basons uniquement sur Matn (المتن).

4 Etat de l'art :

On a ramené ces recherches à partir de l'article « Hadith data mining and classification : a comparative analysis » établi par un groupe de chercheurs, publiée le 08 janvier 2016. [24]

4.1 Classification de Hadiths :

Kabi et autre. (2005) : A prouvé que la classification Hadith peut être manipulée par une méthode simple avec précision raisonnable.

Ils sont classé Hadiths en huit 08 chapitres de Sahih Al Bukhari par le calcul de terme de fréquence (TF term frequency, IDF), Après avoir retiré isnad et arrêter les mots, chaque mot a été converti en sa forme de racine par un système de Stemmer.

Dans Sahih al-Bukhari le même Hadith peut exister dans plusieurs chapitres. Dans ce cas

Leur système affiche deux sujets avec les plus hauts rangs. Afin d'effectuer une pondération à long terme, la méthode TF-IDF a été utilisée et formé par 120 Hadiths.

Test du système avec 80 Hadiths présenté 83,2% de précision.

Dans une proposition de recherche **Kamsin et autre. (2014)** ont souligné l'importance d'un système d'authentification automatique pour le Coran et les Hadiths afin de lutter contre les fausses versions du Coran et du Hadith dans la sphère virtuelle.

Ghazizadeh et autre. (2008) : leurs systèmes experts flous basés sur un ensemble des règles et des opinions d'experts. Afin de rendre l'inférence deux moteurs d'inférence ont été conçus.

Le premier moteur produit le rang de chaque narrateur et il passe au second moteur d'inférence.

Le produit du deuxième moteur d'inférence est le taux de validation de Hadith.

Leur système a été testé en utilisant l'ensemble de données KAFI pour classer Hadiths avec : inconnu, faible, bonté, fiable, et juste ; en ce qui concerne leur taux d'authenticité.

Le système atteint 94% de précision via une combinaison de logique floue triangulaire, logique floue singleton, multiplication inférence, Mamdani multiplication inférence et defuzzifier moyenne.

Les réseaux de neurones artificiels (ANN) sont l'une des méthodes pour effectuer le classement. Qui utilisé une méthode ANN pour classer Hadiths.

Dans leur approche de classification que du texte en utilisant le prétraitement décomposition en valeurs singulières (SVD).

La première étape qui est un processus de nettoyage efficace des données. Il y a 739 vocabulaires uniques dans l'ensemble de données. Chaque fonction a référencé à un vocabulaire.

Le SVD convertit le vecteur dimensionnel clairsemé et haut à un vecteur de 200 dimensions de documents-termes de poids.(document-terms weights)

Ghazizadeh et autre. (2008) ont utilisé l'encyclopédie Prophétique (L'Encyclopédie des Neuf Livres pour les Traditions prophétiques honorables ! 1997), qui comporte 453 documents distribuées sur 14 catégories : foi, coran, connaissances, crimes, al-Jihad, les bonnes manières, les générations passées, biographie, jugements, culte, comportements, nourriture, vêtements, et les états personnels.

Un réseau de neurones à trois couches à action directe avec la fonction d'activation de la tangente hyperbolique de la couche cachée, suivie d'une couche de sortie linéaire a été formé en utilisant un algorithme de rétro-propagation.

Le rappel obtenu 87, précision 90 et F1-score 88% pour la catégorie prédiction Hadith.

Dans un classificateur d'arbre de décision : les feuilles se référer aux étiquettes de classe et les branches se réfèrent à la coïncidence des caractéristiques qui pointent vers labels particulières **(Maazouzi and Bahi 2012).**

Par ailleurs, un algorithme de haut en bas est nécessaire pour parcourir l'arborescence et de prédire les classes.

Harrag et autre. (2009) Illustre une expérience de classification en utilisant 453 Hadiths distribuées sur 14 groupes de l'encyclopédie Prophétique via ID3 classificateur **(Flachsbart et autre. 1994).**

La phase de prétraitement consiste à convertir le document au format Texte brut, en enlevant des mots d'arrêt et à endiguer (stemming).

Après le prétraitement le vecteur construit est composé de tous les termes dans les textes Hadith. Puis, une dimension du vecteur a été réduite à 1938 sur la base de certains critères spécifiques et un poids a été calculé pour chaque dimension à l'aide de fréquence à long terme (TF).

L'évaluation de la phase de test obtenu : 38% de rappel, 47% de précision, et 40% F-score. Une variété de tours de classification erronée en raison de la nature et les caractéristiques dans les documents de Hadith.

Sahih Al-Bukhari divise en plusieurs chapitres concernant le sujet de Hadiths,

À propos de Al Khatib (2010) seulement considéré huit d'entre eux dans l'expérience :

Connaissance, foi, hadj, prière, zakat, salat l kosof (prière d'éclipse), siyam et les bonnes manières. Al Khatib (2010) a examiné quatre algorithmes de classification : Naive Bayes **(Rennie et autre.2003),** algorithme Rachio **(Ragas et Koster, 1998)**, K-plus proche voisin (KNN) **(Zhang et Zhou 2005)**, et machine à vecteurs de support (SVM) **(Hsu et Lin 2002).**

La fréquence du terme document et inverse fréquence (TF-IDF) la méthode utilisée par (Aizawa 2003) pour calculer la fréquence relative pour chaque mot dans le document.

Pour des fins de formation, 1.350 hadiths ont été utilisés et 150 Hadiths pour tester la précision des méthodes de classification.

Le rappel moyen de toutes les méthodes était de 100%, mais la précision de Rachio était 67,11%, Naive Bayes 66,55%, KNN 66,55% et SVM était 63,36%. Par conséquent, l'algorithme Rachio classé Hadiths avec le niveau de précision le plus élevé.

Harrag et autre. (2011b) fourni une étude d'évaluation de plusieurs méthodes de stemming de Hadith catégorisation de textes.

Dans l'étude examinée la recherche dans un dictionnaire de stemming (dictionary-lookup stemming), la racine à base de stemming (rootbased stemming), et light stemming provenant comme une étape de réduction de la fonction.

La recherche dans le dictionnaire de stemming (dictionary-lookup stemming) effectué la troncature par deux ensembles de ressources :

1. Une base d'affixes.
2. Un dictionnaire de mots avec leurs racines correspondantes.

Les stemmers basés sur racine (root-based stemmers) : exécutent le modèle de correspondant

(pattern matching) pour découvrir la racine des mots.

Light stemming : enlève quelques suffixes et/ou préfixes, sans identifier des modèles.

Le ANN et SVM sont choisis pour la phase de catégorisation.

L'expérience a été réalisée sur 453 textes Hadith de l'encyclopédie Prophétique, qui a été distribué plus de 14 catégories.

Après avoir effectué le prétraitement et stemming, les données ont été représentées par quatre vecteurs :

1. unstemmed vector avec 4,055 dimensions.
2. light stemmed vector avec 2536 dimensions.
3. root-based stemmed vector avec 1063 dimensions.
4. dictionary-lookup stemmed vector avec 739 dimensions.

Les résultats expérimentaux ont montré que la méthode ANN était mieux que SVM en termes de F1-score.

Le F1-score obtenu sans stemming pour ANN est de 42%, alors que SVM était de 44%, même si elle a été renforcée après l'utilisation des stemmers.

Le dictionnaire recherche stemming a obtenu la plus haute précision (c-à-dire F1-score de 50%) par rapport à la racine à base de stemming et les méthodes de light-stemming pour le classificateur ANN.

Light stemming a obtenu la plus haute précision (c-à-dire F1-score de 48%) par rapport aux méthodes de la racine à base de stemming et un dictionnaire de recherche de stemming pour le classificateur SVM.

Par ailleurs, La taille abrégée des vecteurs a causé moins d'efforts de calcul pour les deux classificateurs.

Aldhaln et autres. (2012a, b) classer Hadith en quatre grandes classes Sahih, Hasan, Daif et Mawdhoua موضوع .

Afin de former l'entrainement (training) et test (testing) des ensembles de données 999 Hadiths collecté à partir de trois livres : Sahih Al-Bukhari, Jamiu Al-Tirmidhi et Silsilat Al-Hadith Al-Daeifah Wal AlMawdhuah.

Puisque l'ensemble de données a été composé à partir de différents livres, les données de prétraitement sont réalisée pour diminuer la redondance et de rendre le style de Hadith Isnad homogène.

Les caractéristiques des Hadiths déterminées selon les cinq principes de la science du Hadith :

1. narrateurs distingués pour leur franchise.
2. narrateurs distingués pour leur véracité.
3. aucune interruption dans l'Isnad.
4. aucune expression anormale dans le Matn.
5. aucune défectuosité dans Matn.

Aldhaln et autres ont présenté une nouvelle méthode pour traiter les données manquantes dans l'ensemble de données Hadith. La tâche de classification a été réalisée par deux méthodes différentes : Arbres de décision et Naive Bayes.

Cependant, le classificateur Naive Bayes donne de meilleurs résultats avec 97,6% de rappel et 97,597% de précision.

Ils ont également publié une version étendue du papier qui comprend plus de détails techniques sur la même expérience dans un article de journal **(Aldhaln et autre. 2012).**

Dans une étude comparative, Al-Kabi et Al-Sinjlawi (2007) ont démontré que le classificateur Naive Bayes donne les meilleurs résultats par rapport à d'autres classificateurs en termes de Sahih Al-Bukhari.

Najeeb (2014) a proposé une nouvelle approche de classification qui distingue authentifié (Sahih) et faible (Daif) Hadiths.

Classement associative a été utilisé pour assimilé la classification (association des règles mining) Association Rule Mining (ARM). Aussi appelé ABC (Classification Basée sur les Associations).

L'objectif du L'ARM : Découvrir la relation entre les caractéristiques afin de définir un ensemble de règles de classification.

Bien que la fonctionnalité de l'ABC sur le domaine Hadith a été confirmé, sans aucun taux de précision explicite n'a été rapporté.

Table 1 Overview on classification approaches

No.	Approach published by	Methods	Categories	Data source	Performance
1	Kabi et al. (2005)	TF-IDF	Eight subjects	200 Hadiths from Sahih Al-Bukhari	Accuracy: 83 %
2	Ghazizadeh et al. (2008)	Fuzzy expert system	Unknown, weak, goodness, reliable, and right	KAFI Dataset	Accuracy: 94 %
3	Harrag and El-Qawasmeh (2009)	Term indexing, artificial neural network, singular value decomposition	14 subjects such as: faith, Quran, knowledge, crimes, good manners and judgments	Prophetic encyclopedia	Recall: 87 %, Precision: 90 %, F1-score: 88 %
4	Harrag et al. (2009)	Term frequency, ID3 decision tree classifier,	14 subjects such as: faith, Quran, knowledge, crimes, good manners and judgments	Prophetic encyclopedia	Recall: 38 %, Precision: 47 %, F1-score: 40 %
5	Alkhatib (2010)	TF-IDF, Rachio algorithm	Eight subjects	1350 Hadiths from Sahih Al-Bukhari	Precision: 67 %
6	Harrag et al. (2011b)	TF-IDF, dictionary lookup stemming, and artificial neural network	14 groups such as: faith, Quran, knowledge, crimes, good manners and judgments	Prophetic encyclopedia	F1-score: 50 %
7	Aldhaln et al. (2012a, b) and Aldhaln et al. (2012)	Decision trees and Naive Bayes	Sahih, Hasan, Daif and Maudo	999 Hadiths from three books: Sahih Al-Bukhari, Jamiu Al-Termithi, and Silsilat Al-Ahadith Al-Daeifah Wal Al-Mawdhuah	Recall: 97 %, Accuracy: 97 %
8	Najeeb (2014)	Term indexing, associative classification	Sahih and Daif	Not reported	Not reported

Figure 2.2 Overview on classification approches. [24]

4.2 Data mining :

Karim et Hazmi (2005) effectue une analyse qualitative des données en interview avec les étudiants des Études supérieures malaisiennes pour évaluer les informations des Hadiths sur Internet.

Le résultat a montré que presque tous les participants considèrent l'Internet comme une ressource pratique de hadith, cependant il existe un risque d'utilisation défectueuse hadiths.

Shatnawi et autre. (2012) a expliqué une expérience qui contenait deux grandes étapes :

1. Récupérer Hadiths depuis les pages Web.
2. Vérifier l'exactitude de l'Hadiths récupéré.

Ils utilisent une base de données fournie par le cheikh Al-Albani qui contient plus de 17.000 Hadith ainsi que leurs degrés d'authentification.

Shatnawi et autre. (2012) illustre comment tokenize la base de données et de supprimer les mots d'arrêt.

Sauf pour les 28 alphabets arabes, tous les caractères ont été supprimés, y compris les marques de voyelles arabes.

Finalement, on a établi un index de position qui contient plus de 56.000 termes.

Afin d'extraire les textes de Hadith à partir des pages Web, un décapant « nettoyeur Java HTML élimine les codes HTML des pages Web.

Puis, quatre mots adjacents de la page Web ont été comparés à l'index de position de Hadith pour détecter le texte de Hadith.

Lorsque tous les textes Hadith ont été extraits, chaque hadith a été recherché dans la base de données afin de déterminer leur degré d'authenticité.

Lorsque cinq pages Web contenant des textes Hadiths ont été choisis au hasard (Englobant 63 textes Hadith), 76,1% de précision et 42,1% de rappel ont été affecté.

AthenTique est un outil de fouille de données textuelles à rechercher une requête à partir d'un ensemble de données Hadith **(Harrag et autre 2008;. Harrag et Hamdi-Cherif 2007).**

Il affiche une liste des hadiths importants classés par la mesure de similitude.

AthenTique fonctionne selon le modèle d'espace vectoriel (VSM) **(Salton et autre. 1975).**

VSM réfère à un concept qui stocke toutes les informations d'index avant de mesurer la similitude entre la requête et le texte principal.

La première étape Hadith morphologique stemming basé sur un dictionnaire de racine.

Après le prétraitement est effectué sur tous les Hadiths, le processus de pondération à long terme « term weighting process » et d'indexation peuvent commencer par la méthode TF-IDF.

En plus chaque requête est traitée de la même manière que l'ensemble de données Hadiths.

Ensuite, la similitude est mesurée entre la requête et le hadith à l'aide de la technique de mesure du cosinus.

La récupération de Hadith est effectuée en deux tours :

1. Tous les termes des cinq premiers documents pertinents sont classés par leur poids, de leur pertinence pour la requête initiale.
2. Les dix premiers termes des documents récupérés sont intégrés à la requête d'origine pour former une requête enrichie, tandis que les nouveaux termes ont un poids plus faible que les termes de requête initiale.

Une expérience faite avec 60 Hadiths obtenu 66% de précision et de 80% de rappel.

Un système de récupération de l'information conventionnelle récupère un ou plusieurs documents sur la base d'une requête.

Chaque document est généralement long que les utilisateurs ne peuvent pas franchir à l'ensemble de documents récupéré.

Sujet segmentation peut être utilisée dans la recherche d'information, dans lequel la tâche est de diviser un document en un ensemble de segments localement significatifs.

Harrag et autre. (2009) ont décrit deux expériences différentes sur un ensemble de données unique :

La recherche d'information avec une segmentation et sans segmentation.

Pour la récupération de l'information sans segmentation, les racines ont été extraites du texte de la requête pour calculer leur poids par TF-IDF.

En fait, TF-IDF a donné un poids déformé sur les termes rares dans l'ensemble de données Hadith.

Ensuite, l'indexation des termes de Hadiths pertinents, et le compte du poids de la pertinence ont été déterminées par leurs équations spécifiques.

Enfin, le système affiche deux ensembles de Hadiths : Pertinentes et non pertinentes. Dans la phase d'expérimentation, le rappel moyen était de 54%, et la précision était de 41% en utilisant un ensemble de données qui contient 453 Hadiths.

Pour la récupération de l'information avec la segmentation, l'algorithme de segmentation C99 **(Choi 2000)** a été utilisé pour l'évaluation, un texte segmenté a été préparé par des accords appariés entre sept experts humains en utilisant le coefficient Kappa.

Finalement, le système affiche une liste de segments triés en fonction de leur pertinence. Après avoir utilisé le sujet segmentation, une amélioration de 14% pour le rappel, et 51% pour la précision.

Comme souligné par **Aldhaln et autre. (2010)** dans leur article de revue, différents types d'informations peuvent être dérivées comme une forme de connaissance du Hadith,

Y compris législative islamique, L'Armée islamique, et la classification de Hadith.

Jbara (2010) a proposé un système d'extraction de texte pour récupérer une catégorie hadith en réponse à une requête. Dans ce cas, 1321 Hadiths du livre Sahih Al-Bukhari ont été préparés pour former et tester le système réparti sur treize groupes :

Foi, connaissance, prière, appel à la prière, éclipses, l'aumône, les bonnes manières, siam, médecine, nourriture, al hadj, les griefs et les vertus du Prophète Mohammad.

L'expérience comportait trois phases :

1. La première phase de prétraitement, qui consistait à enlever Isnad, tokenization, en supprimons les signes de ponctuation et diacritiques, les mots d'arrêt, et le stemming.

2. La deuxième phase a été utilisée pour la formation, dans laquelle la matrice de fonction (vocabulaires sont des caractéristiques) a été construite en utilisant TF-IDF.
3. La troisième phase de classification activée dans lequel l'ensemble de données d'entraînement résultant de l'étape précédente a été utilisée.

De plus, la requête comporte le calcul du poids et l'extension des requêtes exécutées pendant la troisième phase.

Finalement, la catégorie de prédiction a été effectuée avec une table de coefficients de similitude et de la méthode de similarité cumulée maximale, dans laquelle 45% et 49% de précision F1 score ont été obtenus.

Bilal et Mohsin (2012) introduit un système cloud et distribué les règle basée sur un système expert qui utilise la science Hadith pour les classer comme authentique et inauthentique, connu sous le nom Muhadith.

Les requêtes peuvent être fournies via une interface web telle que transposée dans Mouhadith.

Les cinq principaux modules du Mouhadith sont :

1. moteur d'inférence : un ensemble de if-then-else qui constitue les règles.
2. base de connaissances : tables de décision binaire pour représenter les connaissances.
3. analyseur et fait extracteur : utilisés pour les requêtes pairs fournis par les utilisateurs dans une première étape et utilisé pour extraire les informations pertinentes concernant la requête fournie.
4. explication installation : utilisé pour fournir des détails pour les utilisateurs au sujet de comment et pourquoi une conclusion a été rédigée.
5. Base de données.

Mouhadith est développé comme une architecture orientée services (SOA) système basé sur un expert Nuage (cloud) accessible par le web.

Aucun résultat expérimental ou une méthode d'évaluation rapportée dans le document.

L'entité nommée technique d'extraction a été utilisé pour identifier les entités utiles à partir d'un ensemble de données Hadith **(Harrag et al. 2011a ; Harrag 2014)**

Comme mentionné précédemment, Sahih Al-Bukhari a divisée en plus de 91 chapitres en fonction du sujet des Hadiths. Chaque chapitre est divisé en plusieurs sections, avec un total de 3882 sections. Sahih Al-Bukhari contient plus de 9000 hadiths, et chaque Hadith contient un certain nombre de Hadith, un Isnad (chaîne de narrateurs), un texte principal (Metn), et le début de la phrase (Taraf).

Pour transformer le texte non structuré dans un fichier texte semi-structuré, il a été défini un processus pour détecter les entités souhaitées :

Numéro de chapitre, le titre du chapitre, le numéro de section, titre de section, numéro Hadith, Isnad, Metn, Taraf, et la date.

Un modèle a été réalisé en utilisant un transducteur à états finis **(Roche and Shabes 1997)** sous la forme d'automates.

Les automates étaient représentés par un ensemble d'états et les transitions entre ces états, tandis qu'un texte a été lié à chaque état. Les automates tourné une séquence de vecteurs (c-à-dire les mots) en une séquence de symboles (c-à-dire les entités).

Le modèle encourageant obtenu la précision (71%), le rappel (39%), et F1-score (52%) des taux.

Il effectue l'identification des entités numériques comme le numéro de chapitre, le numéro de la section et le numéro de Hadith, il a mal exécuté afin de détecter l'entité de date parce que les dates ont été écrites en format alphabétique.

Le terme halal réfère à toute action ou un objet qui est permis où autorisé selon la loi islamique.

Beaucoup de scripts sur les produits hallal sont disponibles grâce aux ressources, telles que les pages Web, e-livres et des magazines.

L'utilisateur final peut enquêter sur les données liées halal par des expressions de requête pour chercher une liste de documents pertinents, Pour adresser ce scénario, **Hanum et autres** scruté techniques d'analyse de sujet, c’est-à-dire l’indexation sémantique latente (LSI) (Papadimitriou et autres 1998) et indexation inversée basée sur la fréquence.

L'expérience a été réalisée sur quatre documents malais Hadith traduits contenant 436 mots, et 36 autres documents en langue malaise.

Dans l'ensemble de données, 16 documents englobés divers aspects des sujets lier a la halal.

Pour obtenir un vecteur des mots, après que le tokenization et l'élimination des mots d'arrêt, toutes les tokens ont été converties en leur forme de racine utilisant un stemmer de langue malaise.

La similarité entre la requête et les documents a été mesurée en utilisant la technique de similarité cosinus **(Tata and Patel 2007).**

Cinq ensembles de requêtes, qui contenaient des mots sur les produits halals, ont été construits. Afin d'évaluer le succès de la technique, l'ensemble de données a été analysée manuellement et une liste des jugements pertinents a été compilée.

L'expérience montre que l'utilisation de LSI donne de meilleurs résultats que l'analyse de fréquence, c-à-dire 37% de précision et de 100% de rappel au mieux.

Cependant, LSI a besoin de ressources computationnelles élevées pour traiter des documents volumineux, telles que les pages Web.

Table 2 Overview of data mining approaches

No.	Approach published by	Methods	Extracted data	Data source	Performance
1	Shatnawi et al. (2012)	Web data extraction, and positional index	Extract Hadiths from web and determine their authenticity degrees: Sahih, Hasan, Daif, and Maudo	17,000 Hadiths from Sahih Al-Bukhari	Precision: 76 %, Recall: 42 %
2	Harrag et al. (2008) and Harrag and Hamdi-Cherif (2007)	Vector space model, TF-IDF, cosine similarity, enriched query	Relevant Hadith	60 Hadiths	Precision: 68 %, Recall: 80 %
3	Harrag et al. (2009)	C99 segmentation, TF-IDF	Relevant and irrelevant Hadiths	453 Hadiths	Precision: 92 %, Recall: 68 %
4	Jbara (2010)	TF-IDF and similarity coefficient table	13 Hadith groups e.g. faith, knowledge, praying, eclipses, alms, good manners, fasting	1321 Hadiths from the Sahih Al-Bukhari	Precision: 45 %, F1-score: 49 %
5	Bilal and Mohsin (2012)	Cloud based and distributed rule-based expert system	Authentic versus unauthentic	Not reported	Not reported
6	Harrag et al. (2011a) and Harrag (2014)	Finite state transducer	Chapter number, chapter title, section number, section title, Hadith number, Isnad, Matn, Taraf, and date	2602 Hadiths from Sahih Al-Bukhari	Precision: 71 %, Recall: 39 %, and F1-score: 52 %
7	Hanum et al. (2014)	LSI, TF-IDF, and cosine similarity	Halal related Hadiths	16 Malay translated Hadith documents	Precision: 37 %, Recall: 100 %

Figure 2.3 Overview of data mining approches. [24]

Ces chercheures compilent un corpus de E LHADITH en arabe et en anglais, ce corpus se compose de 452 hadiths, et 6696 hadiths en version anglaise, après avoir préparé leur corpus, ils ont comparu EL Hadith dans différents édition de Sahih Al-Bukhari et sélectionné EL hadith commun dans toutes les éditions.

D'après les travaux de recherche sur EL HADITH, nous avons remarqué que la taille des corpus utilisé est modeste pour la tâche de la classification. Nous avons également constaté qu'il n'existe pas une base de référence pour faire la comparaison entre les méthodes de classification.

Pour cette raison, nous avons décidé de compiler un corpus représentatif qui contient toutes les classes, la section suivante présente notre corpus collecté.

5 Description de notre corpus :

On dispose d'un corpus des documents textes prophétique de Sahîh Al-Boukhârî, ces textes prophétiques nous avons collecté du site موقع الإسلام الدعوي والارشادي http://hadith.al-islam.com.

Notre corpus, dit corpus de référence, est déjà classer pour nous aider à évaluer et mesurer la qualité des algorithmes de classification, nous avons opté pour l'utilisation de deux corpus différents du côté de nombre des classes.

5.1 Corpus 1 :

Ce corpus contient 510 documents textes Hadith de l'encyclopédie prophétique écrits en langue arabe, regroupés en 14 classes, le tableau suivant représente les différentes classes et le nombre des documents dans chacune.

Classe	Nombre des documents
الأحوال الشخصية	32
الأخلاق والأدب	39
الأشربة والأطعمة	39
الأقضية والأحكام	32
الايمان	31
الجنايات	31
الجهاد	32
الحج	42
الطب	53
العبادات	41
العلم	31
القران	33
اللباس والزينة	41
المعاملات	33

Table 2.1 : Le nombre de document dans chaque classe du corpus 1

5.2 Corpus 2 :

La collection des documents de ce corpus est aussi extraite de Sahîh Al-Boukhârî, ce corpus contient 205 documents textes Hadith de l'encyclopédie prophétique, regroupés en 4 classes, le tableau suivant représente les différentes classes et le nombre des documents dans chacune.

Classe	Nombre des documents
الأدب	73
الزكاة	48
الصلاة	38
الطعام	46

Table 2.2 : Le nombre de document dans chaque classe du corpus 2

6 Conclusion :

Dans ce chapitre, nous avons étudié le Corpus Prophétiques de L'imam Al-Boukhârî, leur historique et leurs caractéristiques, Nous avons aussi parlé sur les recherches faites dans le domaine de classification sur les corpus prophétiques (état de l'art précèdent) et leur résultats.

CHAPITRE 3 :

CLASSIFICATION DES TEXTES

1 Introduction :

La classification est une tâche très importante dans la fouille de donnée, elle consiste à créer un modèle qui peut être appliqué aux données, quand en a préparer notre base de données (nettoyage, remplissage, …), en peut appliquer une classification soit supervisé avec différentes méthodes ou non supervisée avec d'autres méthodes.

2 Classification :

La classification est la tâche la plus commune du Data Mining ou fouille de donnée et qui semble être une obligation humaine. Afin de comprendre notre vie quotidienne, nous sommes constamment classifiés, catégorisés et évalués.

La classification est un outil puissant d'exploration des données. Elle est parmi les taches les plus important du data mining, elle consiste à étudier les caractéristiques d'un nouvel objet pour lui attribuer une classe prédéfinie. Les objets à classifiés sont généralement des enregistrements d'une base de données, la classification consiste à mettre à jour chaque enregistrement en déterminant un champ de classe. La tâche de classification est caractérisée par une définition de classes bien précise et un ensemble d'exemples classés auparavant.

Leur objectif est de créer un modèle qui peut être appliqué aux données non classifiées dans le but de les classifiées.

Voici quelques exemples de l'utilisation des tâches de classification dans les domaines de recherche et commerce sont les suivants :

- Déterminer si l'utilisation d'une carte de crédit est frauduleuse.
- Diagnostiquant si une certaine maladie est présente.
- Déterminer quels numéros de téléphone correspondent aux fax.
- Déterminer quelles lignes téléphoniques sont utilisées pour l'accès à Internet.

3 Implémentation d'une classification :

- Choix de la mesure d'éloignement (dissimilarité, distance) entre les individus (généralement distance euclidienne).
- Choix du critère d'homogénéité des classes à optimiser (généralement inertie).
- Choix de la méthode utilisée : la Classification Ascendante Hiérarchique (CAH) ou celle par réallocation dynamique sont les plus utilisées.
- Mesure de la qualité de la classification.
- Choix du nombre de classes et leur interprétation.

La classification est une technique utilisée pour regrouper les objets dans des classes d'objets telles que :

- Les classes sont homogènes (les objets dans une classe ont des caractéristiques semblables)
- Chaque classe a des caractéristiques propres qui la différencient des autres classes.

On distingue deux types d'approches de classification ou d'apprentissage :

- Classification supervisé.
- Classification non supervisé.

3.1 Classification supervisé :

La classification supervisée, dite aussi discrimination est la tâche qui consiste à discriminer des données, de façon supervisée (c.-à-d. avec l'aide préalable d'un expert), un ensemble d'objets ou plus largement de données, de telle manière que les objets d'un même groupe (appelé classes) sont plus proches (au sens d'un critère de (dis) similarité choisi) les unes aux autres que celles des autres groupes. Généralement, on passe par une première étape dite d'apprentissage où il s'agit d'apprendre une règle de classification partir de données annotées (étiquetées) par l'expert et donc pour lesquelles les classes sont connues, pour prédire les classes de nouvelles données, pour lesquelles (on suppose que) les données sont inconnues. La prédiction est une tâche principale utilisée dans de nombreux domaines, y compris l'apprentissage automatique, la reconnaissance de formes, le traitement de signal et d'images, la recherche d'information, etc.

3.2 Classification Non supervisé :

Cette classification est aussi appelée "classification automatique", "clustering" ou encore "regroupement".

Dans ce type de classification on est amené à identifier les populations d'un ensemble de données. On suppose qu'on dispose d'un ensemble d'objets que l'on note par :

X = {x1, x2, ..., xN} caractérisé par un ensemble de descripteurs D.

L'objectif du clustering est de trouver les groupes auxquels appartiennent chaque objet x qu'on note par C = {C1, C2, ..., Cn}. Ce qui revient à déterminer une fonction notée Ys^{-} qui associe à chaque élément de X un ou plusieurs éléments de C. Il faut pouvoir affecter une nouvelle observation à une classe. Les observations disponibles ne sont pas initialement identifiées comme appartenant à telle ou telle population. [4]

4 Les Algorithmes de classification Non Supervisé :

La classification non supervisée ou « Clustering » est l'une des techniques fondamentales de l'extraction de données structurées ou non structurées.

Plusieurs méthodes ont été proposées :

- Classification hiérarchique : arbre de classes
- Classification hiérarchique ascendante : Agglomérations successives
- Classification hiérarchique descendante : Divisions successives
- Classification à plat : algorithme des k-moyennes : Partition, EM.

5 Les Algorithmes de classification Supervisé :

Il existe de nombreuses méthodes d'apprentissage supervisé [16] :

- K plus proches voisins (et ses variantes : Category-based Serch et Cluster-based-Search).
- Arbres de décisions.
- Naive Bayes.
- Réseaux de neurones.
- Machines à support de vecteurs (SVM).
- Programmation génétique.

Ce sont si méthodes-là qui seront utilisé dans notre travail :
K plus proches voisins(KNN), Machines à support de vecteurs (SVM) et Naïve Bayes.

5.1 K plus proche voisin :

L'algorithme de K plus proche voisin (PPV en bref, K Nearest Neighbor en anglais ou KNN) La méthode des K plus proches voisins est une méthode de l'apprentissage supervisé, dédiée à la classification qui peut être étendue à des tâches d'estimation. Est une méthode d'inférence inductive très efficace pour de nombreux problèmes pratiques. Il est robuste à des données d'entraînement bruyant, et tout à fait efficace lorsqu'il est fourni suffisamment un grand ensemble des données d'apprentissage. [35]

En prenant la moyenne pondérée des K plus proche voisin du point de requête, il peut lisser les effets des exemples isolés d'entraînement bruyants, contrairement à d'autres méthodes statistiques, ne nécessite aucun apprentissage (c'est-à-dire qu'il n'y a aucun modèle à ajuster). C'est l'échantillon d'apprentissage, associé à une fonction de distance et d'une fonction du choix de la classe en fonction des classes voisins les plus proches, qui constitue le modèle.

L'objectif de l'algorithme est de classé les exemples non étiquetés sur la base de leur similarité avec les exemples de la base d'apprentissage. Est une méthode de raisonnement à partir de cas.

Son principe est le suivant :

Une donnée de classe inconnue est comparée à toutes les données stockées. On choisit pour la nouvelle donnée la classe majoritaire parmi ses K plus proches voisins (Elle peut donc être lourde pour des grandes bases de données) au sens d'une distance choisie.

Comment identifier le K plus proches voisins ?

- Les instances sont des points dans un espace à *d*-dimensions
  d est le nombre d'attributs.
- Une instance xi est définie par son vecteur d'attributs.
  < a1(x1), a2(x2),..., ad (xi)>
- Chaque instance a également une catégorie *vi*.
- Identifier les voisins les plus proches de *xi*.
- Trouver les k instances ayant la plus petite distance dis (*xi* , *xj*).
- Similarité : une fonction inverse de la distance.

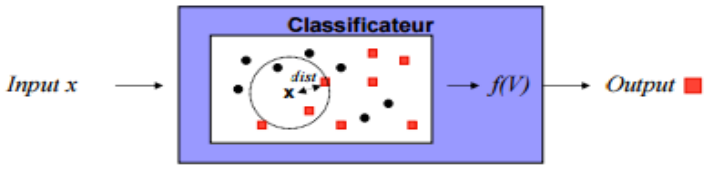

Figure 3.1 classificateur KNN [18]

Afin de trouver les K plus proches d'une donnée à classer, on peut choisir la distance euclidienne. Soient deux données représentées par deux vecteurs xi et xj, la distance entre ces deux données est donnée par :

$$\boldsymbol{dist}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \sqrt{\sum_{\boldsymbol{k=1}}^{\boldsymbol{d}} (\boldsymbol{x_{ik}} - \boldsymbol{x_{jk}})^2} \qquad (1)$$

Ou par La distance de Manhattan (valeurs continues) :

$$\boldsymbol{dist}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \sum_{\boldsymbol{r=1}}^{\boldsymbol{d}} \left| \left(\boldsymbol{a_r}(\boldsymbol{x_i}) - \boldsymbol{a_r}(\boldsymbol{x_j}) \right)^2 \right| \qquad (2)$$

Ou par La distance de Hamming (valeurs discrètes) :

$$\boldsymbol{dist}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \#\left\{ \boldsymbol{r} \in \boldsymbol{d} : \boldsymbol{a_r}(\boldsymbol{x_i}) \neq \boldsymbol{a_r x_j} \right\} \qquad (3)$$

L'algorithme de KNN est utilisé dans de nombreux domaines :

- La reconnaissance de formes.

- La recherche de nouveaux bio-marqueurs pour le diagnostic.
- Algorithmes de compression.
- Analyse d'image satellite.

5.2 Naïve Bayes :

Le classifieur naïf bayésien est l'une des méthodes les plus simples en apprentissage supervisé basée sur le théorème de Bayes. [31]

Ce théorème fournit une façon de calculer la probabilité conditionnelle d'une cause sachant la présence d'un effet, à partir de la probabilité conditionnelle de l'effet sachant la présence de la cause ainsi que des probabilités a priori de la cause et de l'effet. [37]

Il est peu utilisé par les praticiens du data mining au détriment des méthodes traditionnelles que sont les arbres de décisions ou les régressions logistiques.

Un avantage de cette méthode est la simplicité de programmation, la facilité d'estimation des paramètres et sa rapidité (même sur de très grandes bases de données). Malgré ses avantages, son peu d'utilisation en pratique vient en partie du fait que ne disposant pas d'un modèle explicite simple (l'explication de probabilité conditionnelle à priori), l'intérêt pratique d'une telle technique et remise en question. [11]

Qui repose sur une hypothèse simplificatrice forte : les descripteurs (Xj) sont deux à deux indépendants conditionnellement aux valeurs de la variable à prédire (Y). Pourtant, malgré cela, il se révèle robuste et efficace. Ses performances sont comparables aux autres techniques d'apprentissage. Cela, il se révèle robuste et efficace. Ses performances sont comparables aux autres techniques d'apprentissage. [34]

Dans l'approche bayésiennes, on mesure Pr [x|w], c'est-à-dire, la probabilité d'occurrence de l'évènement x si l'évènement w est vérifié : w joue le rôle d'une hypothèse préliminaire que l'on suppose vérifiée pour estimer la probabilité d'occurrence de l'évènement qui nous intéresse, noté x ici. [26]

Le théorème de Bayes :

Soient A, B et C trois évènements. Le théorème (ou règle) de Bayes démontre que :

Pr [A|B, C] = Pr [B|A, C] P r [A|C] P r [B|C]

Où :

- Pr [B|A, C] est la vraisemblance de l'évènement B si A et C sont vérifiés.
- Pr [A|C] est la probabilité a priori de l'évènement A sachant C.
- Pr [B|C] est la probabilité marginale de l'évènement B sachant C.
- Pr [A|B, C] est la probabilité a posteriori de A si B et C.

Dans cette formulation de la règle de Bayes, C joue le rôle de la connaissance que l'on a. [26]

5.3 Machines à support de vecteurs (SVM) :

Les machines à vecteurs de support, ou SVM (Support Vector Machines), est une méthode de classification binaire par apprentissage supervisé, sont une méthode relativement récente de résolution de problèmes de classification (trier des individus en fonction de leurs caractéristiques), SVM sont un algorithme dont de support dont le but est de résoudre les problèmes de discrimination à deux classes. On appelle problème de discrimination à deux classes dans lequel on tente de déterminer la classe à laquelle appartient un individu (individu ici employé au sens de constituant d'un ensemble) parmi deux choix possibles. [2]

SVM est une méthode de classification qui montre de bonnes performances dans la résolution de problèmes variés. Cette méthode a montré son efficacité dans de nombreux domaines d'applications tels que le traitement d'image, la catégorisation de textes ou le diagnostiques médicales et ce même sur des ensembles de données de très grandes dimensions. [23]

Parmi les modèles des SVM, on constate les cas linéairement séparable et les cas non linéairement séparable. Les premiers sont les plus simples de SVM car ils permettent de trouver facilement le classificateur linéaire. Dans la plupart des problèmes réels il n'y a pas de séparation linéaire possible entre les données, le classificateur de marge maximale ne peut pas être utilisé car il fonctionne seulement si les classes de données d'apprentissage sont linéairement séparables. [23]

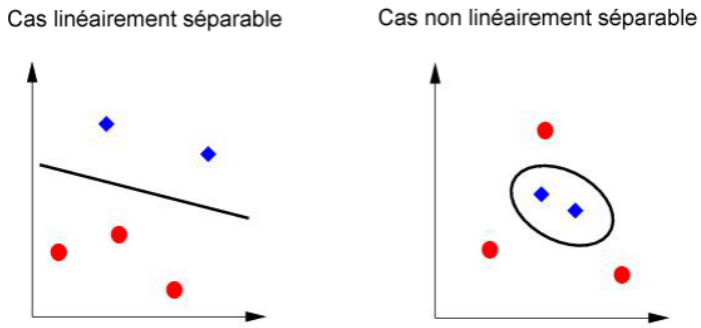

Figure 3.2 les cas linéairement séparable et les cas non linéairement séparable. [23]

6 Les critères de mesure des performances des algorithmes :

Afin d'évaluer les performances d'algorithmes de recherche d'information, les chercheurs se sont dotés d'outils de mesure auxquels appartiennent les taux de rappel de précision et de f-mesure. Ces critères de performance permettent de quantifier l'aptitude d'un système à trouver des résultats complets et pertinents.

6.1 Rappel :

Le rappel mesure la capacité du système à identifier tous les documents valides, il donne le pourcentage de réponses correctes renvoyées parmi tous les documents pertinents de la base de données. Ceci implique de connaitre effectivement toutes les réponses pertinents de la base, ce qui n'est pas réaliste pour des bases quelconques et n'est donc réalisable que sur des bases construites pour évaluer des systèmes de recherche. [1]

$$\text{Rappel}_{\,i} = \frac{\text{le nombre de Documents correctement attribués à la classe}_i}{\text{le nombre de documents appartenant à la classe}_{\,i}} \qquad (4)$$

6.2 Précision :

La précision mesure la capacité du système à trouver des documents valides. Elle donne le pourcentage de réponses correctes parmi les résultats obtenus. [1]

$$\boldsymbol{Précision_i = \frac{Le\ nombre\ de\ documents\ correctement\ attribués\ à\ la\ classe_{\,i}}{le\ nombre\ de\ documents\ classé\ par\ le\ système}} \quad (5)$$

Il est théoriquement possible d'avoir un rappel de 100% en renvoyant la liste de tous les documents de la base, mais la précision sera mauvaise et il sera difficile à l'utilisateur de gérer l'ensemble des résultats retournés. Si un seul résultat pertinent est renvoyé, la précision est excellente. Mais le rappel sera mauvais, Afin de quantifier un compromis, la F-mesure moyenne harmonique a été introduite.

6.3 F-mesure :

C'est la moyenne harmonique de la précision et du rappel Qui mesure la capacité du système. À donner toutes les solutions pertinentes et à refuser les autres, Une mesure populaire qui combine la précision et le rappel est leur pondération. [2]

$$\mathbf{F-mesure} = \frac{\mathbf{2.(précision.rappel)}}{\mathbf{(précision+rappel)}} \qquad (6)$$

6.4 Accuracy (exactitude) :

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

Les deux taux sont estimés comme suit :

$$\text{Acc} = \frac{\text{VP+VN}}{\text{VP+VN+FP+FN}} \qquad (7)$$

$$\text{Err} = \frac{\text{FP+FN}}{\text{VP+VN+FP+FN}} = 1 - \text{Acc} \qquad (8)$$

7 Techniques d'évaluation d'un classificateur :

Après un classificateur est construit, il doit être évalué l'exactitude, il existe de nombreuses façons pour évaluer un classificateur et il y a aussi de nombreuses mesures.

7.1 Ensemble des tests :

Est un ensemble d'exemples utilisés uniquement pour évaluer la performance d'un classificateur. [17]

7.2 Ensemble d'apprentissage :

Est utilisé pour l'apprentissage d'un classificateur. [17]

7.3 K-fold cross validation :

La validation croisée est une technique pour évaluer les modèles prédictifs en divisant l'échantillon d'origine dans un ensemble de formation pour former le modèle, et un critère établi pour l'évaluer.

Dans la validation croisée k fois, l'échantillon initial est divisé au hasard en k sous-échantillons de taille égale. Parmi les sous-échantillons k, un seul sous-échantillon est conservé comme les données de validation pour tester le modèle, et les k-1 sous-échantillons restants sont utilisées comme données d'entraînement. Le processus de validation croisée est ensuite répété k fois (les plis), avec chacun des sous-échantillons de k utilisées exactement une fois que les données de validation. Les k résultats des plis peuvent alors être en moyenne (ou autrement combinés) pour produire une seule estimation. L'avantage de cette méthode est que toutes les observations sont utilisées à la fois la formation et de validation, et chaque observation est utilisé pour la validation exactement une fois. [10]

8 Conclusion :

Dans ce chapitre on a présenté la tâche de la classification et leur importance dans la fouille de donnée, on a aussi situé les types d'apprentissages automatique qui se résument en deux types, apprentissage supervisé qui inclut plusieurs méthodes et algorithmes de classification nous avons aperçu les méthodes les plus connus (Naive Bayes,KNN et SVM).

CHAPITRE 4 :

LES TACHES DE PRETRAITEMENT D'UN TEXTE

1 Introduction :

Après la collecte de l'ensemble de données, une étape de prétraitement doit être faite avant la classification, Dans l'étape de prétraitement, des tâches multiples doit être faite pour assurer le nettoyage des données et la suppression du bruit qui peuvent affecter la précision de la classification.

2 Prétraitement :

Les données textuelles dont la forme particulière de donnée complexes. Elles ne sont pas délimitées, structurées et étiquetées sémantiquement de façon explicite. En conséquence ces données nécessitent un traitement préalable.

De manière générale, l'objectif de prétraitement est de minimiser l'espace de recherche.

Le prétraitement du texte arabe est une étape difficile et importante. [36]

Il peut avoir un impact positif ou négatif sur la précision de tout système d'information de récupération. Et par conséquent l'amélioration de l'étape de prétraitement emmène nécessairement à l'amélioration de tout système d'information de récupération très fortement.

Le prétraitement contient de nombreux sous-processus et chacun à un travail spécifique pour préparer les données pour être en forme optimale donc le résultat peut être amélioré. Le système proposé se focus sur les étapes de prétraitement suivantes :

- Tokenisation.
- Normalisation.
- Lemmatisation.
- Le Stemming.
- La suppression de mots vides.
- Generate n-grame.

2.1 Tokenization :

Tokenization est une étape nécessaire dans le traitement du langage naturel, les systèmes de recherche d'exploitation de données et d'information.

Tokenization est le processus de décomposé les documents en mots, des phrases, des symboles ou d'autres éléments significatifs appelés Token. suppression de la ponctuation, chiffres et les chaines de caractères sans espaces sont considérées comme des mots (termes,tokens),ce qui emmène à la suppression du caractères tels que : *, # , { , }... etc.

L'objectif de la Tokenization est l'exploration des mots dans une phrase la principale utilisation de tokenization est d'identifier les mots clé significatifs, Elle est basée

principalement sur les espaces blancs et les signes de ponctuation comme des séparateurs entre les mots comme la figure suivante.

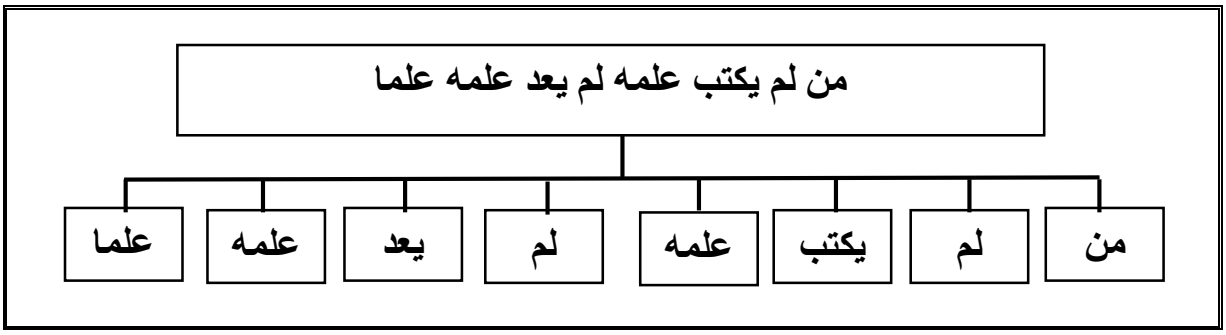

Figure 4.1 : Processus Tokenization.

2.1.1 Token et Terme :

- Token : est une instance d'une séquence de caractères dans un document particulier sont regroupés en une unité sémantique utile pour le traitement.

''من''، ''لم''، ''يكتب''، ''علمه''، ''لم''، ''يعد''، ''علمه''، ''علما''.

- Terme : est un type qui est inclus dans le dictionnaire des systèmes de recherche d'information.

''يكتب''، ''علمه''، ''يعد'' ، ''علما''.

2.2 Normalisation :

Conversion des caractères en majuscule ou minuscule.

- Pour la langue arabe : ـأ-ـآ- ـا -> ا.
- Pour la langue française : é, è -> e, élevé, élève -> eleve.
- Suppression tachkil (mouvement kasra, fatha, tanwin, chadda...).
- Suppression tatwil (كتـــــــــــــــاب), شكرااااا .

2.3 Lemmatisation :

Consiste à trouver la racine des verbes fléchis et à ramener les mots pluriels et/ou féminins à la forme masculine singulier.

Permet le regroupement des formes morphologiques d'une même unité linguistique en seule unité appelé lemma, (Obtention de la forme canonique).

Par exemple :

Soit le mot **العلم**, ce dernier est décomposé en préfixe « **ال** » est lemma « **علم**»

2.4 Stemming :

2.4.1 Le stemming ou la désuffixation :

Qui associer plusieurs mots ayant la même racine, c'est-à-dire enlever les suffixes des mots pour ne conserver que la partie racine, en s'aidant des algorithmes simples basés sur des règles de remplacement de chaine de caractères pour supprimer les suffixes les plus utilisés.

2.4.2 Light stemming :

Light stemming (ou la racinisation légère en français) est un processus d'enlèvement de préfixes et/ou suffixes qui génère une pseudo-racine (stem en anglais), sans se préoccuper des infixes ou de reconnaître les schèmes (patterns en anglais) pour retrouver la racine.

Exemple de Stemming :

Mots	الكتاب	الكاتب	المكتبة
Racine	كتب		

Exemple de light stemming :

Mots	الكتاب	الكاتب
Light stemming	كتاب	كاتب

2.5 Suppression de mots vides :

Les mots qui apparaissent le plus souvent dans un corpus sont généralement les mots grammaticaux, mots vides (empty words) ou mots outils (stop words) : les prépositions, les mots de liaisons, les déterminants, les adverbes, les adjectifs indéfinis, les conjonctions, les pronoms et les verbes auxiliaires etc..., qui constituent une grande part des mots d'un texte, mais malheureusement sont faiblement informatifs, sur le sens d'un texte puisqu'ils sont présents sur l'ensemble des textes.

Dans la langue arabe on peut citer :

حروف العطف، حروف الجر، أسماء الإشارة، اخوات كان ...

2.6 Generate n-grame :

On définira un n-gram de caractères par une suite de n caractères ou par une suite de n termes, ou n est la longueur du n-gramme : b grams pour n=2, tri-grams pour n=3, quadri-grams pour n=4 ... etc. [14]

Par exemple, on donne len-gramme de caractère pour le mot suivant :

« الحديث »

n=3	الح- لحد-حدي-ديث
n=4	الحد-لحدي-حديث

3 Pondération ou calcule du poids :

Pour comparer les termes d'une façon plus efficace nous utilisons un système de pondération, un exemple commun d'utilisation de l'apprentissage automatique réside dans le calcul des poids basés sur la fréquence des mots. La mesure TF.IDF (Term Frequency, Inverse Document Frequency), couramment employée dans la recherche d'information (RI), cette mesure permet d'évaluer l'importance d'un terme contenu dans un document relativement à une collection ou un corpus. [14]

Le TF.IDF d'un terme *i* d'un corpus, est défini par :

$$TF.IDF_{i,j} = TF.log_2 \frac{N}{n}$$

Où TF est la fréquence d'un terme *i* dans un document *j* ; N est le nombre total de documents du corpus et n le nombre de document dans lesquels apparait le terme *i*.

$$TF = \frac{n_{ij}}{N_{ij}}$$

Où IDF (la fréquence inverse de document) est une mesure de l'importance générale du terme (obtenu en divisant le nombre total de documents par le nombre de documents contenant le terme, puis en prenant le logarithme de ce quotient).

$$IDF = Log \frac{|D|}{|\{dj: ti \ \epsilon \ dj\}|}$$

Où :

$|D|$: Le nombre total de documents dans le corpus.

$|\{dj: ti \ \epsilon \ dj\}|$: Le nombre de documents ou de termes ti apparait

Exemple de TF-IDF :

Considérons un document contenant 100 mots où le mot الأدب apparaît 3 fois.

Selon les formules définies précédemment, la fréquence à long terme (TF) pour le mot (الأدب) est ensuite 0,03. TF =3/100 = 0.03.

Supposons que nous avons 10 millions de documents et mot (الأدب) apparaît dans un des milliers de ceux-ci.

La fréquence inverse de document (IDF) est calculée comme :

Log (10 000 000/1 000) = 4.

Le score de TF-IDF est le produit de ces quantités : 0,03 × 4 = 0,12.

4 Conclusion :

Pour appliquer les différents algorithmes d'apprentissage sur les données textuels, il y a un ensemble de techniques et des opérations préliminaires doivent être faites pour épurer le texte de tous les mots inutiles et conserver seulement le textes ceux qui sont porteurs d'informations et utiles pour le processus de classification, et les différentes étapes de prétraitement sont exposées dans ce chapitre.

CHAPITRE 5 :

RESULTATS ET ANALYSES

1 Introduction :

Dans ce chapitre, nous présentons et analysons les différents résultats expérimentaux de chaque algorithme (Naive bayes, KNN et SVM) sur les deux corpus des textes prophétiques cité dans le chapitre 2, et on à comparer les résultats obtenus avec les mesures de classification connus (F-mesure moyenne et Accuracy), après ces comparaisons nous avons définie quel est le meilleur algorithme de classification pour notre corpus.

2 Outils de classification de textes :

Nous utilisons le RapidMiner, RapidMiner (anciennement YALE (Yet Another Learning Environment)) est un environnement pour l'apprentissage de la machine et l'expérimentation d'exploration de données, Il permet aux expériences d'être composé d'un grand nombre d'opérateurs arbitrairement emboîtables. Les opérateurs sont décrits dans des fichiers XML qui sont créés avec l'interface utilisateur graphique de RapidMiner.

RapidMiner est utilisé pour la recherche et les tâches d'exploration de données du monde réel. RapidMiner fournit plus de 1000 opérateurs pour toutes les procédures d'apprentissage machine principales, incluant les entrées et les sorties, et le prétraitement des données et la visualisation. Il est écrit dans le langage de programmation Java et peut donc fonctionner sur tous les systèmes d'exploitation populaires.Il intègre également des programmes d'apprentissage et les attributs évaluateurs de l'environnement d'apprentissage Weka. (Weka est disponible sur le site : https://sourceforge.net/projects/weka/files/weka-3-6/3.6.12/)

Le Processus Documents de fichiers est un opérateur de RapidMiner que génère des vecteurs de mots à partir d'une collection de texte stocké dans plusieurs fichiers. [22]

RapidMiner est disponible sur le site : https://rapidminer.com/products/studio , ou l'on aussi trouve des tutoriaux, des vidéos et des blogs.

Dans la section suivante, nous présentons la partie teste et leurs résultats commentés.

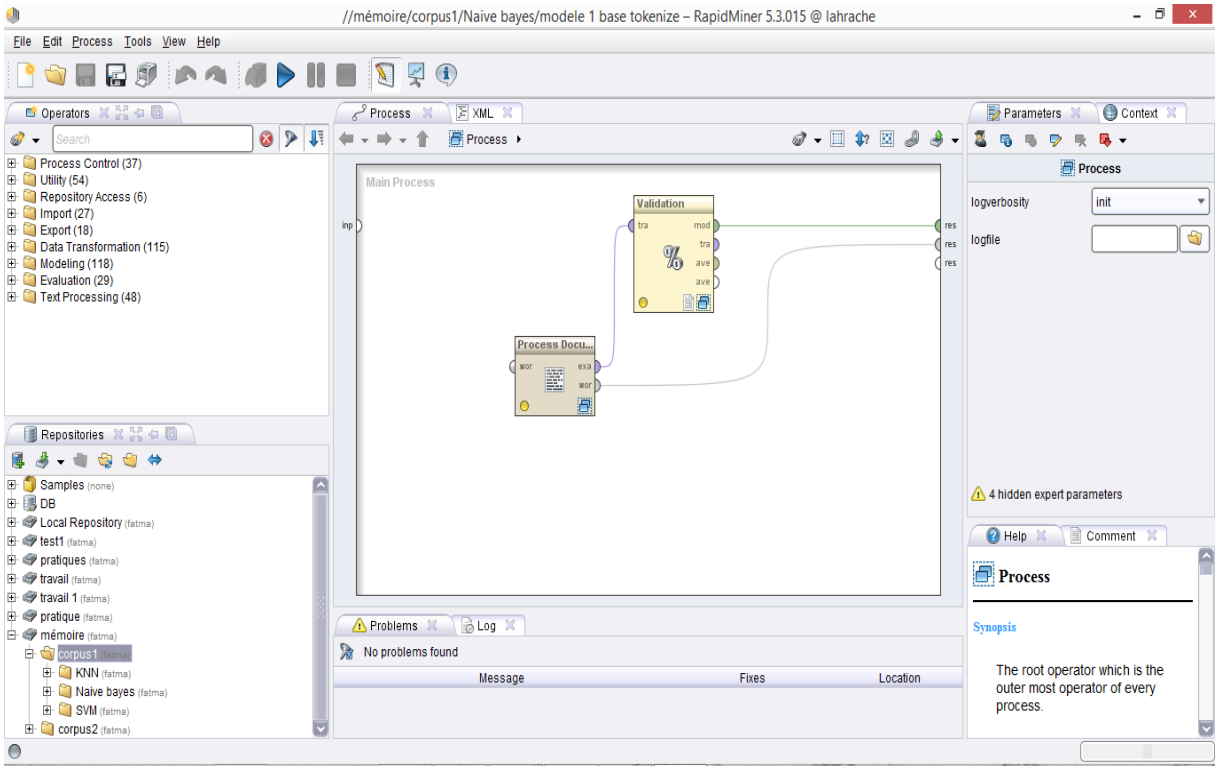

Figure 5.1 L'interface de RapidMiner.

3 Les Résultats Expérimentaux :

Les tableaux suivants contiennent les différents modèles et les différents résultats de chaque algorithme appliqués sur les 2 corpus de textes prophétiques, le corpus 1 qui contient 14 classes (الجهاد, الحج, الطب, الايمان, الجنايات, الأقضية والأحكام, الأشربة والأطعمة, الأخلاق والآداب,الأحوال الشخصية, المعاملات, اللباس والزينة, العلم, القرآن,العبادات) ce corpus contient 510 hadiths et le deuxième corpus contient 4 classes (الطعام, الصلاة, الزكاة,الأدب) qui contient 205 documents hadiths.

On a 7 modèles dans chaque corpus :

1. **modèle1.a :** base tokenization et **modèle1.b :** tokenization+ stopwords.
2. **modèle2.b :** tokenization+ stopwords+stemarabic et **modèle2.a :** tokenization+stemarabic.
3. **modèle3.b :** tokenization+ stopwords+stemlight et **modèle3.a :** tokenization+stemlight.
4. **modèle4.b :** tokenize +stopwords+ngram=3 et **modèle4.a :** tokenize + ngram=3.
5. **modèle5.b :** tokenize +stopwords+ngram=4 et **modèle5.a :** tokenize + ngram=4.
6. **modèle6.b :** tokenize +stopwords+ngram=3+keepterms et **modèle6.a :** tokenize +ngram=3+keepterms.
7. **modèle7.b :** tokenize +stopwords+ngram=4+keepterms et **modèle7.a :** tokenize +ngram=4+keepterms.

3.1 Résultat de Corpus 1 : les tableaux suivants présents les différents résultats du corpus1. Les résultats en gras définie les meilleur résultats dans chaque modèles.

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0,5292*	*0,5031*	***0,5383***	*0,5169*
SVM		***0,5323***	***0,5195***	*0,5343*	***0,5225***
KNN	**Distance Euclidienne K=1**	*0,4930*	*0,4745*	*0,5029*	*0,4912*
	Distance Euclidienne K=10	*0,4578*	*0,4133*	*0,4619*	*0,4344*
	Similarité Cosine K=1	*0,4949*	*0,4743*	*0,5068*	*0,4912*
	Similarité Cosine K=10	*0,4548*	*0,4146*	*0,4618*	*0,4344*

Table 5.1 Modèle1.a : Base Tokenization et Modèle1.b : Tokenization+Stopwords

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0,5191*	*0,5010*	*0,5304*	*0,5109*
SVM		***0,6745***	***0,6791***	***0,6772***	***0,6811***
KNN	**Distance Euclidienne K=1**	*0,5490*	*0,5538*	*0,5559*	*0,5675*
	Distance Euclidienne K=10	*0,6021*	*0,6004*	*0,6028*	*0,6048*
	Similarité Cosine K=1	*0,5490*	*0,5538*	*0,5559*	*0,5675*
	Similarité Cosine K=10	*0,6288*	*0,6004*	*0,6028*	*0,6048*

Table 5.2 : Modèle2.b : Tokenization+Stopwords+Stemarabic et Modèle2.a : Tokenization+Stemarabic

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0,5390*	*0,5150*	*0,5463*	*0,5208*
SVM		***0,6307***	***0,6156***	***0,6380***	***0,6224***
KNN	**Distance Euclidienne K=1**	*0,5202*	*0,4933*	*0,5324*	*0,5090*
	Distance Euclidienne K=10	*0,5033*	*0,5082*	*0,5166*	*0,5284*
	Similarité Cosine K=1	*0,5200*	*0,4933*	*0,5324*	*0,5090*
	Similarité Cosine K=10	*0,5040*	*0,5082*	*0,5166*	*0,5284*

Table 5.3 : Modèle3.b :Tokenization+Stopwords+Stemlight et Modéle3.a :Tokenization+Stemlight

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0,5241*	*0,5287*	*0,5285*	*0,5344*
SVM		***0,6918***	***0,6830***	***0,6909***	***0,6830***
KNN	**Distance Euclidienne K=1**	*0,6024*	*0,5954*	*0,6166*	*0,6107*
	Distance Euclidienne K=10	*0,5943*	*0,6101*	*0,5948*	*0,6126*
	Similarité Cosine K=1	*0,6024*	*0,5954*	*0,6166*	*0,6107*
	Similarité Cosine K=10	*0,5943*	*0,6101*	*0,5948*	*0,6126*

Table 5.4 Modèle4.b : Tokenization+Stopwords+ngarmchar=3 et Modèle4.a : Tokenization+ngramchar=3

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0,5709*	*0,57*	*0,5776*	*0,5794*
SVM		***0,6398***	***0,62***	***0,64***	***0,6244***
KNN	**Distance Euclidienne K=1**	*0,55361*	*0,54*	*0,5676*	*0,5598*
	Distance Euclidienne K=10	*0,54*	*0,55*	*0,5438*	*0,5555*
	Similarité Cosine K=1	*0,55*	*0,54*	*0,5676*	*0,5598*
	Similarité Cosine K=10	*0,55*	*0,56*	*0,5477*	*0,5575*

Table 5.5 : Modèle5.b : Tokenization+Stopwords+ngramchar=4 et Modèle5.a :Tokenization+ngramchar=4

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0.57*	*0.56*	*0,5696*	*0,5599*
SVM		***0.67***	***0.66***	***0,6674***	***0,6577***
KNN	**Distance Euclidienne K=1**	*0.58*	*0.57*	*0,5970*	*0,5853*
	Distance Euclidienne K=10	*0.60*	*0.60*	*0,6028*	*0,6047*
	Similarité Cosine K=1	*0.58*	*0.57*	*0,5970*	*0,5853*
	Similarité Cosine K=10	*0.60*	*0.60*	*0,6028*	*0,6047*

Table 5.6 : Modèle6.b : Tokenization+Stopwords+ngramchar=3+keepterms et Modèle6.a : Tokenization+ngramchar=3+keepterms

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0.58*	*0.57*	*0,5833*	*0,5756*
SVM		***0.61***	***0.63***	***0,6048***	***0,6204***
KNN	**Distance Euclidienne K=1**	*0.55*	*0.54*	*0,5618*	*0,5539*
	Distance Euclidienne K=10	*0.54*	*0.54*	*0,5360*	*0,54*
	Similarité Cosine K=1	*0.55*	*0.54*	*0,5618*	*0,5539*
	Similarité Cosine ++ K=10	*0.54*	*0.54*	*0,5399*	*0,5419*

Table 5.7 : Modèle7.b : Tokenization+Stopwords+ngramchar=4+Keepterms et Modèle7.a : Tokenization+ngramchar=4+keepterms

Nous avons comparé tous les modèle afin d'arriver à un résultat qui donne F-mesure moyenne et Accurancy dans le corpus1, ces comparaisons basées sur les critères suivants :

1. Le meilleur F-mesure moyenne et le meilleur Accurancy dans chaque algorithme et chaque modèle.
2. Le meilleur algorithme dans tous les modèles.
3. Le meilleur modèle entre (3-gram+keepterms) et (3-gram sans keepterms).
4. Le meilleur modèle entre (4-gram+keepterms) et (4-gram sans keepterms).
5. Le meilleur F-mesure moyenne et Acurancy pour chaque algorithme.

Après la comparaison de tous les tableaux ci-dessus on a les résultats suivant :

- Depuis le tableau **5.4** qui contient le **modèle4.b** (Tokenization+stopwords+ngramchar=3) et le **modèle4.a** (Tokenization+ngramchar=3) on a obtenu le meilleur résultat de F-mesure 0,6918 et le meilleur Accurancy 0,6909. On constate que le meilleur algorithme et le meilleur modèle qui sont le **SVM** et le **modèle4.b** (Tokenization+stopwords+ngramchar=3).
- Depuis les résultats de F-mesure moyenne et Accurancy qu'on a obtenu dans les différents modèles. On a remarqué que le meilleur algorithme appliqué sur le corpus 1 c'est le **SVM**.
- Après la comparaison entre les deux tableaux **5.4** et **5.6** , tableau **5.4** qui contient le **modèle4.b** (tokenization+stopwords+ngramchar=3) et le **modèle4.a**(tokenization+ngramchar=3), tableau **5.6** qui contient le **modèle6.b** (tokenization+stopwords+ngramchar=3+keepterms) et le **modèle6.a** (tokenization+ngramchar=3+keepterms), On a trouvé que les résultats du tableau **5.4** (F-mesure moyenne = 0,6918 et Acurancy = 0,6909 pour le modèle 7), est meilleur que

les résultats du tableau **5.6** (F-mesure moyenne=0,67 et Accurancy=0,6674 pour le **modèle6.b**), donc le meilleur modèle c'est le **modèle4.a** (tokenization+stopwords+ngramchar =3).

- Après la comparaison entre les deux tableaux **5.5** et **5.7**, tableau **5.5** qui contient le **modèle5.b** (tokenization+stopwords+ngramchar=4) et le **modèle5.a**(tokenization+ngramchar=4), tableau **5.7** qui contient le **modèle7.b**(tokenization+stopwords+ngram=4+keepterms) et le **modèle7.a** (tokenization+ngramchar=4+keepterms). On a trouvé que les résultats du tableau **5.5** (F-mesure moyenne = 0,6389 et Accurancy = 0,64 pour le **modèle5.b**) est meilleur que les résultats de tableaux **5.7** (F-mesure moyenne =0,63 et Accurancy = 0,6204 pour le **modèle7.a**), donc le meilleur modèle c'est le **modèle5.b** (tokenization+stopwords+ngramchar=4).
- Quand on compare les résultats de F-mesure moyenne et Accurancy qu'on a obtenu dans tous les modèles pour chaque algorithme on est arrivé aux résultats suivants :

- **Pour l'algorithme Naive Bayes :** Le meilleur F-mesure moyenne est 0,58 dans le modèle7.b (tokenize +stopwords+ngram=4+keepterms) et le meilleur Accurancy est 0,5833 dans le modèle 7.b (tokenize +stopwords+ngram=4+keepterms).
- **Pour l'algorithme SVM :** le meilleur F-mesure moyenne est 0,6918 dans le modèle4.b (tokenize +stopwords+ngram=3) et le meilleur Accurancy est 0,6909 dans le modéle4.b (tokenize +stopwords+ngram=3).
- **Pour l'algorithme KNN :**
  - **Dans la distance Euclidienne K=1** le meilleur F-mesure moyenne est 0,6024 dans le **modèle4.b** (tokenize +stopwords+ngram=3) et le meilleur Accurancy est 0,6166 aussi dans le **modèle4.b**.
  - **Dans la distance Euclidienne K=10** le meilleur F-mesure moyenne est 0,6101 dans le **modèle4.a** (tokenize + ngram=3) et le meilleur Accurancy est 0,6126 aussi dans le **modèle4.a**.
  - **Dans la distance similarité de Cosine K=**1 le meilleur F-mesure moyenne est 0,6024 dans le **modèle4.b** (tokenize +stopwords+ngram=3) et le meilleur Accurancy est 0,6166 aussi dans le **modèle4.b**.

- **Dans la distance similarité de Cosine K=10** le meilleur F-mesure moyenne est 0,6288 dans le **modèle2.b** (tokenization+ stopwords+stemarabic) et le meilleur Accurancy est 0,6126 dans le **modèle4.a** (tokenize + ngram=3).

3.2 Résultat de corpus 2 : les tableaux suivants présents les différents résultats du corpus2.

Les résultats en gras définies les meilleures résultats dans chaque modèle.

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		***0,7329***	***0,7329***	***0,7321***	***0,7321***
SVM		*0,6908*	*0,7096*	*0,6938*	*0,7121*
KNN	**Distance Euclidienne K=1**	*0,7313*	*0,6913*	*0,6874*	*0,6874*
	Distance Euclidienne K=10	*0,5905*	*0,6112*	*0,5860*	*0,6207*
	Similarité Cosine K=1	*0,7032*	*0,6913*	*0,6983*	*0,6874*
	Similarité Cosine K=10	*0,6383*	*0,6112*	*0,5660*	*0,6207*

Table 5.8 : Modèle1.a : Base tokenization et Modèle 1.b :Tokenization+Stopwords

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0,6874*	*0,6974*	*0,6931*	*0,6983*
SVM		***0,7941***	***0,7971***	***0 ,7955***	***0,8002***
KNN	**Distance Euclidienne K=1**	*0,7091*	*0,7332*	*0,7167*	*0,7417*
	Distance Euclidienne K=10	*0,6861*	*0,6725*	*0,6738*	*0,6790*
	Similarité Cosine K=1	*0,7091*	*0,7332*	*0,7167*	*0,7417*
	Similarité Cosine K=10	*0,6861*	*0,6725*	*0,6738*	*0,6790*

Table 5.9 :Modèle2.b :Toeknization+Stopwords+stemarabic et Modèle2.a:Tokenization+Stemarabic

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		***0,7349***	*0,6912*	*0,7319*	*0,6931*
SVM		*0,7329*	***0,7394***	***0,7381***	***0,7429***
KNN	**Distance Euclidienne K=1**	*0,6505*	*0,6781*	*0,6502*	*0,6781*
	Distance Euclidienne K=10	*0,6355*	*0,6771*	*0,6243*	*0,6743*
	Similarité Cosine K=1	*0,6526*	*0,6781*	*0,6502*	*0,6781*
	Similarité Cosine K=10	*0,6266*	*0,6771*	*0,6148*	*0,6743*

Table 5.10 : Modèle3.b :Tokenization+Stopwords+Stemlight et Modèle3.a : Tokenization+Stemlight

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0,7202*	*0,7151*	*0,7176*	*0,7181*
SVM		***0,8392***	***0,8412***	***0,8360***	*0,5352*
KNN	**Distance Euclidienne K=1**	*0,7115*	*0,7364*	*0,7069*	***0,7317***
	Distance Euclidienne K=10	*0,7023*	*0,6949*	*0,7031*	*0,6890*
	Similarité Cosine K=1	*0,7115*	*0,7364*	*0,7069*	*0,7317*
	Similarité Cosine K=10	*0,7023*	*0,6949*	*0,7031*	*0,6890*

Table 5.11 : Modèle4.b : Tokenization+Stopwords+ngramchar=3 et Modèle4.a :Tokenization+ngramchar=3

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0,7455*	*0,7385*	*0,7424*	*0,7379*
SVM		***0,7743***	***0,7426***	***0,7671***	***0,7424***
KNN	**Distance Euclidienne K=1**	*0,7147*	*0,7185*	*0,7133*	*0,7176*
	Distance Euclidienne K=10	*0,6714*	*0,7096*	*0,6748*	*0,7031*
	Similarité Cosine K=1	*0,7147*	*0,7185*	*0,7133*	*0,7176*
	Similarité Cosine K=10	*0,6743*	*0,7092*	*0,6693*	*0,7031*

Table 5.12 : Modèle5.b :Tokenization+Stopwords+ngramchar=4 et Modèle5.a :Tokenization+ngramchar=4

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0,7320*	*0,7366*	*0,7329*	*0,7376*
SVM		***0,7930***	***0,8098***	***0,7869***	***0,8012***
KNN	**Distance Euclidienne K=1**	*0,7135*	*0,7341*	*0,7071*	*0,7271*
	Distance Euclidienne K=10	*0,7229*	*0,7412*	*0,7224*	*0,7376*
	Similarité Cosine K=1	*0,7135*	*0,7341*	*0,7071*	*0,7271*
	Similarité Cosine K=10	*0,7229*	*0,7412*	*0,7224*	*0,7376*

Table 5.13 :Modèle6.b :Tokenization+Stopwords+ngramchar=3+keepterms et Modèle6.a :Tokenization+ngramchar=3+keepterms

Algorithmes		*F-mesure moyenne*		*Accurancy (Exactitude)*	
		Avec StopWords	*Sans StopWords*	*Avec StopWords*	*Sans StopWords*
Naive bayes		*0,6993*	*0,7358*	*0,7029*	*0,7331*
SVM		***0,7631***	***0,7428***	***0,7571***	***0,7374***
KNN	**Distance Euclidienne K=1**	*0,7254*	*0,7089*	*0,7233*	*0,7083*
	Distance Euclidienne K=10	*0,7132*	*0,7390*	*0,7081*	*0,7276*
	Similarité Cosine K=1	*0,7254*	*0,7089*	*0,7233*	*0,7083*
	Similarité Cosine K=10	*0,6840*	*0,7390*	*0,6738*	*0,7276*

Table 5.14 : Modèle7.b :Tokenization+Stopwords+ngramchar=4+Keepterms et Modèle7.a:Tokenization+ngramchar=4+Keepterms

Nous avons comparé tous les modèle afin d'arriver à un résultat qui donne F-mesure moyenne et Accurancy dans le corpus2.

Après la comparaison de tous les tableaux ci-dessus on a les résultats suivant :

- Depuis le tableau **5.11** qui contient le **modèle4.b** (Tokenization+stopwords+ngramchar=3) et le **modèle4.a** (Tokenization+ngramchar=3) on a obtenu le meilleur résultat de F-mesure 0,8412 et le meilleur Accurancy 0,8360. On constate que le meilleur algorithme et le meilleur modèle qui sont le **SVM** et le **modèle4.b**.
- Depuis les résultats de F-mesure moyenne et Accurancy qu'on a obtenu dans les différents modèles. On est arrivé que le meilleur algorithme appliqué sur le corpus 2 c'est le SVM.
- Après la comparaison entre les deux tableaux **5.11** et **5.13**, tableau **5.11** qui contient le **modèle4.b** (tokenization+stopwords+ngramchar=3) et le **modèle4.a** (tokenization+ngramchar=3), le tableau **5.13** qui contient le **modèle6.b** (tokenization+stopwords+ngramchar=3+keepterms) et le **modèle6.a** (tokenization+ngramchar=3+keepterms), On a trouvé que les résultats du tableau **5.11** (F-mesure moyenne = 0,8412 et Acurancy = 0,8360 pour le **modèle4.b**) est meilleur que les résultats du tableau **5.13** (F-mesure moyenne=0,8098 et Accurancy=0,8012 pour le **modèle6.b**).Donc le meilleur modèle c'est le **modèle4.b** (tokenization+stopwords+ngramchar =3).
- Après la comparaison entre les deux tableaux **5.12** et **5.14**, tableau **5.12** qui contient le **modèle5.b** (tokenization+stopwords+ngramchar=4) et le **modèle5.a** (tokenization+ngramchar=4), tableau **5.14** qui contient le **modèle7.b** (tokenization+stopwords+ngram=4+keepterms et le **modèle7.a**

(tokenization+ngramchar=4+keepterms).On a trouvé que les résultats du tableau **5.12** (F-mesure moyenne = 0,7743 et Accurancy = 0,7671 pour le **modèle5.b**) est meilleur que les résultats de tableaux **5.14** (F-mesure moyenne =0,7631 et Accurancy = 0,7571 pour le **modèle7.a**), donc le meilleur modèle c'est le modèle 9 (tokenization+stopwords+ngramchar=4).

- Quand on compare les résultats de F-mesure moyenne et Accurancy qu'on a obtenu dans tous les modèles pour chaque algorithme on est arrivé aux résultats suivants :
  - Pour l'algorithme Naive Bayes : Le meilleur F-mesure moyenne est 0,7455 dans le **modèle5.b** (tokenize +stopwords+ngram=4) et le meilleur Accurancy est 0,7424 aussi dans le **modèle5.b**.
  - Pour l'algorithme SVM : le meilleur F-mesure moyenne est 0,8412 dans le **modèle4.a** (tokenize + ngram=3) et le meilleur Accurancy est 0,8360 dans le **modèle4.b** (tokenize +stopwords+ngram=3).
  - Pour l'algorithme KNN :
  - Dans la distance Euclidienne K=1 le meilleur F-mesure moyenne est 0,7364 dans le **modèle4.a** (tokenize + ngram=3) et le meilleur Accurancy est 0,7417 dans le **modèle2.a** (tokenization+ stopwords).
  - Dans la distance Euclidienne K=10 le meilleur F-mesure moyenne est 0,7412 dans le **modèle6.a** (tokenize + ngram=3+keepterms) et le meilleur Accurancy est 0,7376 aussi dans le **modèle6.a**.
  - Dans la distance similarité de Cosine K=1 le meilleur F-mesure moyenne est 0,7364 dans le **modèle4.a** (tokenize + ngram=3) et le meilleur Accurancy est 0,7417 dans le **modèle2.a**.
  - Dans la distance similarité de Cosine K=10 le meilleur F-mesure moyenne est 0,7412 dans le **modèle6.a** (tokenize + ngram=3+keepterms) et le meilleur Accurancy est 0,7376 aussi dans le **modèle6.a** (tokenization+ stemarabic).

Après la comparaison de tous les résultats obtenus dans les deux corpus, nous ne concluons que ces résultats suivants :

Corpus 1 : contient 14 classes.

- Le meilleur algorithme appliqué sur ce corpus c'est le SVM.

- Les meilleurs résultats trouvés dans les 7 modèles appliqué sur le corpus 1 à démontrer que l'utilisation du stopwords est utile.
- L'utilisation de 3gram et 4gram sans keepterms nous a donné un meilleur résultat.
- Les résultats obtenus du F-mesure moyenne et Accurancy quand on a appliqué le 3gram est meilleure que l'utilisation de 4gram.

Corpus 2 : contient 4 classes.

- Le meilleur algorithme appliqué sur ce corpus c'est le SVM.
- Les meilleurs résultats trouvés dans les 7 modèles appliqué sur le corpus 2 à démontrer que l'utilisation des modèles sans stopwords est utile.
- L'utilisation de 3gram et 4gram sans keepterms nous a donné un meilleur résultat.
- Les résultats obtenus du F-mesure moyenne et Accurancy quand on a appliqué le 3gram est meilleure que l'utilisation de 4gram.

De toutes ces études on a constaté que les meilleurs algorithmes de classification appliquée sur les textes prophétique sont : SVM (ce résultat est attendu vu que les recherches sur la classification des textes arabes ont montré que SVM donne les meilleurs résultats), ensuite l'algorithme de Naive bayes, KNN (similarité de cosine, k=10) et le meilleur modèle c'est 3gram sans keepterms (ce résultat aussi attendu vu que les recherches ont montré que 3gram donne les meilleurs résultats du la classification des textes arabes)

4 Conclusion :

Dans ce chapitre on a testé les 3 algorithmes d'apprentissage supervisé (Naive bayes, KNN, SVM) et appliqué les différents étapes du prétraitement sur les textes prophétiques, on a également comparé les résultats obtenus pour conclure le meilleure algorithme appliqué et le meilleur modèle sur le texte prophétique.

Conclusion Générale :

Dans le cadre de ce mémoire, ont c'est basé sur l'étude des différents méthodes de classification des connaissances sur les deux corpus des différents nombres de classes qui contiennes des textes prophétique de Sahih Elboukhari.

On a fait une étude comparative entre ces deux corpus dont on a appliqué les trois méthodes de classification supervisé (Naive bayes, KNN et SVM), pour chaque méthode on a appliqué des différents modèles (stem arabic, stemlight, n-gram), après avoir appliqué ces méthodes on à comparer les résultats obtenus avec les mesures de classification (F-mesure moyenne et Accurancy).

Après ces études nous nous sommes intéressés dans ce mémoire à comparer les différents résultats obtenu par l'environnement de l'apprentissage RapidMiner.

Pour les perspectives de ce travail. Nous proposons pour l'amélioration de cette étude les points suivants :

- Collecter un corpus plus large à partir de Sahih Elboukhari et autres (Imam Mouslem).
- Appliquer autres algorithmes de classifications (Réseaux de neurones,).
- Pour enrichir les textes prophétiques, l'utilisation des dictionnaires ontologie parait une solution prometteuse.
- Combiner les techniques de classification non supervisés (K-means, E-M...) avec les techniques de catégorisation pour atteindre les meilleurs résultats.
- Améliorer les performances de classification en utilisant les techniques de sélection d'attributs (Feature Selection).

Bibliographie :

Ouvrage :

[1] CAMPEDEL Marine, HOOGSTOËL Pierre,Marine Campedel,Pierre Hoogstoël , Sémantique et multimodalité en analyse de l'information] LAVOISIER,2011] .

[3] Dong, Guozhu, Pei, Jian, Sequence Data Mining, Springer Edition, 2007].

[14] Juan-Manuel Torres-Moreno ,Résumé automatique de documents, LAVOISIER/,rue Lavoisier 75008 paris /ISBN 978-2-7462-3212-9 /ISSN 1968-8008 .

[17] Liu,Bing ,Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data ,Second edition ;2011 .

[20] Michael J. A. Berry Gordon S. Linoff , Mastering Data Mining: The Art and Science of Customer Relationship Management,1st edition 2000.

[21] Michael J. A. Berry, Gordon S. Linoff, Data Mining Techniques For Marketing, Sales, and Customer Relationship, Management, Second Edition, april 2004.

[29] S. PRABHU, N. VENKATESAN, Data Mining and Warehousing, New Age International (P) Ltd., Publishers, New Delhi, 2007.

[35] Tom M. Mitchell, Machine Learning, (March 1, 1997 .

Article :

[2] Dominik francoeur ,Machines A Vecteurs de support une introduction /CaMUS 1 (2010).

[4] Fatma Karem∗ , Mounir Dhibi∗ Arnaud Martin , Combinaison de classification supervisée et non-supervisée par la théorie des fonctions de croyance « ch3 » « article ».

[5] G. DONG, J. PEI, Sequence Data Mining, Springer Edition, 2007.'ch1' « cours ».

[15] Kanaan G., Al-Shalabi R., Ghwanmeh S., "A comparison of text-classification techniques applied to Arabic text", Journal of the American Society for Information Science and Technology, 60(9), pp. 1836 – 1844, 2009.

[16] Laurent denoue, classification suppervisé de document , 2011,pdf « ch3 » « article ».

[18] Luc Lamontagne ,Apprentissage a base d'exemple / Concepts avancés pour systèmes intelligents .

[22] Mierswa I., Wurst M., Klinkenberg R., Scholz M., Euler T., "YALE: Rapid Prototyping for Complex Data Mining Tasks", *in the Proc. of the 12th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining KDD-06*, 2006.

[23] Mohamadally Hasan Fomani ,SVM : Machines à Vecteurs de Support ou Séparateurs à Vastes Marges ,16 janvier 2006 .

[24] Mohammad Arshi Saloot1 · Norisma Idris1 · Rohana Mahmud1 · Salinah Ja'afar1 · Dirk Thorleuchter2 · Abdullah Gani1, Hadith data mining and classification: a comparative analysis, 8 January 2016ch2 ». « article ».

[25] Osmar R. Zaian, Principles of Knowledge Discovery in Databases, CMPUT690, University of Alberta, 1999.

[26] Ph. PREUX , Fouille de données Notes de cours Université de Lille 3 ,26 mai 2011].

[27] Ph. PREUX, Fouille de données : Notes de cours, Université de Lille 3, 9 octobre 2008.

[28] Pio NardielloAffiliated withMercurioWeb SNC, Fabrizio Sebastiani, Alessandro Sperduti, Discretizing Continuous Attributes in AdaBoost for Text Categorization, Volume 2633 of the series Lecture Notes in Computer Science pp 320-334, 15 April 2003,,» « ch1 » « article ».

[30] Said D., Wanas N., Darwish N., Hegazy N., "A Study of Arabic Text preprocessing methods for Text Categorization", In the 2nd Int. conf. on Arabic Language Resources and Tools, Cairo, Egypt, 2009.

[31] Sebastian Raschka ,Naive Bayes and Text Classification I Introduction and Theory/ October 4, 2014 .

[32] Taghva, K., Elkhoury, R., Coombs, J., "Arabic stemming without a root dictionary", Information Technology: Coding and Computing, ITCC, Vol. 1, pp. 152 – 157, 2005.'

[33] Taïeb Baccouche L'Information Grammaticale Année 1998 Volume 2 Numéro 1 pp. 49-54 Fait partie d'un numéro thématique : Numéro spécial Tunisie .

[34] Tanagra_Naive_Bayes_Classifier_Explained.pdf.

[36] Waad A. Al-Harbi1 and Ahmed Emam Ph.D, EFFECT OF SAUDI DIALECT PREPROCESSING ON ARABIC SENTIMENT ANALYSIS ; ISSN:2319-7900.

Mémoire :

[37] Yasmine Hanane zeggane Mokhtar ,Algorithmesd'apprentissage pour la classification de documents ,Université de Mostaganéme -Algérie- - licence 2009] .

[19] Matallah hocine ; classification automatique de textes approche orientée agent ; UNIVERSITE ABOUBEKR BELKAID-TLEMCEN FACULTE DES SCIENCES DEPARTEMENT D'INFORMATIQUE .

[37] Z Simon,Outils classificatoires par objets pour l'extraction de connaissances dans les bases de donnée.thése de doctorat de l'université Henri Poincaré-Nancy 1,Nancy,2000.

Site web :

[7] http://america.pink/arabic-diacritics_442541.html.

[8] http://scholarpedia.org/article/Text_categorization» .

[9] http://www.iqrashop.com/Sahih-Al-Boukhari-arabe-francais-Al-imam-Zain-oud-Din-Ibn Abdoullatif-Az-Zoubaidi-Livre-livres-Hadiths-p597-.html .

[10] http://www.openml.org/a/estimation-procedures/1 .

[11] http://www.r-bloggers.com/classifieur-naif-bayesien/ .

[12] https://abjadia.wordpress.com/tag/alif-wasla.

[13] https://rapidminer.com/products/studio/.

ملخص:

في هذا البحث يتم اجراء دراسة مقارنة بين مجموعتين من الأحاديث النبوية لصحيح البخاري مختلفة من حيث العدد الكلي للأحاديث، والتي طبقنا عليها أساليب التصنيف المختلفة (Naive bayes,KNN,SVM) بعد تطبيق هذه الأساليب تمت مقارنة النتائج المتحصل عليها على أساس مقاييس التصنيف (F-mesure moyenne, Accurancy).

أظهرت النتائج أن أفضل مصنف مطبق على الأحاديث النبوية هو SVM كما أظهرت أن أفضل نموذج مطبق هو 3gram دون إبقاء الشروط.

للقيام بالتصنيف نستخدم RapidMiner.

الكلمات المفتاحية:

الأحاديث النبوية، التصنيف، اللغة العربية 3gram.، SVM، Naive bayes، KNN .

Résumé :

Dans ce mémoire, une étude comparative est faite sur deux corpus de textes prophétiques de Sahih Elboukhari de différent nombre de classe, dont on a appliqué les méthodes de classification (Naive bayes, KNN et SVM), après avoir appliqué ces méthodes on a comparé les résultats obtenus sur la base de mesures de classification (F-mesure moyenne et Accurancy).

Les résultats obtenus montrent que le meilleur algorithme de classification appliquée sur les textes prophétiques est le SVM (Support Vector Machine) et le meilleur modèle est 3gram sans keepterms.

Pour la classification on a utilisé l'environnement de RapidMiner.

Les mots clés :

Classification, Naïve bayes, KNN, SVM, Text prophétique, 3gram, Langue Arabe.

Abstract :

In this thesis, a comparative study is made on two corpuses of prophetic texts of Saheeh Elboukhari of various number of class, the methods of classification of which we applied (Naive Bayes, KNN and SVM), after having applied these methods were compared the results obtained on the basis of measures of classification (average F-measure and Accurancy).

The obtained results show that the best algorithm of classification applied on the prophetic texts is the SVM (Support Vector Machine) and the best model 3gram without keepterms.

For the classification. we used the environment of RapidMiner.

Keywords :

Classification, Naive bayes, KNN, SVM, Prophetic Texts, 3gram, Arabic Language.