

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE MINISTÈRE DE
L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
Université de Djilali BOUNAAMA Khemis Miliana



FACULTÉ DES SCIENCES ET DE LA TECHNOLOGIE DÉPARTEMENT DE MATHÉMATIQUES
ET D'INFORMATIQUE

MÉMOIRE PRÉSENTÉ

POUR L'OBTENTION DE DIPLÔME DE

Master en Informatique

Option : INGÉNIERIE DU LOGICIEL ET SYSTÈMES DISTRIBUÉS

TITRE :

Vers un système d'aide à l'automatisation des réponses aux requêtes de Fatwas islamiques

Réalisé par :
BENHADJ AMAR Bilel
METRITER Rofaida

Devant le jury composé de :
Mr D. Bahloul : ENCADRANT

ANNÉE UNIVERSITAIRE 2019/2020

Remerciements

Nous tenons tout d'abord à remercier Allah le tout puissant, le très miséricordieux et le créateur de toutes choses, qui nous a doté d'intelligence, et nous a maintenu en santé pour mener à bien cette année d'étude. de nous avoir donné le courage, la force et la patience d'achever ce modeste travail.

Ce travail est un moyen de gratitude pour ses innombrables grâces et pour s'acquitter de notre responsabilité de partager les connaissances en Islam.

Nous estimons à notre humble avis qu'il serait impérieux de souligner que la réalisation de ce présent travail n'est pas un simple fait relevant des efforts binomiaux, mais par contre, le fruit des efforts de plusieurs personnes qui de près ou de loin ont consenties pour que cette œuvre soit une réalité.

Nous voudrions adresser nos sincères et chaleureux remerciements à notre promoteur de recherche Monsieur « Djamel BAHLOUL » dont nous avons eu la chance de l'avoir comme encadrant et qui a bien voulu nous confier ce travail riche d'expériences et nous guider dans chaque étape de sa consécration.

Nous lui exprimons ici nos remerciements pour ses constantes orientations de notre recherche, ainsi que pour ses précieux conseils, sa disponibilité et son extrême amabilité malgré sa grande charge de travail.

Le mérite d'un mémoire appartient certes à l'auteur, mais également à son promoteur qui l'encadre. Notre promoteur a été d'un soutien et d'une attention exceptionnels qu'elle nous a témoignée pour nous permettre de mener à bien ce travail.

Nous souhaitons aussi adresser nos remerciements aux membres du jury pour nous avoir fait l'honneur de juger cette thèse et examiner notre travail et de l'enrichir par leurs propositions. Veuillez accepter l'expression de nos vives gratitude.

Nous tenons à saisir cette occasion et adresser nos profonds remerciements et nos profondes reconnaissances à tous le corps professoral, administratif et pédagogique de département mathématique et informatique à l'université de Djilali BOUNAAMA à Khemis Miliana, ses qualités scientifiques et éducatifs ont abouti à la réussite de nos études universitaires.

Nous remercions également BELARAIBI Walid et SETTI Zakariya, leur ensemble de données a été un coup de pouce initial à notre mise en œuvre.

Enfin, nous tenons à remercier toutes les personnes qui ont contribué de près ou de loin, d'une manière directe ou indirecte à l'élaboration de ce travail de fin d'études, nous vous présentons nos remerciements, notre respect et gratitude.

C'est certes avec joie et fierté que nous déposons aujourd'hui ce mémoire, mais aussi avec un brin de nostalgie que nous termine nos études universitaires et nous concluons ce deuxième travail de recherche.

En espérant que ce mémoire répond à vos espérances et reflète l'ambition de ses écrivains, nous vous souhaitons une bonne lecture.

Dédicaces

*Ce travail est l'aboutissement d'un dur labeur et de beaucoup de sacrifices, que je
veux dédier à :*

*Tous les membres de ma famille paternelle « BENHADJ AMAR » et maternelle
« HADJADJI » qui m'ont gratifié de leurs amours et leurs motivations. Je leurs
adresse toute ma gratitude du fond du cœur, en espérant que ce sera une fierté
familiale en plus d'être professionnel.*

*Mes très chers sœurs « Zineb » et « Rim » et parents pour leur amour, leurs
conseils ainsi que leur soutien inconditionnel, à la fois moral et financier, qui
m'a permis de réaliser les études que je voulais et par conséquent ce mémoire.
Le meilleur encadreur que j'ai l'honneur de travaillé avec, je veux aborder une fois
de plus les soutiens de toutes formes dont j'ai bénéficié de votre part, vos qualités
méritent toute ma gratitude, que ce travail soit pour nous une joie partagée.*

*Tous les amis et collègues avec qui j'ai eu la chance d'étudier tout au long de ma
démarche universitaire, spécialement à CHERRANI Toufik et DILMI Abdelhakim
pour leurs compagnons agréables et leurs concurrences de travail.*

*Enfin, je dédie mon binôme de projet « Rofaida » qui a ma gratitude, j'ai eu
beaucoup de plaisir à travailler avec vous, j'admire ton esprit et persévérance de
travail.*

*Je vous souhaite tous mes meilleurs vœux de bonheur, santé et réussite dans vos
vie professionnelle et personnelle.*

Dédicaces

Avec l'expression de ma reconnaissance, je dédie ce modeste travail à ceux qui, quels que soient les termes embrassés, je n'arriverais jamais à leur exprimer mon amour sincère.

A l'homme, mon précieux offre du dieu, qui doit ma vie, ma réussite et tout mon respect : mon cher père Mourad.

A la femme qui a souffert sans me laisser souffrir, qui n'a jamais dit non à mes exigences et qui n'a épargné aucun effort pour me rendre heureuse : mon adorable mère BENKAHLA Salima.

A mes chères sœurs Lina , Alae qui n'ont pas cessée de me conseiller, encourager et soutenir tout au long de mes études. Que Dieu les protège et leurs offre la chance et le bonheur.

A mon adorable petit frère Mohamed moulay qui sait toujours comment procurer la joie et le bonheur pour toute la famille.

A mes grands-mères, mes oncles et mes tantes. Que Dieu leur donne une longue et joyeuse vie.

A tous les cousins, les voisins et les amis que j'ai connu jusqu'à maintenant.

Merci pour leurs amours et leurs encouragements.

Sans oublier mon binôme Bilel pour son soutien moral, sa patience et sa compréhension tout au long de ce projet.

Merci

Résumé

La plupart des questions posées à propos de la charia islamique (autrement dites « demandes de Fatwa ») ont déjà été répondues par les Muftis au fil du temps. Etant donné difficile de contacter directement les Muftis pour avoir les réponses immédiates et adéquates à des questions particulières, l'utilisation intelligente des Fatwas précédentes semble une très bonne solution. En d'autre part, les Muftis et dans plusieurs situation répètent les mêmes Fatwas pour des questions similaires.

Le but de cette recherche est de d'éviter cette répétition en mettant en place un système qui répond automatiquement aux questions fatwas posées par les musulmans lors d'un échange de langue naturelle arabe.

La tâche de ce système est de classer la question dans son sujet et de rechercher des questions similaires dans une base de données à l'aide de modèles d'intelligence artificielle, afin de fournir une réponse à la question.

Plusieurs systèmes ont été mis en œuvre dans ce domaine, en utilisant divers paradigmes de réponse aux questions, mais seulement deux ont utilisé une approche d'apprentissage automatique. [43]

Les modèles d'intelligence artificielle sont capables de fournir des résultats instantanés et très précis sans aucune intervention humaine.

Outre l'utilisation de modèles d'apprentissage automatique, ce travail est le premier à utiliser un modèle d'apprentissage profond dans le domaine des systèmes de réponse aux requêtes de Fatwas islamiques.

L'approche proposée est composée de deux principaux modules, un premier concerne le routage des demandes de Fatwa dans les catégories adéquates de la Chriaa et le deuxième répond concrètement à la demande en faisant une recherche basée sur la similarité sémantique dans les bases de données existantes.

Mots clés : Systèmes de Question-Réponse, Classification de textes, Langue arabe, Routage de Fatwa, Apprentissage automatique, Apprentissage profond, Fatwa islamique.

Abstract

Most of the questions asked about Islamic Sharia (otherwise known as “Fatwa’s demands”) have already been answered by the Muftis over time. Given the difficulty of contacting Muftis directly for immediate and adequate answers to specific questions, the intelligent use of previous Fatwas seems a very good solution. On the other hand, Muftis in several situations repeat the same Fatwas for similar questions.

The goal of this research is to surpass that repetition by implementing a system which automatically answers Fatwas questions asked by muslims during an exchange of arabic natural language.

The task of that system is to classify the question in its topic and search for similar questions in a database using artificial intelligence models, in order to provide an answer to the question.

Several systems were implemented in this domain, using various question answering paradigms, but only two of them used machine learning approach. [43]

Artificial intelligence models are capable of providing instant and very accurate results without the need of any human intervention.

Besides using machine learning models, this work is the first one to use a deep learning model in the domain of fatwas question answering systems (as far as we know).

The proposed approach is composed of two main modules, a first concerns the routing of Fatwa requests in the appropriate categories of the Chriaa and the second responds concretely to the request by doing a search based on semantic similarity in existing databases.

Keywords : Question Answering system, Text classification, Arabic, Fatwa routing, Machine learning, Deep learning, Islamic Fatwa.

الملخص

معظم الأسئلة المطروحة حول الشريعة الإسلامية (المعروفة باسم « مطالب الفتوى ») سبق أن أجاب عنها المفتون مع مرور الوقت. نظرا لصعوبة الاتصال بالمفتين مباشرة للحصول على إجابات فورية وكافية لأسئلة محددة ، فإن الاستخدام الذكي للفتاوى السابقة يبدو حلا جيدا للغاية. من ناحية أخرى ، يكرر المفتون وفي عدة مواقف نفس الفتاوى لأسئلة مماثلة.

الهدف من هذا البحث هو تجاوز ذلك التكرار من خلال تطبيق نظام يجب تلقائيا على أسئلة الفتاوى التي يطرحها المسلمين أثناء تبادل اللغة العربية.

الهدف من هذا المشروع هو تصنيف السؤال في موضوعه والبحث عن أسئلة مماثلة له في قاعدة بيانات باستخدام نماذج التعلم الآلي ، من أجل تقديم إجابة على السؤال.

تم تنفيذ العديد من الأنظمة في هذا المجال ، باستخدام نماذج مختلفة للإجابة على الأسئلة ، ولكن اثنين فقط منها استخدموا نهج التعلم الآلي. [٤٣]

نماذج الذكاء الاصطناعي قادرة على تقديم نتائج فورية ودقيقة للغاية دون الحاجة إلى أي تدخل بشري. إلى جانب استخدام نماذج التعلم الآلي ، يعد هذا العمل هو الأول الذي يستخدم نموذج التعلم العميق في مجال أنظمة الإجابة على أسئلة الفتاوى (على حد علمنا).

يتألف النهج المقترح من وحدتين رئيسيتين ، الأولى تتعلق بإدراج طلبات الفتوى في الفئات المناسبة من الشريعة والثانية تستجيب بشكل مختص للطلب من خلال البحث على أساس التشابه الدلالي في قواعد البيانات الموجودة.

الكلمات المفتاحية : أنظمة الإجابة على الأسئلة ، تصنيف النصوص ، اللغة العربية ، توجيه الفتوى ، التعلم الآلي ، التعلم العميق ، الفتوى الإسلامية

Table des matières

Table des matières	8
Table des figures	9
Liste des tableaux	10
Liste des Abréviations	11
1 État de l'art sur les systèmes de question-réponse	14
1.1 Généralités sur les systèmes de questions-réponses SQR	14
1.2 Qu'est-ce qu'une question ?	14
1.3 Qu'est-ce qu'une réponse ?	15
1.4 A propos de la question-réponse	16
1.5 Les domaines de système de questions-réponse	19
1.6 Taxonomies des systèmes de question réponse	20
1.7 Architecture générique d'un SQR	24
1.8 Les approches de système questions-réponses	26
1.9 Système de question-réponse Arabe	28
1.10 Système de question-réponse de Fatwa Islamique	32
2 Classification de textes et Apprentissage profond	36
2.1 Apprentissage profond	36
2.2 Les réseaux de neurones	38
2.3 Classification de texte	48
2.4 Apprentissage de similarité	57
3 Contribution et implémentation	63
3.1 Architecture globale (fonctionnement)	63
3.2 Architecture de formation	64
3.3 Interface graphique	76
4 Expérimentations et déploiement	78
4.1 Métriques d'évaluation	78
4.2 Evaluation des algorithmes	81
4.3 Proposition d'un modèle de déploiement	89
Bibliographie	

Table des figures

1.1	Intersection de la question-réponse avec différents domaines de recherche.	16
1.2	Recherche d'information.	18
1.3	Architecture générique d'un système de Questions/Réponses. [73]	25
1.4	Evolution de la question-réponse en arabe depuis son apparition. [52]	29
2.1	Réseaux de Neurones Biologique. [60]	38
2.2	Modèle d'un neurone artificiel. [60]	39
2.3	Représentation matricielle du modèle d'un neurone artificiel. [60]	41
2.4	Fonctions d'activations : (a) du neurone « seuil »; (b) du neurone « linéaire », et (c) du neurone « sigmoïde ». [60]	42
2.5	Schéma d'un réseau de neurones monocouche. [60]	44
2.6	Schéma d'un réseau de neurones non bouclé (Perceptron multicouches). [60]	45
2.7	Schéma d'un réseau de neurones à connexions locales. [60]	45
2.8	Schéma de réseau de neurones bouclé. [60]	46
2.9	Le modèle de Kohonen. [60]	47
2.10	La séparation linéaire entre la classe A et B. [60]	48
2.11	Classification de texte. [37] [76]	50
2.12	Les techniques d'apprentissage. [76]	50
2.13	Techniques d'apprentissage automatique. [37]	51
2.14	Processus de formation d'un classifieur de texte d'apprentissage automatique. [37]	52
2.15	Étapes de classification du texte. [76]	52
2.16	La fonction de cosinus. [5]	59
2.17	La distance cosinus. [5]	59
3.1	Architecture de fonctionnement.	63
3.2	Préparation de corpus.	65
3.3	Architecture de formation des modèles de routage.	66
3.4	Exemple d'un corpus déséquilibré. [24]	67
3.5	Distribution de données par classe.	67
3.6	Architecture de formation des modèles de similarité.	71
3.7	Interface graphique représentent le résultat d'une requête aléatoire.	76
4.1	Précision et rappel. [29]	79
4.2	Précision et justesse. [1]	80
4.3	Les justesses des modèles de classification.	81
4.4	Matrice de confusion de Classificateur de vecteur de support linéaire.	82
4.5	Matrice de confusion de Régression logistique.	83
4.6	Matrice de confusion de Naïf bayes multinomiaux.	84
4.7	Matrice de confusion de Classificateur de forêt aléatoire.	85
4.8	Page de soumission des questions.	90
4.9	Page de confirmation de classification et choix des modèles.	90
4.10	Page de présentation des résultats.	91

Liste des tableaux

1.1	Classification des systèmes de question-réponse proposée par [75]. [52]	23
1.2	Exemple de questions répondues par JAWEB. [52]	31
2.1	Analogie entre le neurone biologique et le neurone formel. [60]	40
2.2	Différentes fonctions d'activations utilisées dans les RNA. [60]	42
3.1	Paramètres de vectoriseur des modèles de classification.	68
3.2	Représentation d'une question avec le vectoriseur TF-IDF.	68
3.3	Paramètres d'algorithme de Support de vecteur linéaire. [35]	69
3.4	Paramètres d'algorithme de régression logistique. [33]	69
3.5	Paramètres d'algorithme Naïf bayes multinomiaux. [17]	70
3.6	Paramètres d'algorithme de classificateur de forêt aléatoire. [32]	71
3.7	Paramètres généraux des modèles de similarité.	72
3.8	Représentation d'une question avec le modèle de sac des mots.	72
3.9	Taille des dictionnaires des modèles de similarité.	72
3.10	Paramètres d'algorithme tf-idf. [23]	73
3.11	Paramètres d'algorithme LSI. [22]	73
3.12	Paramètres de modèle LDA. [21]	75
3.13	Paramètres d'algorithme du modèle doc2vec. [20]	75
4.1	Matrice de confusion. [8]	78
4.2	Comparaison des justesses par modèle.	81
4.3	Métriques d'évaluation de Classificateur de vecteur de support linéaire.	82
4.4	métriques d'évaluation de Régression logistique.	83
4.5	Métriques d'évaluation de Naïf bayes multinomiaux.	84
4.6	Métriques d'évaluation de modèle classificateur de forêt aléatoire.	85
4.7	Comparaison de précision des modèles de similarité (classe de Hadj).	86
4.8	Comparaison de précision des modèles de similarité (classe de Salat).	86
4.9	Comparaison de précision des modèles de similarité (classe de Sawm).	86
4.10	Comparaison de précision des modèles de similarité (classe de Zakat).	86
4.11	Comparaison de précision des modèles de similarité.	87
4.12	Comparaison de combien la première sortie d'un modèle avait la plus grande similitude entre les autres modèles.	87
4.13	Comparaison de taux maximum de similarité eu dans chaque corpus.	87
4.14	Comparaison de combien la première sortie d'un modèle avait la plus faible similitude entre les autres modèles.	88
4.15	Comparaison de taux minimum de similarité eu dans chaque corpus.	88
4.16	Comparaison des différents résultats selon le corpus de test de Hadj.	88
4.17	Comparaison des différents résultats selon le corpus de test de Salat.	88
4.18	Comparaison des différents résultats selon le corpus de test de Sawm.	89
4.19	Comparaison des différents résultats selon le corpus de test de Zakat.	89

Liste des Abréviations

AQAS question-answering system.

AQUASYS Arabic QUestion-Answering SYStem.

ArabiQA Arabic Question Answering.

CA Classical Arabic.

DefArabicQA Arabic Definition Question Answering System.

Doc2Vec Modèle Vectoriel de Paragraphe.

EM Espérance-Maximisation.

FAQ Foire Aux Questions.

FQAS Fatwa Question Answering Systems.

IA Intelligence Artificielle.

IDRAAQ Information and Data Reasoning for Answering Arabic Questions.

JAWEB web-based Arabic question answering application system.

KNN k-Nearest Neighbours.

LDA Allocation Dirichlet Latente.

LSI Indexation sémantique latenteL.

MSA Modern Standard Arabic.

NLP Natural Language Processing.

PHP Hypertext Preprocessor.

PICO System for Natural Langage GEneration of Answers.

POS Part-of-speech.

QARAB Arabic Question Answering System.

QASAL Question- Answering System for Arabic Language.

RDF Resource Description Framework.

RI Recherche d'Information.

RNA Réseaux de neurones Artificiels.

SQR systèmes de questions-réponses.

SVM Support Vector Machine.

SYNGE-A SYstem for Natural langage GEneration of Answers.

TALN Traitement Automatique du Langage Naturel.

Tf-Idf Fréquence du terme * Fréquence inverse du document.

WWW World Wide Web.

Introduction générale

La Fatwa dans la communauté musulmane est considérée comme l'un des besoins importants et nécessaires de l'individu et de la société, elle représente un phare d'orientation par lequel un musulman peut se tenir sur ses décisions, en plus de connaître la récompense et la punition, et les conséquences pour toute action accomplie dans la vie actuelle et son sort dans l'au-delà.

À cette époque, les moyens de transmettre la Fatwa aux gens variaient ; Ainsi, les gens reçoivent des Fatwas par le biais de la radio, de la télévision, des chaînes satellite et les forums d'internet à travers le monde, par l'intermédiaire desquelles le questionneur devrait accéder pour poser sa question, et le mufti lui répond. ce processus est entièrement manuel, c'est pourquoi le temps de réponse peut varier indéfiniment.

En plus, beaucoup de gens posent des questions auxquelles les muftis ont déjà répondu au fil du temps, la répétition des questions conduit à prolonger le temps de réponse et à empêcher les muftis de répondre aux nouvelles questions.

Les travaux réalisés dans le cadre de ce mémoire sont liés à proposer et réaliser un système qui aide la communauté des musulmans à trouver les réponses adéquates à leurs demandes de Fatwa sans avoir recourir à un contact direct avec les Muftis.

L'approche proposée tirera profit de la puissance des techniques et outils de traitement automatique de la langue, de classification et d'apprentissage automatique, tandis que le système est destiné au grand public.

Pour cela nous avons structuré notre travail en quatre chapitres :

Chapitre1 : ce chapitre est consacré à l'exploration des systèmes de questions-réponses.

Chapitre 2 : sera consacré à la présentation d'une manière détaillée des éléments importants de l'apprentissage automatique et de la classification du texte.

Chapitre 3 : Le troisième chapitre est dédié à la description de l'approche proposée, le processus de formation et de fonctionnement de système ainsi que les paramètres des algorithmes utilisés.

Chapitre 4 : Le quatrième chapitre expose les performances des modèles utilisés et leur déploiement dans une application web.

Enfin, notre travail s'achève par une conclusion générale résumant les grands points qui ont été abordés.

Chapitre 1

État de l'art sur les systèmes de question-réponse

Introduction

La quantité de documents électroniques mise à disposition, notamment grâce aux réseaux informatiques, a largement modifié la notion de recherche d'information. Les utilisateurs ont en effet un accès de plus en plus direct à l'information. Cependant, pour accéder plus facilement à une information pertinente, des systèmes de recherche d'information se révèlent incontournables. Bien que les moteurs de recherche constituent une solution efficace pour trouver des documents correspondant à une requête utilisateur, ils s'avèrent moins performants concernant la recherche d'une donnée précise. De ce fait, il est primordial de faire appel à des systèmes plus élaborés capables de retourner une information fiable à un besoin d'information précis. C'est l'ambition des systèmes de question-réponse. Pour cela, il s'avère nécessaire de présenter d'abord ce système et ses approches afin de les comprendre.

1.1 Généralités sur les systèmes de questions-réponses SQR

La question-réponse est conçue comme un type particulier de recherche d'information précise. Elle consiste à trouver des réponses courtes et précises à des questions en langage naturel. Cette discipline est une forme avancée de la recherche d'information. C'est une évolution importante des systèmes de recherche d'informations. La question-réponse est un processus complexe. Il demande d'abord la compréhension d'un besoin d'information exprimé en langue naturelle par une question. En effet, il assure la réduction d'un fardeau de multiples documents qui peuvent être assez fastidieux. Simultanément, cette technologie est conçue pour minimiser le temps de recherche et de navigation et maximiser l'utilité des connaissances scientifiques et de données.

Cette technologie a potentiellement accompagné par une évolution parallèle à celle de la croissance exponentielle et continue de l'information. En particulier, grâce à l'impact crucial de ces informations sur la recherche d'informations et les applications du monde réel, la demande est toujours en croissance pour les systèmes de question-réponse. Ces derniers fournissent aux chercheurs la possibilité de trouver l'information précise.

Plusieurs définitions ont été proposées à ce jour pour un système de question-réponse. Selon Usunier et ses collègues [90], un système de question-réponse est un moyen de génération des réponses exactes et pertinentes à des questions formulées en langage naturel [90]. En outre, ce système est capable de répondre à des questions en cherchant la réponse dans un corpus de textes ou sur des sites Internet [64]. Dans la plupart des définitions, un système de question-réponse est classiquement constitué d'un ensemble de modules. Ces derniers réalisent respectivement une analyse de la question, une recherche de portions de documents pertinents et une extraction de la réponse.[52]

1.2 Qu'est-ce qu'une question ?

Une question est une phrase du langage naturel, qui commence habituellement par un mot interrogatif et exprime un besoin d'information de l'utilisateur. Parfois, une question a une forme de construction impérative et commence par un verbe. Dans un tel cas, la demande d'information est appelée déclaration [71].[52]

La qualité de la réponse dépend souvent de la qualité de la question. Donc la façon de poser des questions est la clé pour obtenir des informations correctes et spécifiques. Comme les systèmes question/réponse sont des systèmes qui répondent à une question posée en langage naturel, par l'extraction d'une réponse précise à partir d'un corpus de documents, vous devez donc poser de bonnes questions qui vous permettent de trouver ce qui vous convient le mieux. Il existe différents types de questions, notamment :

1.2.1 Fermée

Ce type de question mène à une réponse brève où l'on y répond par oui, ou par non. C'est une forme classique pour préciser certaines informations. [44]

Exemple : As-tu atteint ton objectif de la semaine ?

1.2.2 Ouverte ou d'approfondissement

La question ouverte sert à obtenir une information nouvelle. Donc, on utilise un mot interrogatif comme qui (à qui, de qui, avec qui...), que, quoi (à quoi, de quoi, avec quoi...), où, comment, pourquoi, combien, à quelle heure etc. La réponse contient donc une information nouvelle qui n'est pas dans la question.[44]

Exemple : Où partez-vous en vacances ? Pourquoi as-tu fait ça ? Combien de fois s'est-il absenté cette semaine ?

1.2.3 Factuelle

La question factuelle permet de se centrer sur les faits, de décrire une situation en éliminant les perceptions ou les jugements.[44]

Exemple : Peux-tu m'expliquer le fil des événements ? quel est la date de déclenchement de la première guerre mondiale ?

Il existe d'autres questions qui ont des résultats sous forme de liste.

Exemple : citez les pays de l'Union Européen ?

1.3 Qu'est-ce qu'une réponse ?

Avec un système de question-réponse, nous devons apporter à l'utilisateur une réponse à la question qu'il a formulée. Or, cette notion qui semble intuitive pose des problèmes de définition. Tout d'abord, la notion de « réponse » qui peut référer à un très court fragment de texte aussi bien qu'à une longue phrase justificative n'est pas clairement définie dans le langage courant. En conséquence, les personnes peuvent répondre d'une manière différente à une même question.

Classiquement, la quasi-totalité des systèmes de question-réponse possèdent des architectures communes mais cela ne signifie pas qu'ils soient similaires. La différence principale entre ces systèmes réside dans l'approche proposée pour chacun d'eux. De surcroît, cette différence se survient dans les techniques et les outils utilisés par ces systèmes. Ainsi, pour quelle langue, un tel système est mis en place.[52]

1.4 A propos de la question-réponse

La question-réponse constitue un champ d'étude majeur en recherche d'informations précises, plus particulièrement, dans le domaine de génération des réponses à des questions posées en langage naturel. En effet, les notions des systèmes de recherche d'informations et des systèmes de question-réponse sont étroitement liées via des concepts qui sont situés à l'intersection de plusieurs domaines, dont notamment le TALN, la recherche d'informations (RI) et l'interface homme-machine (figure 1.1).

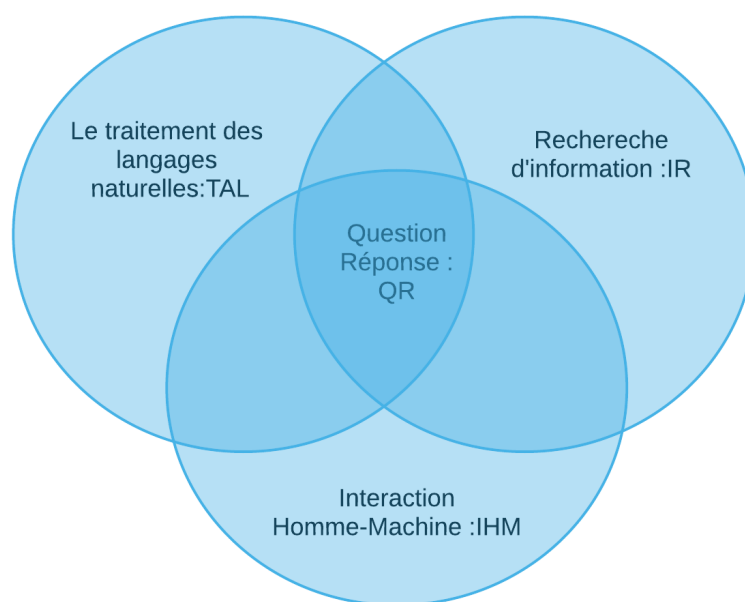


FIGURE 1.1 – Intersection de la question-réponse avec différents domaines de recherche.

De nos jours, les usagers qui demandent des informations précises ont besoin d'une vision synthétique et globale des informations afin de guider et d'adapter leur prise de décision. Pour faciliter ce processus, ils utilisent des systèmes de question-réponse. Ces outils permettent aux demandeurs d'informations d'avoir choisi parmi des masses de documents seulement ceux qui peuvent contenir l'information désirée et exacte ainsi de l'extraire.[52]

1.4.1 Le traitement du langage naturel (TALN)

Les meilleurs systèmes de Question-Réponse sont des systèmes basés sur le traitement du langage naturel (TALN) .

Le Langage Naturel est le moyen de communication employé par les êtres humains de façon inné dans la vie quotidienne. L'anglais, l'arabe et le français sont des exemples de langages naturels. Ils sont construits selon une syntaxe, une grammaire et peuvent contenir beaucoup d'ambiguïtés.

Ils diffèrent ainsi des Langages Formels, utilisés pour transférer des informations à propos desquelles aucune ambiguïté n'est possible. Les mathématiques, les langages informatiques comme le PHP ou le binaire sont des exemples de langages formels.

Les ordinateurs savent très bien interpréter les langages formels, mais un des plus grands challenges de l'informatique est de créer des solutions capables de comprendre le langage naturel. Une discipline de l'intelligence artificielle, le Traitement Automatique du Langage Naturel (TALN ou NLP en anglais), qui se consacre aux interactions entre les machines et l'humain s'attache précisément à cet objectif.

Le traitement naturel du langage, ou Natural Language Processing (NLP) en anglais, est une technologie d'intelligence artificielle visant à permettre aux ordinateurs de comprendre le langage humain.

L'objectif de cette technologie est de permettre aux machines de lire, de déchiffrer, de comprendre et de donner sens au langage humain. D'importants progrès ont été effectués dans ce domaine au fil des dernières années, et le traitement naturel du langage est aujourd'hui exploité pour une large variété de cas d'usage...

C'est pourquoi La NLP est un élément essentiel de toute intelligence artificielle face à l'homme. Un système NLP efficace est capable d'ingérer ce qui lui est dit, de le décomposer, de comprendre sa signification, de déterminer l'action appropriée et de répondre dans un langage que comprendra l'utilisateur.[27]

1.4.1.1 Fonctionnement du traitement du langage naturel TALN

La plupart des techniques de Traitement du langage naturel reposent sur le Deep Learning ou apprentissage profond. Les algorithmes d'intelligence artificielle sont entraînés à partir de données afin d'apprendre analyser le langage humain pour y trouver des patterns et des corrélations.

Les algorithmes ont pour rôle d'identifier et d'extraire les règles du langage naturel, afin de convertir les données de langage non structuré sous une forme que les ordinateurs pourront comprendre.

Au passé, les anciennes approches de Traitement du langage naturel reposaient sur une approche basée sur des règles. Les algorithmes de Machine Learning de l'époque recevaient pour consigne de chercher des mots et des phrases dans un texte et donnaient des réponses spécifiques en fonction. Cependant, le Deep Learning permet une approche plus flexible, plus intuitive, et donc plus proche du langage naturel et de la façon dont les humains l'apprennent pendant l'enfance.

En règle générale, une interaction entre humains et machines via le TALN se déroule de la façon suivante : dans un premier temps, l'humain pose une question à la machine, la machine capture le texte, les données textuelles sont traitées puis à nouveau converties sous forme d'une phrase en langage naturel et enfin la machine répond à l'interlocuteur humain.[27]

1.4.1.2 Les différentes techniques de TALN

Les deux principales techniques utilisées pour le Traitement du langage naturel sont l'analyse syntaxique et l'analyse sémantique. L'analyse syntaxique consiste à identifier les règles grammaticales dans une phrase afin d'en déchiffrer le sens.

Plusieurs techniques d'analyse sémantique existent. Le « parsing » consiste à analyser la grammaire d'une phrase. La segmentation par mot consiste à diviser un texte en unités, tandis que la segmentation morphologique divise les mots en groupes.

L'analyse sémantique quant à elle consiste à déchiffrer directement le sens d'un texte en utilisant des algorithmes pour analyser les mots et la structure des phrases. Les algorithmes peuvent notamment se baser sur le contexte, ou comparer les textes avec des

bases de données pour en comprendre le sens. Cependant, il s'agit d'une approche complexe et aucun algorithme réellement capable de comprendre le sens d'un texte de cette façon n'existe pour l'instant... [27]

1.4.2 Recherche d'information (RI)

La recherche d'information (RI) est le domaine qui étudie la manière de retrouver des informations dans un corpus. Celui-ci est composé de documents d'une ou plusieurs bases de données, qui sont décrits par un contenu ou les métadonnées associées. Les bases de données peuvent être relationnelles ou non structurées, telles mises en réseau par des liens hypertexte comme dans le World Wide Web, l'internet et les intranets. Le contenu des documents peut être du texte, des sons, ou des images.

La recherche d'information est historiquement liée aux sciences de l'information et à la bibliothéconomie qui visent à représenter des documents dans le but d'en récupérer des informations, au moyen de la construction d'index. L'informatique a permis le développement d'outils pour traiter l'information et établir la représentation des documents au moment de leur indexation, ainsi que pour rechercher l'information.[52]

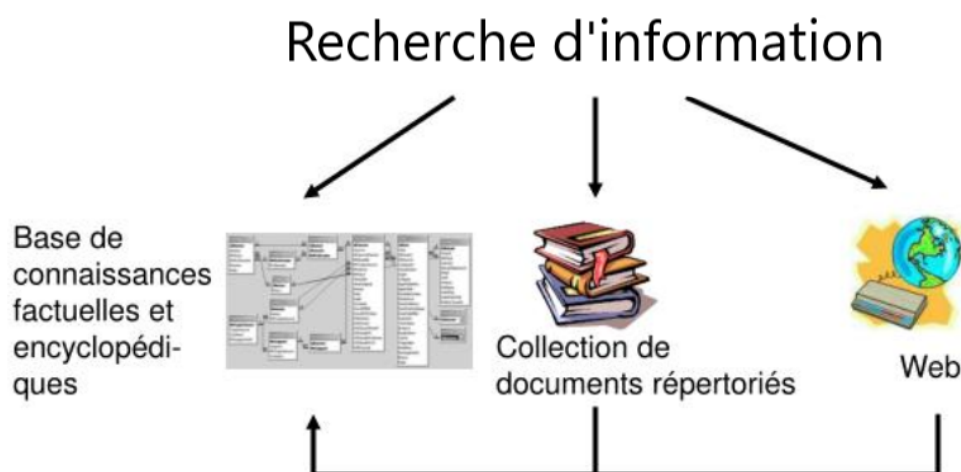


FIGURE 1.2 – Recherche d'information.

1.4.2.1 Base de connaissances factuelles et Encyclopédiques

Les sources factuelles permettent d'accéder directement à des informations ponctuelles telles que des définitions, des statistiques, des données financières, historiques ou géographiques mais aussi à des informations sur une personne, un pays, ou un événement. Elles regroupent entre autres des dictionnaires, des encyclopédies, des annuaires et des répertoires.

Une encyclopédie est un ouvrage de référence composé d'articles de base sur un grand nombre de sujets généraux ou spécifiques. Comme Wikipédia, c'est une encyclopédie gratuite et aujourd'hui c'est un site très puissant et l'une des entreprises les plus pionnières dans le domaine de la connaissance par rapport aux autres encyclopédies.[4]

1.4.2.2 Collection des documents répertoriés

Un répertoire est un outil de collecte de données, initialement d'adresses ou de noms de personnes. C'est un instrument de recherche présentant des informations, quel qu'en soit le support, classées par ordre alphabétique, numérique, chronologique ou systématique pour l'identification, la description ou la localisation de personnes, de documents, d'organismes, de lieux, de ressources Web ou d'objets.

Un répertoire est un fichier spécial contenant les adresses d'autres fichiers. Les répertoires sont souvent représentés par des dossiers dans lesquels les autres fichiers sont inclus. [30]

1.4.2.3 Web

Le World Wide Web a (d'araignée), abrégé www ou le Web, le réseau mondial ou la Toile, est un système hypertexte public fonctionnant sur Internet. Le Web permet de consulter avec un navigateur des pages accessibles sur des sites. L'image de la toile d'araignée vient des hyperliens qui lient les pages web entre elles.

Parmi les services Web, le moteur de recherche qui est une application Web permettant de trouver des ressources à partir d'une requête sous forme de mots. Les ressources peuvent être des pages Web, des articles, des fichiers, etc.[41]

1.5 Les domaines de système de questions-réponse

Le système de réponse aux questions (SQR) est un système de recherche d'informations dans lequel une réponse directe est attendue.

Et pour cela, La recherche en SQR tente de traiter un large éventail de types de questions, notamment : faits, liste, définition, comment, pourquoi, questions hypothétiques, contraintes sémantiquement et multilingues.

Généralement, le système de réponse aux questions peut être classé en système de réponse aux questions du domaine fermé et en système de réponse aux questions d'un autre ouvert.

La réponse aux questions en domaine ouvert traite des questions sur presque tout et ne peut s'appuyer que sur Le corpus de documents pour la recherche des réponses, peut éventuellement être le web avec des formats et des styles très divers, ou alors des grands ensembles de documents électroniques tels que des collections de journaux, d'articles, de dépêches, de textes législatifs, etc.[42][88]

Cette réponse prend la forme de textes courts plutôt qu'une liste de documents jugés pertinents. Dans ce cadre, [63] suggèrent que les systèmes de question-réponse ouverts ont besoin de vastes connaissances pour atteindre une couverture élevée.

La réponse aux questions en domaine fermé traite des questions dans un domaine spécifique et peut être considérée comme une tâche plus facile, par exemple le domaine de fatwa islamique. Alternativement, le domaine fermé peut se référer à une situation où seul un ensemble limité de questions est accepté, telles que des questions demandant des informations descriptives plutôt que procédurales.[42][65]

1.6 Taxonomies des systèmes de question réponse

Dans cette section, nous présentons les différentes classifications effectuées pour les systèmes de question-réponse dans différentes langues. Sur la base de la littérature étudiée, nous identifions huit classifications disponibles pour un grand nombre de systèmes de question-réponse. Par ailleurs, nous présentons plusieurs travaux qui favorisent des classifications de ces systèmes. Dans la suite de cette section, nous citons en détail quelques-unes d'entre elles.[52]

1.6.1 Taxonomie proposée par [78]

Cette première classification étudie la complexité des questions et la difficulté du processus d'extraction des réponses [78]. Ce genre de classification favorise cinq classes de systèmes de question-réponse avec croissance de complexité, y compris :[52]

- Classe 1 : Systèmes répondeurs à des questions factuelles.
- Classe 2 : Systèmes favorisent des processus de raisonnement simples.
- Classe 3 : Systèmes extraient des réponses à partir de multiples sources.
- Classe 4 : Systèmes qui proposent un dialogue interactif avec l'utilisateur.
- Classe 5 : Systèmes capables d'effectuer un raisonnement analogique.

1.6.2 Taxonomie proposée par [51]

Quelques années après, Athenikos et Han proposent une classification des systèmes en se basant sur deux critères [51]. Cette classification nécessite l'appui sur des connaissances sémantiques, elle est composée de trois classes de systèmes :[52]

- SQR sémantiques,
- SQR basés sur les inférences,
- SQR fondés sur des représentations logiques.

Simultanément, dans ce travail les auteurs présentent une autre classification des systèmes de question-réponse en domaine médical en deux classes telles que :

- Systèmes de question-réponse médicaux sémantiques.
- Systèmes de question-réponse médicaux non sémantiques.

1.6.3 Taxonomie proposée par [74]

Une troisième classification proposée par Lopez et ses collègues. En outre, ces auteurs prennent en considération les sources des réponses et les entrées/sorties des systèmes de question-réponse comme un critère de classification [74]. Plus précisément, en s'appuyant sur les ontologies ou sur les ressources de réponses extraites, ces auteurs présentent un travail illustrant un état de l'art sur la catégorisation de ces systèmes. A cet égard, les auteurs favorisent deux types de classification : Le premier est basé sur les ontologies, ce genre de classification engendre l'existence de trois classes de systèmes, à savoir :[52]

- Interfaces en langage naturel pour les bases de données.
- Questions-réponses à partir de documents textuels.
- Questions-réponses avec des données/textes/langages propriétaires.

Le deuxième type définit une classification suivant les sources des réponses. Il favorise trois classes de SQR, y compris :

- Des systèmes dont les sources sont des bases de données structurées.

- Des systèmes dont les sources sont des textes non structurés.
- Des systèmes dont les sources sont des bases de connaissances sémantiques précompilées.

1.6.4 Taxonomie proposée par [81]

Selon une classification présentée par Pho, les systèmes de question-réponse sont basés sur des méthodes de génération des réponses. Ce genre de classification renferme trois types des systèmes de question-réponse : une première classe de systèmes qui sont fondés sur les patrons [91], la deuxième repose sur la reformulation de la réponse [83]. L'intégration de ces deux genres de méthodes présente une troisième classe de systèmes. Dans ce contexte Pho développe un système de génération des énoncés en langage naturel, appelé SYNGE-A (System for Natural Language GEneration of Answers) [81]. Ce système prend en entrée un couple de question-réponse en utilisant des ressources externes, qui lui rend générique, paramétrable et adaptable à tous les systèmes de question-réponse, à savoir : des patrons, des lexiques et des règles grammaticales. L'usage de la troisième catégorie des systèmes favorise la particularité du système SYNGE-A par rapport aux autres systèmes : une génération fondée sur les patrons, ainsi sur les règles grammaticales. En outre, quand la question posée par l'utilisateur a été analysée à l'avance, une étape d'annotation est mise en évidence. La génération des réponses se fonde sur les patrons. Ce qui diffère quand la génération est fondée sur les règles grammaticales dont une analyse syntaxique de la question permet d'obtenir un arbre syntaxique.[52]

1.6.5 Taxonomie proposée par [56]

Les travaux de [55], [56] s'appuient sur l'analyse des questions traitées pour produire deux groupes principaux de systèmes : des systèmes à base des approches surfaciques-syntaxiques et des systèmes à base d'approches profondes sémantiques. La première catégorie de ces systèmes ne force pas l'analyse sémantique des questions traitées. L'extraction d'une réponse parvient par la recherche des passages de textes qui contiennent cette réponse. Dans ce cadre, une étape d'indexation est effectuée. Cette étape note les expressions qui présentent le mieux document dont la méthode utilisée pour assurer cette indexation est de rendre un document en un « sac de mots ». L'indexation s'appuie sur des traitements linguistiques de documents (une analyse morphologique et/ou syntaxique) pour assurer son enrichissement. Dans cette catégorie, l'utilisation des techniques de traitement des langues (TALN) pour l'extraction des réponses ne se plie pas l'analyse sémantique des questions et des documents.

Néanmoins, la deuxième catégorie exige obligatoirement une analyse de la question traitée ainsi que les documents qui peuvent la répondre. Cette catégorie représente formellement le sens de la question. En se basant sur cette catégorie, divers travaux ont été trouvés [80], [82]. Plus précisément, Niu et ses collègues élaborent les enjeux généraux des technologies pour les question-réponse en médecine. En outre, ces auteurs ont utilisé le format PICO (Patient/Problème, Intervention, Comparaison, Outcome) pour identifier des rôles sémantiques dans la question et les textes qui seront utilisés dans l'étape d'extraction des réponses. Alors que, le travail de [82] favorise des représentations logiques des questions et des documents.[52]

1.6.6 Taxonomie proposée par [66]

En se basant sur les méthodes utilisées, les auteurs exposent une autre classification des systèmes de question-réponse en deux grandes catégories. La première regroupe les systèmes qui subissent des méthodes de traitement du langage naturel et de recherche d'informations. Les tâches principales effectuées dans cette catégorie sont le marquage des entités nommées, le traitement de la syntaxe, etc. La deuxième catégorie rassemble les systèmes qui exercent un raisonnement avec le langage naturel. Ces deux taxonomies sont considérées très importantes.[52]

Ces deux catégories proposent quatre classes de systèmes de question-réponse, à savoir :

- Systèmes de question-réponse basés sur le web.
- Systèmes de question-réponse basés sur la recherche d'information / extraction d'information.
- Systèmes de question-réponse en domaine restreint.
- Systèmes de question-réponse basés sur des règles.

1.6.7 Taxonomie proposée par [75]

L'étude de la catégorisation des systèmes en d'autres langues que l'arabe s'achève par l'interrogation de [75]. Dans leur travail, ces auteurs proposent, explicitement, une catégorisation des systèmes de question-réponse en huit taxonomies. D'abord, Mishra et Jain discutent les détails leur classification. Ensuite, ils donnent une description générale pour chaque groupe ainsi que pour ses classes. Enfin, ils discutent les avantages et les inconvénients des systèmes pour chaque classe. Dans ce cadre, les auteurs catégorisent les systèmes sur la base de différents critères, tels que les types de questions traitées, les types de sources de données consultées, les types de traitement effectués sur les questions et les sources de données, les types de modèle de récupération, les formulaires de réponses générées et les caractéristiques des sources de données. Cette classification favorise huit groupes de systèmes. Chaque groupe contient plus qu'une classe. Le tableau 1.1 affiche les groupes des systèmes. Il montre également les critères étudiés pour chaque classification ainsi les classes pour chaque groupe.[52]

Groupe	Critères	Classes
Groupe 1	Domaine d'application	Domaine restreint. Domaine ouvert.
Groupe 2	Types de questions posées	Questions factuelles. Question liste. Questions hypothétiques. Question de confirmation. Question causales.
Groupe 3	Types d'analyses effectuées aux questions	000000 Analyse morphologique. Analyse syntaxique. Analyse sémantique. Analyse pragmatique. Analyse du type de réponse attendu. Reconnaissance de l'objet des questions.

Groupe 4	Types de sources de données	Sources de données structurées. Sources de données semi structurées. Sources de données non structurées. Web sémantique.
Groupe 5	Types de fonctions correspondantes utilisées dans les différents modèles de récupération	Définir les modèles théorique. Le modèle algébrique. Les modèles de probabilité. Les modèles à base de fonctionnalités.
Groupe 6	Caractéristique des sources de données	La taille de la Source. La langue.
Groupe 7	Techniques utilisés	Des techniques de fouille de données. Des techniques de recherche d'information. Des techniques de compréhension du langage naturel. Extraction des connaissances et découverte des techniques.
Groupe 8	Formes de réponses générées par les systèmes	Texte extrait. Extraits ou d'autres multimédias. Réponse générée

TABLE 1.1 – Classification des systèmes de question-réponse proposée par [75]. [52]

1.6.8 Classification des systèmes arabes

Selon les classifications qui ont été préalablement mentionnées, nous pouvons noter qu'elles sont mises en place pour certaines langues telles que l'anglais, le français, le japonais et le chinois, etc. Relativement à la classification présentée dans le tableau I.1, une classification pour les systèmes de question-réponse pour l'arabe a été proposée par [49]. En l'occurrence, ces systèmes sont catégorisés en quatre classes. D'ailleurs, cette classification sert effectuée à la base de plusieurs outils et techniques qui sont extensivement utilisés par chaque système.

- **Systèmes basés sur l'interrogation des bases de données.**

Les systèmes appartenant à cette catégorie introduisent une approche fondée sur le dialogue Homme-Machine. En effet, ces systèmes transforment la question en une requête et interrogent des bases de données afin de sélectionner la réponse. AQAS [77] est considéré parmi les premiers systèmes qui sont apparus dès les années 60, il cherche des réponses à partir des bases de données structurées. En revanche, QARAB [67] cherche des réponses à des questions à partir de documents non structurés extraits à partir du journal Al-RAYA.

- **Systèmes basés sur les techniques de TAL et de RI.**

Cette catégorie illustre les principaux systèmes qui reposent extensivement sur des techniques de TALN et de recherche d'information pour trouver la réponse précise. En effet, la quasi-totalité des systèmes traitent un type particulier de questions, à savoir, la question factuelle et reposent sur des approches morpho-syntaxiques. Par exemple, ArabiQA [57] recourt aux techniques de reconnaissance des entités nommées; QASAL [58] repose sur la plateforme Nooj pour extraire la réponse à partir d'un livre d'éducation. Par contre, AQUASYS [54] permet d'analyser la question et d'extraire la réponse à partir

d'un corpus. En plus, JAWEB [72] a été construit sur la base d'AQUASYS en fournissant une interface utilisateur comme une extension.

- **Systèmes basés sur la compréhension automatique de textes.**

Une troisième catégorisation repose sur la compréhension automatique de textes pour répondre à des questions. L'extraction de meilleures réponses nécessite un certain type d'inférence et un examen de bases de connaissances acquises précédemment [53]. Dans ce cadre, la plupart des systèmes favorisent la compréhension automatique d'un texte et utilisent des prétraitements, tels que la résolution d'anaphores, la coréférence, ou la reconnaissance d'entités nommées. En fait, IDRAAQ [46] participe à la tâche QA4MRE@CLEF. Cette dernière a inclus pour la première fois la langue arabe. D'autre part, ALQASIM proposé par [62] est basé sur la sélection et la validation de la réponse, il répond à questions à choix multiples. Ce système prend en compte la compréhension en lecture des questions.

- **Systèmes basés sur la logique et l'inférence.**

Cette taxonomie des systèmes de question-réponse s'appuie sur le raisonnement logique et l'inférence textuelle afin de trouver la réponse précise à une question en langue naturelle. A notre connaissance, ce type d'approches est peu utilisé jusqu'à présent dans la question-réponse arabe. L'usage de la logique et de l'inférence dans ce domaine a fait l'objet des travaux rares, à savoir [79]. Ce dernier travail procure une représentation sémantique contrainte en utilisant un cadre d'unification explicite fondé sur l'expansion de la requête (des synonymes et des antonymes) et des similitudes sémantiques.[52]

1.7 Architecture générique d'un SQR

Un système de Questions-Réponses peut être schématiquement décrit au travers d'un enchaînement de différents modules correspondant à trois étapes principales : une analyse (à percevoir comme une « compréhension ») de la question, un traitement des documents et enfin une extraction d'une ou de plusieurs réponses. Concernant, l'étape intermédiaire de traitement des documents, elle est très souvent décomposée en une recherche documentaire classique suivie d'une exploitation des documents sélectionnés à l'issue de cette recherche dans le but de localiser des passages susceptibles de contenir des réponses dans la perspective d'une extraction finale. La figure 1 présente cette architecture schématique à trois, et plus généralement quatre composants ainsi que leurs principales interactions. Chaque étape produit en sortie des informations utilisées en entrée de l'étape suivante (voire en entrée de plusieurs de celles situées en aval).[73]

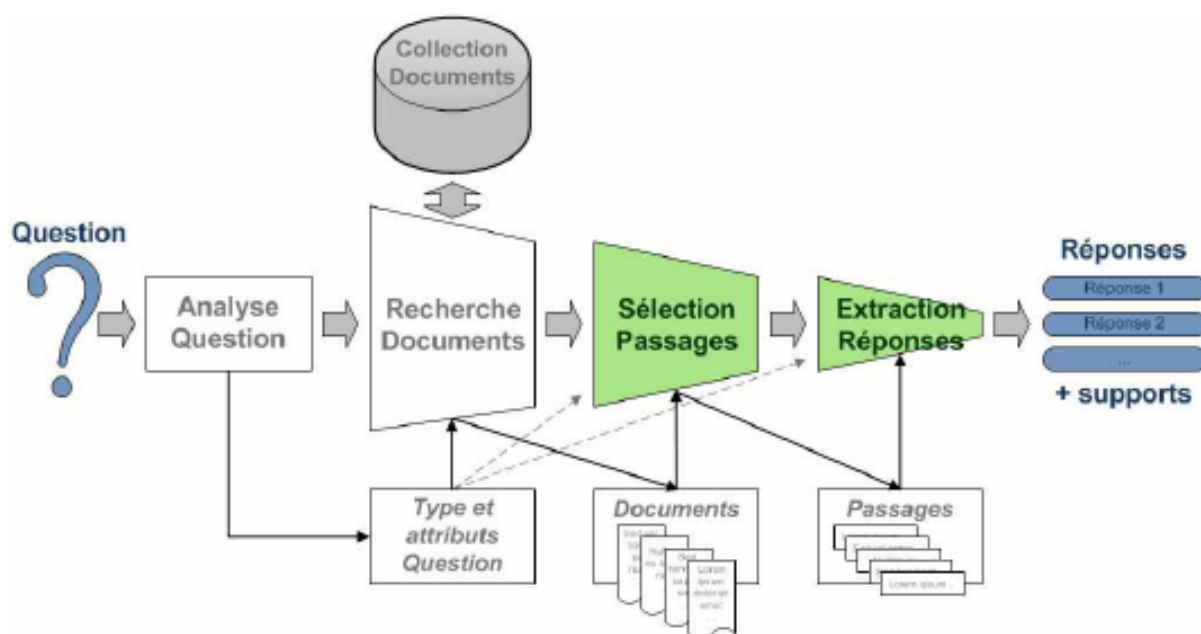


FIGURE 1.3 – Architecture générique d'un système de Questions/Réponses. [73]

1.7.1 Analyse de question

L'étape d'analyse de la question identifie la nature de l'information qui va être recherchée (une étiquette d'entité recherchée ou un type de réponse attendue) et permet de récupérer des informations essentielles pour identifier la réponse correspondante à la question.

Elle construit également une requête à destination d'un système de recherche documentaire. [73]

1.7.2 Traitement des documents

Le système de recherche documentaire propose une liste d'identifiants de documents du corpus contenant tous les mots clés extraits de la requête.

Cette liste pourrait être ordonnée suivant une similarité prenant en compte la fréquence des mots clés.

Le composant en charge du traitement des documents effectue un balisage des entités de type date. Il découpe ces documents en blocs ou passages, effectue une sélection des passages intéressants et élimine par exemple, ceux qui ne contiennent ni une entité date, ni des mots clés.[73]

1.7.3 Extraction des réponses

L'extraction d'une réponse consiste à choisir la « meilleure » réponse à proposer comme résultat, cela en accord avec l'étiquette sémantique associée lors de la première étape. Cette sélection de la ou des meilleures réponses pourrait être faite grâce à l'utilisation de patrons morphosyntaxiques ou grâce à un calcul de proximité des différents termes associés aux mots clés intéressants de la question dans les passages.[73]

1.8 Les approches de système questions-réponses

Les SQR combinent des techniques issues de l'intelligence artificielle, du traitement automatique du langage naturel, de l'analyse statistique, de l'appariement de modèles, de la recherche d'information et de l'extraction d'information. La plupart des travaux récents intègrent une partie ou la totalité de ces approches pour construire des systèmes performants capables de faire face aux faiblesses de ces approches.[47]

Dans la littérature, on distingue trois grandes approches :

1. Approche linguistique.
2. Approche statistique.
3. Approche de filtrage par pattern (motif).
 - Basée sur la surface par pattern.
 - Basée sur les modèles.

1.8.1 Approche linguistique

Un SQR nécessite la compréhension du texte en langage naturel, la linguistique et des connaissances générales. Par conséquent, beaucoup de chercheurs utilisaient des méthodes basées sur l'IA (intelligence artificielle) qui intègrent des techniques de TALN (traitement automatique du langage naturel) et une base de connaissances ou un corpus pour construire des modules pour SQR. Les connaissances sont organisées sous la forme de règles de production, de logiques, de trames, RDF (représenté avec des relations triplets), d'ontologies et de réseaux sémantiques, qui sont utilisés lors de l'analyse de la paire question-réponse. Des techniques linguistiques, telles que la Tokenization, le POS tagging et l'analyse syntaxique, sont implémentées sur les modules de traitement de la question de l'utilisateur pour formuler une requête précise qui extrait simplement la réponse correspondante de la base de données structurée.

Cependant, le déploiement d'une base de connaissances sur un domaine spécifique pose un problème de portabilité, car un domaine d'application différent requiert des règles de grammaire et des règles de mapping différentes. De plus, la construction d'une base de connaissances appropriée est un processus qui prend beaucoup de temps, de sorte que ces systèmes sont généralement appliqués à des problèmes ayant des besoins d'information à long terme pour un domaine particulier.

Certains SQR existants exploitent le Web en tant que ressource de données. Ces systèmes appliquent leurs propres heuristiques pour stocker des informations à partir du Web (données structurées ou non-structurées) dans la base de connaissances locale, à laquelle il faut ensuite accéder et s'appuyer sur des techniques linguistiques et de filtrage pour la génération des réponses à partir des documents pour les données non-structurées ou à partir du résultat d'exécution des requêtes pour les données-structurées.[47]

1.8.2 Approche statistique

Actuellement, la croissance rapide des données Web disponibles (structurées et non-structurées) a accru l'importance des approches statistiques. Ces approches mettent en avant de telles techniques, qui peuvent non seulement traiter la très grande quantité de données mais aussi leur hétérogénéité.

De plus, les approches statistiques peuvent formuler des requêtes en langage naturel. Ces approches nécessitent fondamentalement une quantité suffisante de données pour un

apprentissage statistique précis, mais une fois correctement apprises, elles produisent de meilleurs résultats par rapport à d'autres approches informatiques.

L'algorithme d'apprentissage statistique peut être facilement adapté à un nouveau domaine indépendamment de toute forme de langage. Cependant, l'un des inconvénients majeurs des approches statistiques est qu'elles traitent chaque terme indépendamment et ne parviennent pas à identifier les caractéristiques linguistiques pour la combinaison de mots ou de phrases.

En général, les techniques statistiques ont jusqu'ici été appliquées avec succès aux différentes étapes d'un système QR. Les machines à vecteurs de support ou séparateurs à vaste marge (en anglais support vector machine, SVM), les classificateurs bayésiens et les modèles à entropie maximale, sont des techniques qui ont été utilisées à des fins de classification de questions. Ces mesures statistiques analysent les questions permettant de prédire le type de la réponse attendue par les utilisateurs. Ces modèles sont entraînés sur un corpus de questions ou de documents qui a été annoté avec les catégories mentionnées dans le système.

L'un des travaux pionniers basés sur le modèle statistique était le SQR statistique d'IBM [68]. Ce système utilise le modèle d'entropie maximale pour la classification questions/réponses en fonction de plusieurs caractéristiques de N-gramme ou des ensembles de mots.[47]

1.8.3 Approche de filtrage par pattern

Cette approche utilise la puissance expressive des patterns de texte pour remplacer le traitement sophistiqué dans d'autres approches informatiques. Les patterns sont des formes de langage; la reconnaissance des patterns est étudiée dans de nombreux domaines, notamment la psychologie, la psychiatrie, l'ethnologie, les sciences cognitives et l'informatique [70]. Par exemple, la question « Où se tenait la coupe du monde de cricket 2012? » suit le pattern « Où se tenait <Nom de l'événement> ? » et la réponse suit le pattern « <Nom de l'événement> a eu lieu à <Lieu> ». Actuellement, de nombreux SQR apprennent automatiquement les patterns de texte à partir de passages de texte plutôt que d'utiliser des connaissances ou des outils linguistiques compliqués, tels que l'analyse syntaxique, l'identificateur d'entité nommée NER, l'ontologie, WordNet, etc., pour récupérer des réponses.

La simplicité de tels systèmes les rend plutôt favorables aux petites et moyennes applications, qui ne nécessitent pas des solutions complexes demandant beaucoup de temps et de compétences pour installer et maintenir le système.[47]

1.8.4 La surface par pattern

Cette approche extrait les réponses de la structure linguistique des documents trouvés par le moteur de recherche en s'appuyant sur une longue liste de patterns. La réponse à une question est identifiée sur la base de la similarité entre les patterns ayant une certaine sémantique. Ces patterns sont comme des expressions régulières. Bien que la conception d'un tel ensemble de patterns nécessite beaucoup de compétences humaines et de temps, mais l'approche a montré une grande précision aussi.

Initialement, la méthode basée sur la surface par pattern vise à trouver des réponses à des questions factuelles, car leurs réponses sont limitées à une ou deux phrases. Afin de concevoir un ensemble optimal de patterns, la majorité des SQR basés sur la méthode de

surface par pattern utilisent la méthode décrite par Ravichandran et Hovy [59]. Ils ont mis en place une méthode d'apprentissage automatique qui utilise le bootstrapping pour construire un grand ensemble de patterns commençant seulement avec quelques exemples de paires question/réponse à partir du Web.[47]

1.8.5 Approche basée sur les modèles

Une approche basée sur un modèle utilise des modèles préformates pour les questions. Cette approche vise beaucoup plus l'illustration que l'interprétation des questions et des réponses. L'ensemble des modèles est construit de façon à contenir le nombre optimal des modèles en s'assurant qu'il couvre adéquatement l'espace du problème. Le principe des SQR basés sur les modèles est très similaire au système de réponse automatisé FAQ (Foire aux questions) qui répond avec des réponses préenregistrées à la question de l'utilisateur, contrairement aux FAQ statiques. [47]

1.9 Système de question-réponse Arabe

Dans les sections précédentes, différents systèmes et approches permettent d'obtenir des réponses précises à des questions en langues latines ont été présentés. En effet, depuis son émergence, les systèmes question-réponse a réalisé une performance globale dans ces différentes langues, particulièrement pour l'anglais et quelques autres langues latines qui sont beaucoup plus bénéficiées de l'avancement dans le domaine de la question-réponse. Toutefois, les études qui ont été proposées pour l'arabe ne sont pas aptes de suivre le même rythme d'évolution en raison des défis spécifiques de cette langue. D'ailleurs, la plupart de ces études avaient porté sur des techniques de TALN pour extraire la réponse exacte. Etant donné que, d'après une revue bibliographique, nous remarquons que la quasi-totalité de ces systèmes de question-réponse en arabe reposent sur des approches morphosyntaxiques et que peu d'entre elles fournissent des approches fondées sur la sémantique, l'inférence et la logique. Mais, malgré ce manque, il existe dans la littérature quelques tentatives qui ont obtenu des résultats acceptables dans ce domaine de recherche. La technologie de question-réponse arabe a été étudiée, entre autres, depuis les années 1990 (figure 1.4).

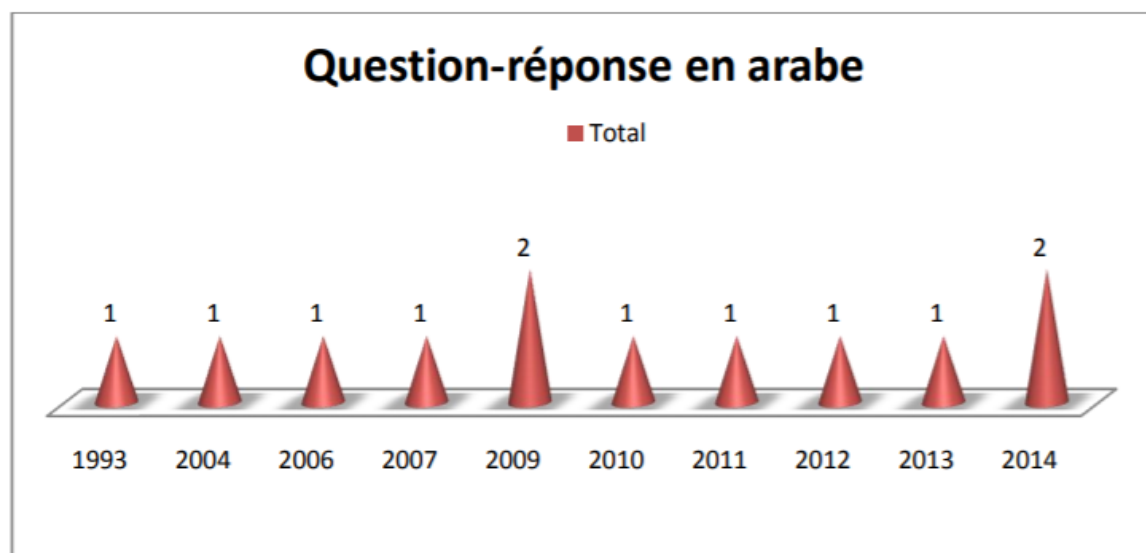


FIGURE 1.4 – Evolution de la question-réponse en arabe depuis son apparition. [52]

La nécessité des systèmes de question-réponse accroît dans le cadre de la langue arabe. Ceci est dû à la pénurie de ces systèmes, qui peut être attribuée aux grands défis qu'ils présentent à la communauté des chercheurs, y compris la spécificité de la langue arabe, tels que les voyelles courtes, absence de lettres majuscules, morphologie complexe, etc. C'est pratiquement à partir de 2004, que quelques tentatives exploratoires ont été réalisées autour des systèmes de question-réponse arabes.[52]

1.9.1 Les défis de la langue arabe

L'importance de la technologie de la question-réponse est très évidente et pertinente en matière de recherche d'informations. Cette technologie a été établie pour plusieurs langues latines telles que l'anglais, le français [48], [54], etc. Néanmoins, les systèmes de question-réponse accomplis en arabe sont encore peu nombreux et immatures en raison des aspects uniques de la langue arabe. Ceci est principalement dû à deux raisons : le manque d'accessibilité aux ressources et aux outils linguistiques (e.g. les corpus et les outils de TALN arabes) et la nature très complexe de la langue elle-même (e.g. l'arabe est flexionnelle et non-concaténâtes). Bien que, l'arabe est dans les dix premières langues dans l'Internet, il manque de nombreux outils et ressources. L'arabe n'a pas les lettres majuscules par rapport aux langues latines, comme dans le cas de l'anglais. Ce problème rend si dur le traitement du langage naturel, comme la reconnaissance des entités nommées. En outre, l'arabe est l'une des langues les moins considérées par les chercheurs dans le champ de la question-réponse [46].

De leur côté, [45] illustrent quelques difficultés de la langue arabe. En effet, ce langage est très flexionnel et dérivationnel ce qui rend son analyse morphologique une tâche très complexe. D'abord, dérivationnel où tous les mots arabes ont trois ou quatre caractères racines. Ensuite, flexionnel où chaque mot se compose d'une racine et zéro ou plusieurs affixes (préfixe, infixes, suffixes). L'arabe se caractérise par des marques diacritiques (voyelles courtes), le même mot avec différents diacritiques peut exprimer des significations différentes. Les diacritiques qui provoquent l'ambiguïté sont généralement omises. Ainsi,

l'absence des lettres majuscules en arabe est un obstacle contre la reconnaissance des entités nommées.

Par conséquent, nous illustrons qu'il existe plusieurs facteurs motivants pour choisir la langue arabe. Par la suite, nous citons quelques exemples de ces facteurs :

- La langue arabe est la sixième langue la plus parlée dans le monde.
- La langue arabe a approximativement 280 millions de parlants natifs et environ 250 millions de parlants non-natifs.
- La langue arabe est considérée également l'une des six langues officielles des Nations Unies (anglais, arabe, chinois, anglais, français, russe et espagnol).
- Croissance des données textuelles arabes sur le web et l'évolution des demandes de logiciels arabe de haute qualité.
- La langue arabe est hautement flexionnelle et dérivationnelle, ce qui rend l'analyse morphologique une tâche très complexe.
- Pas de capitalisation en arabe. Cela complique le processus d'identification des noms propres, acronymes et abréviations.
- Le sens d'écriture est de droite à gauche. En outre, certains caractères changent leurs formes en fonction de leur emplacement dans un mot. [52]

1.9.2 Principaux systèmes proposés

De nos jours, la plupart des systèmes de question-réponse traitent des questions en langues latines (l'anglais, le français, le chinois, le japonais, etc.). Le besoin de développer des systèmes de question-réponse dédiés à l'arabe devient de plus en plus inévitable ces dernières années à cause des difficultés liées à la langue elle-même, aussi par le manque d'outils disponibles pour aider les chercheurs. Par conséquent, les approches proposées signalent qu'il n'y a pas vraiment une méthode standard à adopter lorsqu'il s'agit d'extraire les bonnes réponses à une question. Tout va dépendre du type de problème que nous voulons traiter et du cadre de recherche dans lequel nous nous voulons inscrire. Dans cette section, nous avons présenté quelques systèmes qui sont accomplis dans cette langue depuis leur naissance avec AQAS de [77] jusqu'à Al-Bayan [45] qui est récemment présenté.

La question-réponse arabe a vu le jour avec le système AQAS (question-answering system), présenté par [77]. Ceci est basé sur la connaissance, il extrait des réponses uniquement à partir de données structurées. AQAS accepte une phrase arabe (des états déclaratifs ou une question) et génère la sortie appropriée à l'utilisateur dans le domaine de rayonnement. Le système proposé est considéré comme étant une mise en œuvre en arabe pour le traitement du langage naturel, il est dédié pour le domaine de radiation.

L'étude des questions factuelles était le sujet de plusieurs systèmes de question-réponse, surtout en arabe. De ce fait, de nombreux systèmes tentent de trouver des entités nommées pour répondre aux questions. L'idée est que les questions factuelles se répartissent en plusieurs types distinctifs, tels que « personne », « emplacement », « date », « organisation », etc. La tâche d'identifier ces types est appelée la reconnaissance de l'entité nommée, elle est généralement effectuée par un outil ou dispositif de reconnaissance d'entités nommées. Dans ce cadre, certains systèmes en arabe adoptent une stratégie de recherche appropriée en s'appuyant sur la reconnaissance des entités nommées. Par exemple, AQUASYS (Arabic Question-Answering System) de [54] répond aux questions qui commencent par les pronoms interrogatifs (qui من, quoi ما, où اين, quand متى, combien كم العددية, combien كم الكمية). Effectivement, AQUASYS a été la base d'un autre

travail connexe proposé par [72] pour concevoir et réaliser le système de question-réponse JAWEB (web-based Arabic question answering application system). La particularité de ce système par rapport à AQUASYS est qu'il fournit une interface utilisateur comme une extension. Ce système est axé sur le web pour répondre à des questions commençant par "كم الكمية , كم العددية , متى , اين , ما , من". Un exemple de ces questions est bien illustré dans le tableau 1.2.[52]

N°	Type	Question en arabe	Question en anglais
1	WHO	من هو محمد طنجة؟	Who is Muhammad Tangier ?
2	WHEN	متى توحدت المملكة العربية السعودية؟	When was Kingdom of Saudi Arabia united ?
3	WHAT	ما هي الاهرامات المصرية؟	What are Egyptian pyramids ?
4	WHERE	اين تقع المملكة العربية السعودية؟	Where is the Kingdom of Saudi Arabia located ?
5	HOW MUCH	كم تبلغ درجه حراره القشره الارضيه؟	How much is the temperature of the Earth's crust ?
6	HOW MANY	كم عدد سكان الرياض؟	How many residents are there in Riyadh ?

TABLE 1.2: Exemple de questions répondues par JAWEB. [52]

Nous pouvons évoquer à ce sujet, un autre outil, qui est si important et qui est utilisé par [58] pour développer leur système de question-réponse QASAL (Question- Answering System for Arabic Language), c'est la plateforme NOOJ. Cette dernière est employée afin de trouver des réponses aux questions factuelles à partir d'un ensemble de livres d'éducation. Ainsi, les expérimentations réalisées par Brini et ses collaborateurs ont signalé que pour un ensemble de données de test de 50 questions le système a atteint 67,65% comme précision, 91% comme rappel et 72,85% comme F-mesure.

En réalité, l'un des facteurs clés de la réussite dans le domaine de la question-réponse est l'organisation annuelle de campagnes d'évaluation. Néanmoins, la langue arabe est approximativement absente dans la majorité de ces pistes. C'est pourquoi, les systèmes de question-réponse en langue arabe présentent de nombreux inconvénients en termes de leur processus d'évaluation. Sauf le système ArabiQA (ArabiQA : Arabic QA system) de [57], où leur évaluation a respecté le même pourcentage pour chaque type des entités nommées comme dans CLEF 2006. En fait, ArabiQA est doté d'une architecture générique composée de trois modules : un système de récupération de passages (JIRS), un système de reconnaissance des entités nommées arabes et un module d'extraction de la réponse.

Non seulement le type des questions factuelles à été traité dans la question-réponse arabe. Le deuxième système de question-réponse présenté en arabe, est le système QARAB (Arabic Question Answering System) proposé par [67]. Ce système est basé sur un ensemble de règles pour chaque type de question à l'exception des deux types : "ماذا , كيف" comment et pourquoi). QARAB favorise des réponses courtes à des questions. Il cherche les réponses dans des documents non structurés extraits du journal AlRaya. Il aborde la question comme un "sac de mots".

Mais encore, les questions de définitions sont prises en compte. Considérant qu'une question de définition est une question qui demande des informations importantes à propos de quelqu'un ou de quelque chose. En arabe le premier système qui traite ce type de question est DefArabicQA (Arabic Definition Question Answering System), proposé par [89]. Ce système cherche les définitions candidates en utilisant un ensemble de patrons lexicaux et les catégorise en exploitant des règles heuristiques. De surcroît, DefArabicQA classe les définitions en utilisant une approche statistique.

La succession annuelle des campagnes d'évaluation de la question-réponse telles que TREC et CLEF a permis l'amélioration du développement des tâches plus avancées parmi lesquelles la tâche de la question-réponse pour la compréhension en lecture (QA4MRE).

Cette tâche a été introduite pour la première fois en 2011 pour l'anglais dans la CLEF. L'objectif de cette tâche est de se focaliser sur la compréhension en lecture dans la question-réponse. Néanmoins, l'édition de QA4MRE @ CLEF a été étudiée pour la première fois pour l'arabe en 2012 avec certains systèmes comme IDRAAQ (Information and Data Reasoning for Answering Arabic Questions) de [46]. Ce système est mis en œuvre via une approche à trois niveaux afin d'améliorer la recherche des passages. Également, IDRAAQ couvre deux tâches très importantes : la reconnaissance d'implication textuelle et la validation de la réponse. À noter que, IDRAAQ atteint une précision de 0, 13 et c @ 1 est égal à 0, 21 sans l'utilisation des collections de base de données de CLEF. Ainsi, il repose également sur la densité du modèle de distance N-gramme, l'expansion sémantique et le WordNet arabe.[52]

De même, un autre travail a également porté sur la tâche de QA4MRE@CLEF, ALQASIM dont son abréviation est (Question Answer Selection and Validation system). Ce système est développé par [62], il se focalise sur la sélection et la validation de la réponse et cherche des réponses à choix multiples. ALQASIM a réalisé une performance de 0,31 précision et 0,36 c @ 1 sans utiliser la collection de base de données de CLEF. De surcroît, Ezzeldin et ses associés ont comparé leur système aux trois autres proposés en 2012, un système pour l'anglais et deux autres pour l'arabe.[52]

Dans la question-réponse arabe, les approches fondées sur l'inférence et la logique sont dans leurs premières étapes par rapport aux autres langues comme l'anglais. Au meilleur de notre connaissance, il existe peu de systèmes qui adoptent ce type d'approches. Par exemple, [79] ont proposé un système de question-réponse basé sur la récupération de paragraphes. Il vise à récupérer les paragraphes (de longueur variable) qui contiennent des réponses à la question. Ces auteurs ont utilisé un corpus de 20 documents, et une collection de 100 questions de type oui/non. Celles-ci sont transformées en des représentations logiques. Néanmoins, nous ne trouvons pas d'informations pour la continuité de leur proposition. D'ailleurs, le manque ou l'absence de ce genre d'approches en arabe favorise la pertinence et la faisabilité de l'exploration de nouvelles approches et de nouveaux systèmes que les adoptent. C'est dans ce cadre que nous allons apporter une pierre à la proposition d'une nouvelle approche sémantique et logique pour améliorer la question-réponse arabe. Cette approche se diffère à la majorité des recherches proposées qui sont concentrées sur des aspects morphologiques et syntaxiques.[52]

Une vision moderne de la question-réponse arabe s'intéresse à une compréhension sémantique de documents pour répondre à des questions en langue naturelle. Nous pouvons notamment mentionner le travail de [45] qui a introduit Al-Bayan dont son abréviation est (An Arabic Question Answering System for the Holy Quran), un nouveau système de question-réponse arabe qui est spécialisé pour le Saint Coran. Ce système fournit une compréhension sémantique du Coran pour répondre aux questions des utilisateurs en utilisant les ressources coraniques fiables. Il récupère les versets les plus pertinents et extrait le passage qui contient la réponse du Saint Coran et des livres d'interprétation (Tafsir), AlBayan atteint une précision de 85%.[52]

1.10 Système de question-réponse de Fatwa Islamique

Environ 90% des Arabes sont musulmans. De nombreux musulmans ont besoin de consultations religieuses sur des questions spécifiques. Ces consultations doivent être ac-

quises auprès d'une source légitime telle que des érudits islamiques. La réponse à ces consultations s'appelle une fatwa, qui est considérée comme un point de vue islamique sur de telles questions.[61]

La Fatwa est un domaine à réponse restreinte sur les décisions et opinions juridiques en Islam. Le but est de satisfaire les demandes continues des musulmans d'apprendre, de rappeler ou d'étudier ce qui est légitime dans l'islam.[43]

Dans le domaine de Fatwa, il existe un grand référentiel de réponses archivées sous forme de paires de questions- réponses, ces réponses sont préparées par un expert autorisé « Mufti » ou une organisation officielle Fatwa. FQAS (Fatwa Question Answering Systems) utilisent cette source d'information fiable pour répondre à des nouvelles questions mais similaires posées en langage naturel. Cependant, il existe certains défis auxquels sont confrontés les systèmes de FQA, notamment :[43]

- Les réponses doivent être à un degré élevé de précision dans le domaine religieux.
- L'écart sémantique entre la nouvelle question fatwa qui se posait dans un arabe standard local ou moderne (MSA), et les réponses fatwa bien formées en MSA et en arabe classique (CA).
- Le manque d'ontologie en langue arabe pour les formes arabes MSA et CA.

Plusieurs efforts de recherche ont été mis en œuvre pour fournir des systèmes de réponse aux questions pour le domaine islamique. Mis en œuvre par différents paradigmes de réponse aux questions pour surmonter les défis ci-dessus, y compris les paradigmes basés sur les connaissances, les IR et les ontologies.

L'approche basée sur l'IR implique deux méthodes pour répondre aux questions de langage naturel; la première méthode récupère les réponses les plus similaires à partir d'un corpus de documents contenant des formes répétées de réponses. Cette méthode est largement utilisée en FQA; cependant, il fonctionne bien pour les questions de type factoiide. La deuxième méthode adopte des techniques de résumé ou d'abstraction pour extraire automatiquement une réponse du corpus de documents, cette approche fonctionne bien pour le type complexe de questions, cependant, elle n'a pas été utilisée dans les systèmes FQA, car une génération automatique de réponses n'est pas autorisée dans le domaine de la Fatwa en raison de la sensibilité de la religion.

Plusieurs systèmes de réponses aux questions ont été proposés pour la fatwa islamique. Cependant, ces systèmes utilisent une ontologie en domaine ouvert telle que l'arabe WordNet. Le problème d'une telle ontologie réside dans la variété des définitions de concepts qui sont habituellement généraux (c'est-à-dire qui ne sont pas liés au domaine islamique).[86]

Il n'y a eu qu'une seule étude qui a proposé une ontologie spécifique au domaine, l'ontologie a été construite en utilisant une collection de fatwas, qui a été collectée auprès d'Ibn Uthaymeen-Prayer Fatwas. Plusieurs tâches de prétraitement ont été appliquées afin d'éliminer les données non pertinentes (par exemple, les chiffres, les lettres non arabes et la ponctuation). De plus, le terme fréquence-fréquence inverse du document (TFIDF) a été utilisé afin de fournir les concepts principaux du domaine pour la construction de l'ontologie.[61]

Conclusion

Dans ce chapitre, nous avons introduit la notion de système de questions -réponses, ses différents types, ses domaines d'application ainsi que ses différentes approches. Afin

de comprendre le fonctionnement d'un SQR, nous avons décrit son architecture comme nous avons présenté quelques SQR Arabes et les SQR de Fatwas Islamiques.

Chapitre 2

Classification de textes et Apprentissage profond

Introduction

La classification de texte est fortement basée sur des techniques d'apprentissage profond avec l'évolution de divers réseaux de neurones profonds, pour cela, il est important de bien comprendre ce qu'est l'apprentissage profond et la classification de texte.

Ce chapitre sera consacré à l'apprentissage profond, les concepts de base de RNA ainsi que toutes les notions liées à la classification de texte.

2.1 Apprentissage profond

L'apprentissage profond est une forme d'intelligence artificielle, dérivée d'apprentissage automatique (Machine Learning). Pour comprendre ce qu'est l'apprentissage profond, il convient donc de comprendre ce qu'est l'apprentissage automatique.[11]

2.1.1 Apprentissage machine

L'apprentissage machine est une tentative de comprendre et reproduire la faculté de l'apprentissage humain dans des systèmes artificiels. Il s'agit de concevoir des algorithmes capables, à partir d'un nombre important d'exemples, d'en assimiler la nature afin de pouvoir appliquer ce qu'ils ont ainsi appris aux cas futurs. Ainsi, le but essentiel de l'apprentissage machine est de déterminer la relation entre les objets et leurs catégories pour la prédiction et la découverte des connaissances [87]. [11] [10]

2.1.2 Apprentissage profond

La notion d'apprentissage profond est tout d'abord une traduction directe du terme anglais « deep Learning », que certain préfère traduire par la notion d'apprentissage statistique. De même que sa traduction, sa définition varie également, mais principalement au niveau des détails. Pour définir cette notion dans les grandes lignes, on pourrait dire que :

L'apprentissage profond est un algorithme d'abstraction de haut niveau qui permet de modéliser les données à partir de grands ensembles de données apprises.

Précisons quelques termes :

- **L'abstraction** suppose que les données initiales diffèrent largement des données de sorties, avec pour résultat possible la classification d'images, la prédiction d'un comportement ou une traduction. L'abstraction signifie qu'il n'y a pas de relation simple entre l'entrée et la sortie.

- **La modélisation** signifie que nous tentons de créer un certain scénario réaliste de sorte qu'une classification ou un résultat réaliste en découle.

- **La notion relative aux grands ensembles de données apprises** que les données d'entrée sont extrêmement diverses. L'apprentissage profond ou l'apprentissage automatique implique généralement que les propriétés importantes de ces données sont détectées lors du processus d'apprentissage.

On distingue ainsi trois types d'apprentissage : l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage semi-supervisé.[2] [3]

2.1.3 Apprentissage supervisé (Classification)

L'apprentissage supervisé (ou classification) consiste à construire un modèle basé sur un jeu d'apprentissage et des labels (nom des catégories ou des classes) et à l'utiliser pour classer des données nouvelles (Silva et Ribeiro, 2009 ; Joachims, 2002). Cette technique est utilisée dans plusieurs applications telles que les diagnostics médicaux, la prédiction des pannes et la détection des opinions trompeuses dans les réseaux sociaux.

Il existe plusieurs algorithmes et techniques utilisés pour la classification supervisée telles que :

- **Classification Bayésienne** : C'est une méthode de classification statistique qui se base principalement sur le théorème de Bayes. Elle est utilisée dans plusieurs applications telles que les applications de détection de pourriels (ou Spams) pour séparer les bons courriels des mauvais.

- **Machine à vecteurs de support (SVM)** : Il s'agit d'un ensemble de techniques destinées à résoudre des problèmes de discrimination (prédiction d'appartenance à des groupes prédéfinis) et de régression (analyse de la relation d'une variable par rapport à d'autres) [87].

- **Réseau neuronal** : c'est une technique de type induction c'est-à-dire que, par le biais d'observations limitées, elle essaye de tirer des généralisations plausibles. Elle est basée sur l'expérience qui se constitue une mémoire lors de la phase d'apprentissage (qui peut être aussi non supervisée) appelée entraînement [87].

- **Forêts d'arbres décisionnels (Random Forest)** : C'est une application de graphe en arbres de décision permettant ainsi la modélisation de chaque résultat sur une branche en fonction des choix précédents. On prend ensuite la meilleure décision en fonction des résultats qui suivront. On peut considérer ceci comme une forme d'anticipation.

- **Le Boosting** : Il s'agit d'une méthode de classification émettant des hypothèses qui sont au départ de moindre importance. Plus une hypothèse est vérifiée, plus son indice de confiance augmente. Ce qui prend de l'importance dans la classification [87]. [2]

2.1.4 Apprentissage non supervisé

L'apprentissage non supervisé (en anglais. Clustering) vise à construire des groupes (clusters) d'objets similaires à partir d'un ensemble hétérogène d'objets [87]. Chaque cluster issu de ce processus doit vérifier les deux propriétés suivantes :

- La cohésion interne (les objets appartenant à ce cluster sont les plus similaires possibles).

- L'isolation externe (les objets appartenant aux autres clusters sont les plus distincts possibles).

Le processus de « clustering » repose sur une mesure précise de la similarité des objets qu'on veut regrouper. Cette mesure est appelée distance ou métrique. Le « clustering » est utilisé dans plusieurs applications telles que le traitement d'images, les études démographiques, la recherche génétique, le forage des données et l'analyse des opinions. On distingue plusieurs algorithmes de clustering, exemple :

- **K-moyennes (KMeans)** : Un algorithme de partitionnement des données en K groupes ou clusters. Chaque objet sera associé à un seul cluster. Le K est fixé par l'utilisateur.

- **Fuzzy KMeans** : Il s'agit d'une variante du précédent algorithme proposant qu'un objet ne soit pas associé qu'à un seul groupe.

- **Espérance-Maximisation (EM)** : Cet algorithme utilise des probabilités pour décrire qu'un objet appartient à un groupe. Le centre du groupe est ensuite recalculé par rapport à la moyenne des probabilités de chaque objet du groupe.

- **Regroupement hiérarchique** : deux sous-algorithmes en découlent : le « bottom · up » qui a pour fonction d'agglomérer des groupes similaires, donc en réduire le nombre (les rendre plus lisibles) et d'en proposer un ordre hiérarchique et le « top down » qui fait le raisonnement inverse en divisant le premier groupe récursivement en sous-ensembles. [2]

2.2 Les réseaux de neurones

Un réseau de neurones est un assemblage de constituants élémentaires interconnectés (appelés « neurones » en hommage à leur modèle biologique), qui réalisent chacun un traitement simple mais dont l'ensemble en interaction fait émerger des propriétés globales complexes. Chaque neurone fonctionne indépendamment des autres de telle sorte que l'ensemble forme un système massivement parallèle. Un réseau de neurone ne se programme pas, il est entraîné grâce à un mécanisme d'apprentissage. Les réseaux de neurones artificiels consistent en des modèles plus ou moins inspirés du fonctionnement cérébral de l'être humain en se basant principalement sur le concept de neurone. [60]

2.2.1 Réseaux de Neurones Biologique

Le cerveau humain contient environ 100 milliards de neurones. Ces neurones vous permettent, entre autres, de lire ce texte tout en maintenant une respiration régulière permettant d'oxygéner votre sang, en actionnant votre cœur qui assure une circulation efficace de ce sang pour nourrir vos cellules, etc.

Un neurone est une cellule particulière comme la montre la figure 2.1. Elle possède des extensions par lesquelles elle peut distribuer des signaux (axones) ou en recevoir (dendrites).

Dans le cerveau, les neurones sont reliés entre eux par l'intermédiaire des axones et des dendrites. On peut considérer que ces sortes de filaments sont conductrices d'électricité et peuvent ainsi véhiculer des messages depuis un neurone vers un autre. Les dendrites représentent les entrées du neurone et son axone sa sortie et la synapse qui transmet les signaux entre un axone et une dendrite.[60]

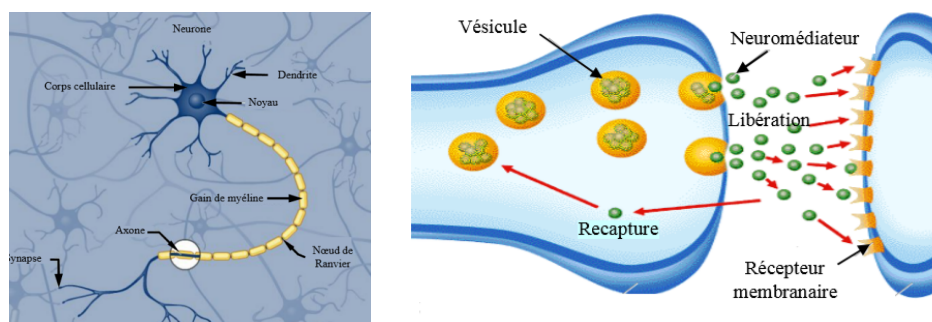


FIGURE 2.1 – Réseaux de Neurones Biologique. [60]

2.2.2 Les réseaux de neurones artificiels

Un réseau de neurone Artificiel peut être considéré comme une boîte noire, qui reçoit des signaux d'entrée et produit des signaux de sortie c'est un modèle mathématique composé d'un grand nombre d'éléments de calculs organisée sous forme de couches interconnectées.

D'autre définition sont donnés comme suite :

Les réseaux de neurones artificiels sont des réseaux fortement connectés de processeurs élémentaires fonctionnant en parallèle. Chaque processeur calcule une sortie unique sur la base des informations qu'il reçoit. Toute structure hiérarchique de réseaux est évidemment un réseau. [60]

2.2.3 Les neurones formels

Un "neurone formel" (ou simplement "neurone") est une fonction algébrique non linéaire et bornée, dont la valeur dépend des paramètres appelés coefficients ou poids. Les variables de cette fonction sont habituellement appelées "entrées" du neurone, et la valeur de la fonction est appelée sa "sortie".

Un neurone est donc avant tout un opérateur mathématique, dont on peut calculer la valeur numérique par quelques lignes de logiciel. On a pris l'habitude de représenter graphiquement un neurone comme indiqué sur la figure 2.2.

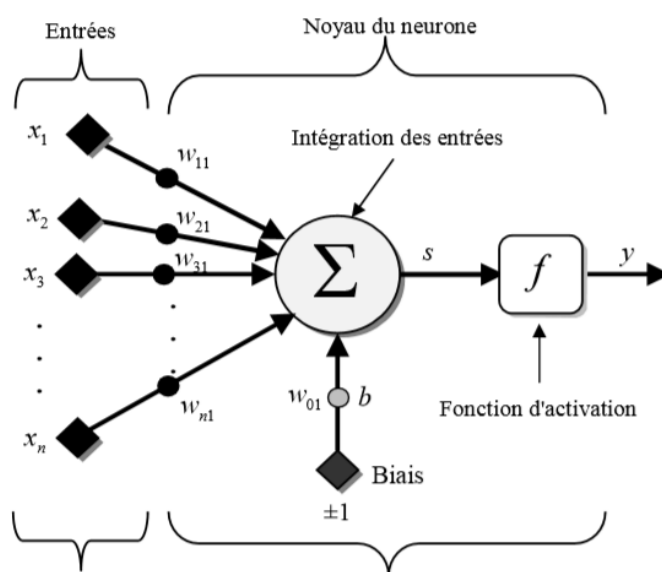


FIGURE 2.2 – Modèle d'un neurone artificiel. [60]

Des observations de neurone biologique, découle le modèle du neurone formel proposé par W. M. Culloch et W. Pitts en 1943 :

- Les x_i représentent les vecteurs d'entrées, elles proviennent soit des sorties d'autres neurones, soit de stimuli sensoriels (capteur visuel, sonore...);
- Les w_{ij} sont les poids synaptiques du neurone j . Ils correspondent à l'efficacité synaptique dans les neurones biologiques ($0 < w_{ij}$: synapse excitatrice; $0 > w_{ij}$: synapse inhibitrice). Ces poids pondèrent les entrées et peuvent être modifiés par apprentissage ;

- Biais : entrée prend souvent les valeurs -1 ou +1 qui permet d'ajouter de la flexibilité au réseau en permettant de varier le seuil de déclenchement du neurone par l'ajustement des poids et du biais lors de l'apprentissage ;
- Noyau : intègre toutes les entrées et le biais et calcul la sortie du neurone selon une fonction d'activation qui est souvent non linéaire pour donner une plus grande flexibilité d'apprentissage. [60]

2.2.4 Modélisation d'un neurone formel

La modélisation consiste à mettre en œuvre un système de réseau de neurones sous un aspect non pas biologique mais artificiel, cela suppose que d'après le principe biologique on aura une correspondance pour chaque élément composant le neurone biologique, donc une modélisation pour chacun d'entre eux. On pourra résumer cette modélisation par le tableau 2.1, qui nous permettra de voir clairement la transition entre le neurone biologique et le neurone formel.

Neurone biologique	Neurone formel
Synapses	Poids des connexions
Axones	Signal de sortie
Dendrites	Signal d'entrée
Noyau ou Somma	Fonction d'activation

TABLE 2.1: Analogie entre le neurone biologique et le neurone formel. [60]

Le modèle mathématique d'un neurone artificiel est illustré dans la figure 2-2. Un neurone est essentiellement constitué d'un intégrateur qui effectue la somme pondérée de ses entrées. Le résultat s de cette somme est ensuite transformé par une fonction de transfert f qui produit la sortie y du neurone. En suivant les notations présentées à la section précédente, les n entrées du neurone correspondent au vecteur $x = [x_1, x_2, x_3, \dots, x_n]^T$, alors que $w = [w_1, w_2, w_3, \dots, w_n]^T$ représente le vecteur des poids du neurone. La sortie s de l'intégrateur est donnée par l'équation suivante :

$$s = \sum_{i=1}^n w_{ij} x_i \pm b \quad (2.1)$$

$$= w_{11}x_1 + w_{21}x_2 + w_{31}x_3 + \dots + w_{n1}x_n \pm b$$

Que l'on peut aussi écrire sous forme matricielle :

$$s = w^t x \pm b \quad (2.2)$$

Cette sortie correspond à une somme pondérée des poids et des entrées plus ce qu'on nomme le biais b du neurone. Le résultat s de la somme pondérée s'appelle le niveau d'activation du neurone. Le biais b s'appelle aussi le seuil d'activation du neurone. Lorsque le niveau d'activation atteint ou dépasse le seuil b , alors l'argument de f devient positif (ou nul). Sinon, il est négatif

Un autre facteur limitatif dans le modèle que nous nous sommes donnés concerne son caractère discret. Nous allons supposer que tous les neurones sont synchrones, c'est à dire qu'à chaque temps tt , ils vont simultanément calculer leur somme pondérée et produire une sortie :

$$y(t) = f(s(t)) \quad (2.3)$$

Dans les réseaux neurones biologiques, tous les neurones sont en fait asynchrones.

Revenons donc à notre modèle artificiel tel que formulé par l'équation (2) et ajoutons la fonction d'activation f pour obtenir la sortie du neurone :

$$y = f(s) = f(w^T x \pm b) \quad (2.4)$$

En remplaçant w^T par une matrice $W = w^T$ d'une seule ligne, on obtient une forme générale que nous adopterons tout au long de ce chapitre :

$$y = f(Wx \pm b) \quad (2.5)$$

L'équation (5) nous amène à introduire un schéma de notre modèle plus compact que celui de la figure 2.2. La figure 2-3 illustre celui-ci. On y représente les n entrées comme un rectangle noir. De ce rectangle sort le vecteur x dont la dimension matricielle est $n \times 1$. Ce vecteur est multiplié par une matrice W qui contient les poids (synaptiques) du neurone. Dans le cas d'un neurone simple, cette matrice possède la dimension $1 \times n$. Le résultat de la multiplication correspond au niveau d'activation qui est ensuite comparé au seuil b (un scalaire) par soustraction. Finalement, la sortie du neurone est calculée par la fonction d'activation f . La sortie d'un neurone est toujours un scalaire. [60]

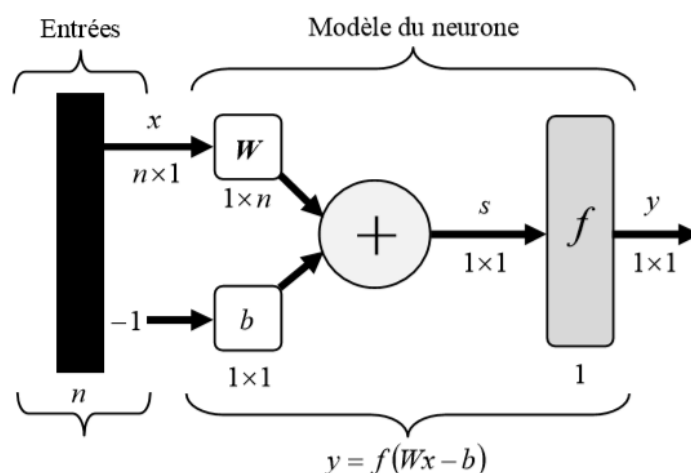



FIGURE 2.3 – Représentation matricielle du modèle d'un neurone artificiel. [60]

2.2.5 Fonctions d'activations

Jusqu'à présent, nous n'avons pas spécifié la nature de la fonction d'activation de notre modèle. Il se trouve que plusieurs possibilités existent. Différentes fonctions de transfert pouvant être utilisées comme fonction d'activation du neurone sont énumérées au tableau 2.2.

Nom de la fonction	Relation entrée/sortie	Icône	Nom Matlab
Seuil	$y = 0$ si $s < 0$ $y = 1$ si $s \geq 0$		hardlim









Seuil symétrique	$y = -1$ si $s < 0$ $y = 1$ si $s \geq 0$		hardlims
Linéaire	$y = s$		purelin
Linéaire saturée	$y = 0$ si $s \leq 0$ $y = s$ si $0 \leq s \leq 1$ $y = 1$ si $s > 1$		saltin
Linéaire saturée symétrique	$y = -1$ si $s < -1$ $y = s$ si $-1 \leq s \leq 1$ $y = 1$ si $s > 1$		saltins
Linéaire positive	$y = 0$ si $s \leq 0$ $y = s$ si $s \geq 0$		poslin
Sigmoïde	$y = \frac{1}{1+e^{-s}}$		logsig
Tangente hyperbolique	$y = \frac{e^s - e^{-s}}{e^s + e^{-s}}$		tansig
Compétitive	$y = 1$ si $s = \text{maximum}$ $y = 0$ autrement		compet

TABLE 2.2 – Différentes fonctions d'activations utilisées dans les RNA. [60]

Les fonctions d'activations les plus utilisées sont les fonctions « seuil » (en anglais « hard limit »), « linéaire » et « sigmoïde ».

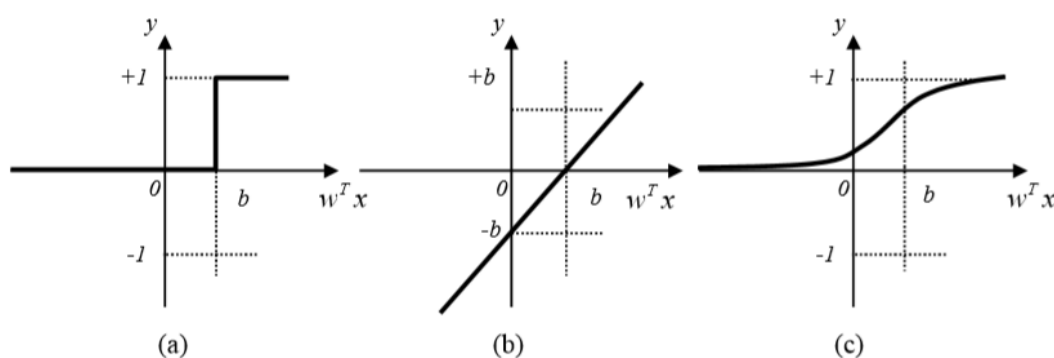


FIGURE 2.4 – Fonctions d'activations : (a) du neurone « seuil » ; (b) du neurone « linéaire », et (c) du neurone « sigmoïde ». [60]

2.2.5.1 Fonction seuil

Comme son nom l'indique, la fonction seuil applique un seuil sur son entrée. Plus précisément, une entrée négative ne passe pas le seuil, la fonction retourne alors la valeur 0 (on peut interpréter ce 0 comme signifiant faux), alors qu'une entrée positive ou nulle ne dépasse le seuil, et la fonction retourne à 1 (vrai). Utilisée dans le contexte d'un neurone, cette fonction est illustrée à la figure 4.a. On remarque alors que le biais b dans l'expression de $y = \text{hardlim}(w^T x - b)$ (équation 4) détermine l'emplacement du seuil sur l'axe $w^T x$, où la fonction passe de 0 à 1. Nous verrons plus loin que cette fonction permet de prendre des décisions binaires.

2.2.5.2 La fonction linéaire

La fonction linéaire est très simple, elle affecte directement son entrée à sa sortie :

$$y = s \quad (2.6)$$

Appliquée dans le contexte d'un neurone, cette fonction est illustrée à la figure 4.b. Dans ce cas, la sortie du neurone correspond à son niveau d'activation dont le passage à zéro se produit lorsque $w^T x = b$

2.2.5.3 La fonction de transfert

La fonction de transfert sigmoïde est quant à elle illustrée à la figure 4.c. Son équation est donnée par :

$$y = \frac{1}{1 + \exp^{-s}} \quad (2.7)$$

Elle ressemble soit à la fonction seuil, soit à la fonction linéaire, selon que l'on est loin ou près de b , respectivement. La fonction seuil est très non linéaire car il y a une discontinuité lorsque $w^T x = b$. De son côté, la fonction linéaire est tout à fait linéaire. Elle ne comporte aucun changement de pente. La sigmoïde est un compromis intéressant entre les deux précédentes. Notons finalement, que la fonction « tangente hyperbolique (tanh) » est une version symétrique de la sigmoïde. [60]

2.2.6 Architecture des réseaux de neurones

L'architecture d'un réseau de neurones est l'organisation des neurones entre eux au sein d'un même réseau. Autrement dit, il s'agit de la façon dont ils sont ordonnés et connectés. La majorité des réseaux de neurones utilise le même type de neurones. Quelques architectures plus rares se basent sur des neurones dédiés. L'architecture d'un réseau de neurones dépend de la tâche à apprendre.

Un réseau de neurone est en général composé de plusieurs couches de neurones, des entrées jusqu'aux sorties. On distingue deux grands types d'architectures de réseaux de neurones : les réseaux de neurones non bouclés et les réseaux de neurones bouclés. [60]

2.2.6.1 Les réseaux de neurones non bouclés

Un réseau de neurones non bouclé réalise une (ou plusieurs) fonctions algébriques de ses entrées, par composition des fonctions réalisées par chacun de ses neurones. Un

réseau de neurones non bouclé est représenté graphiquement par un ensemble de neurones "connectés" entre eux, l'information circulant des entrées vers les sorties sans "retour en arrière" ; si l'on représente le réseau comme un graphe dont les nœuds sont les neurones et les arêtes les "connexions" entre ceux-ci, le graphe d'un réseau non bouclé est acyclique. Le terme de "connexions" est une métaphore : dans la très grande majorité des applications, les réseaux de neurones sont des formules algébriques dont les valeurs numériques sont calculées par des programmes d'ordinateurs. [60]

2.2.6.1.1 Réseaux de neurones monocouches La structure d'un réseau monocouche est telle que des neurones organisés en entrée soient entièrement connectés à d'autres neurones organisés en sortie par une couche modifiable de poids (figure 2.5).

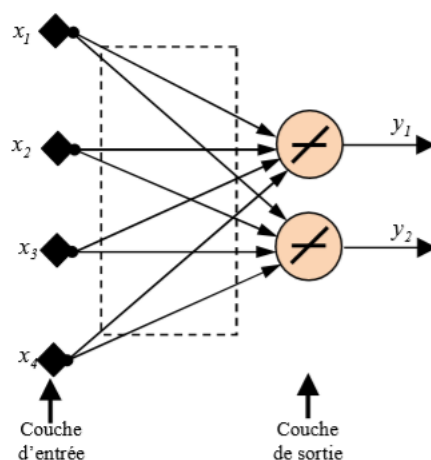


FIGURE 2.5 – Schéma d'un réseau de neurones monocouche. [60]

2.2.6.1.2 Réseaux de neurones multicouches Les neurones sont arrangés par couche. Il n'y a pas de connexion entre neurones d'une même couche, et les connexions ne se font qu'avec les neurones de couches avales. Habituellement, chaque neurone d'une couche est connecté à tous les neurones de la couche suivante et celle-ci seulement. Ceci nous permet d'introduire la notion de sens de parcours de l'information (de l'activation) au sein d'un réseau et donc définir les concepts de neurone d'entrée, neurone de sortie. Par extension, on appelle couche d'entrée l'ensemble des neurones d'entrée, couche de sortie l'ensemble des neurones de sortie. Les couches intermédiaires n'ayant aucun contact avec l'extérieur sont appelées couches cachées.

La figure 2.6 représente un réseau de neurones non bouclé qui a une structure particulière, très fréquemment utilisée : il comprend des entrées, deux couches de neurones cachés et des neurones de sortie. Les neurones de la couche cachée ne sont pas connectés entre eux. Cette structure est appelée Perceptron multicouches.

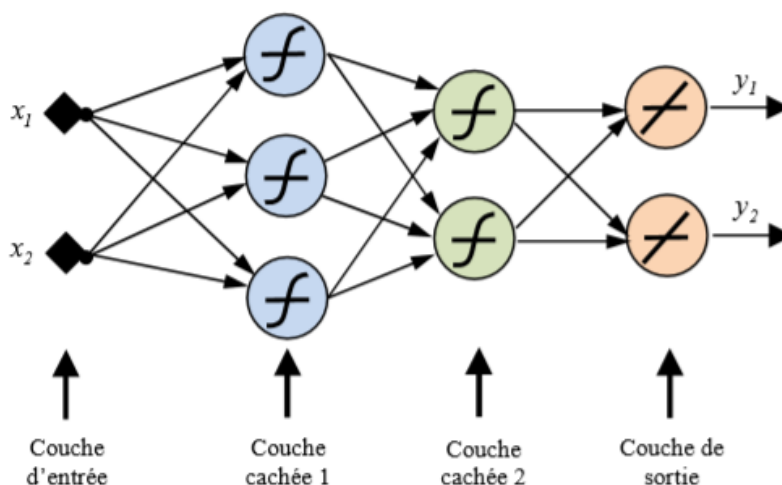


FIGURE 2.6 – Schéma d'un réseau de neurones non bouclé (Perceptron multicouches). [60]

On note aussi que Les réseaux multicouches sont beaucoup plus puissants que les réseaux simples à une seule couche.

2.2.6.1.3 Réseaux de neurones à connexions locales Il s'agit d'une structure multicouche, mais qui à l'image de la rétine conserve une certaine topologie. Chaque neurone entretient des relations avec un nombre réduit et localisé de neurones de la couche avale. Les connexions sont donc moins nombreuses que dans le cas d'un réseau multicouche classique (figure 2.7).

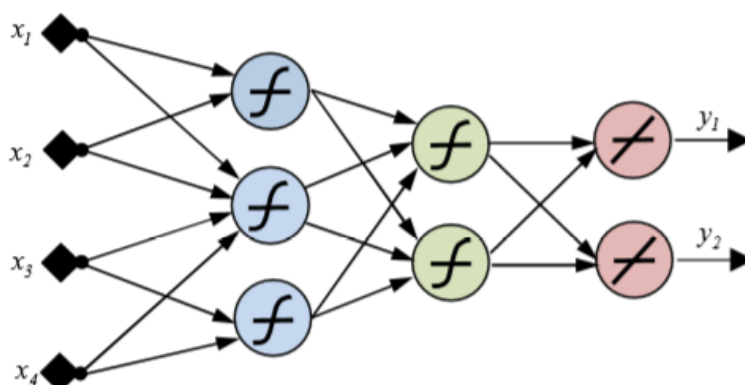


FIGURE 2.7 – Schéma d'un réseau de neurones à connexions locales. [60]

Les réseaux de neurones non bouclés sont des objets statiques : si les entrées sont indépendantes du temps, les sorties le sont également. Ils sont utilisés principalement pour effectuer des tâches d'approximation de fonction non linéaire, de classification ou de modélisation de processus statiques non linéaire. [60]

2.2.6.2 Les réseaux de neurones bouclés

Contrairement aux réseaux de neurones non bouclés dont le graphe de connexions est acyclique, les réseaux de neurones bouclés peuvent avoir une topologie de connexions quelconque, comprenant notamment des boucles qui ramènent aux entrées la valeur d'une ou plusieurs sorties. Pour qu'un tel système soit causal, il faut évidemment qu'à toute boucle soit associé un retard : un réseau de neurones bouclé est donc un système dynamique, régi par des équations différentielles ; comme l'immense majorité des applications sont réalisées par des programmes d'ordinateurs, on se place dans le cadre des systèmes à temps discret, où les équations différentielles sont remplacées par des équations aux différences. Il s'agit donc de réseaux de neurones avec retour en arrière (feedback network or recurrent network), (Figure 2.8).

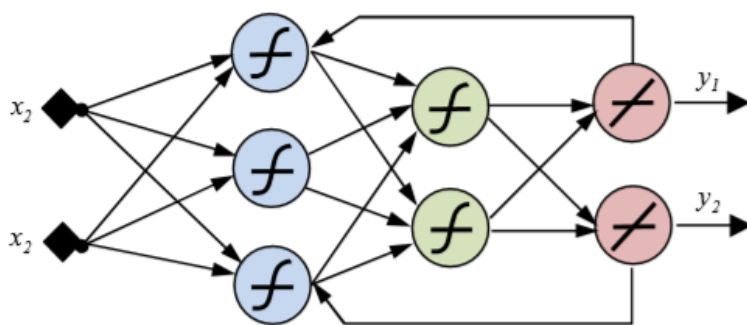


FIGURE 2.8 – Schéma de réseau de neurones bouclé. [60]

Un réseau de neurones bouclé à temps discret est donc régi par une (ou plusieurs) équations aux différences non linéaires, résultant de la composition des fonctions réalisées par chacun des neurones et des retards associés à chacune des connexions.

La forme la plus générale des équations régissant un réseau de neurones bouclé est appelée forme canonique ;

$$x(k+1) = \varphi[x(k), u(k)] \quad (2.8)$$

$$y(k) = \psi[x(k), u(k)] \quad (2.9)$$

Où φ et ψ sont des fonctions non linéaires réalisées par un réseau de neurones non bouclé (mais pas obligatoirement, un perceptron multicouche), et où K désigne le temps (discret).

Les réseaux de neurones bouclés sont utilisés pour effectuer des tâches de modélisation de systèmes dynamiques, de commande de processus, ou de filtrage. [60]

2.2.7 Modèles des réseaux de neurones

2.2.7.1 Modèle de Hopfield

Le modèle de Hopfield fut présenté en 1982. Ce modèle très simple est basé sur le principe des mémoires associatives. C'est d'ailleurs la raison pour laquelle ce type de réseau est dit associatif (par analogie avec le pointeur qui permet de récupérer le contenu

d'une case mémoire). Le modèle de Hopfield utilise l'architecture des réseaux entièrement connectés et récurrents (dont les connexions sont non orientées et où chaque neurone n'agit pas sur lui-même). Les sorties sont en fonction des entrées et du dernier état pris par le réseau. [60]

2.2.7.2 Modèle de Kohonen

Ce modèle a été présenté par T. Kohonen en 1982 en se basant sur des constatations biologiques. Il a pour objectif de présenter des données complexes et appartenant généralement à un espace discret de grandes dimensions dont la topologie est limitée à une ou deux dimensions. Les cartes de Kohonen sont réalisées à partir d'un réseau à deux couches, une en entrée et une en sortie. Notons que les neurones de la couche d'entrée sont entièrement connectés à la couche de sortie (figure 2.9).

Les neurones de la couche de sortie sont placés dans un espace d'une ou de deux dimensions en général, chaque neurone possède donc des voisins dans cet espace. Et qu'enfin, chaque neurone de la couche de sortie possède des connexions latérales récurrentes dans sa couche (le neurone inhibe, les neurones éloignés et laisse agir les neurones voisins). [60]

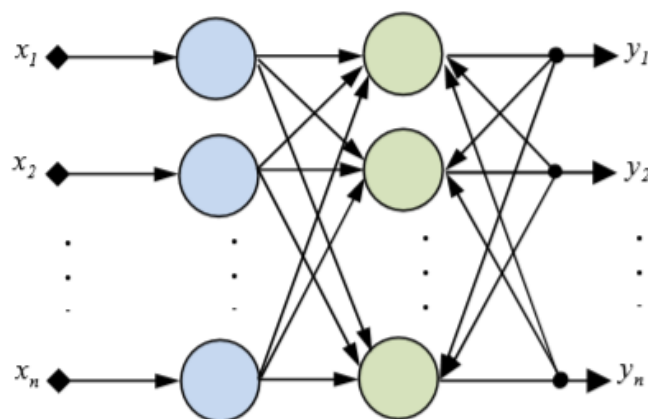


FIGURE 2.9 – Le modèle de Kohonen. [60]

2.2.7.3 Le modèle de Perceptron

Le mécanisme perceptron fut inventé par le psychologue F. Rosenblatt à la fin des années 50. Il représentait sa tentative d'illustrer certaines propriétés fondamentales des systèmes intelligents en générale.

Le réseau dans ce modèle est formé de trois couches : Une couche d'entrée, fournissant des données à une couche intermédiaire, chargée des calculs, cela en fournissant la somme des impulsions qui lui viennent des cellules auxquelles elle est connectée, et elle répond généralement suivant une loi définie avec un seuil, elle-même connectée à la couche de sortie (couche de décision), représentant les exemples à mémoriser. Seule cette dernière couche renvoie des signaux à la couche intermédiaire, jusqu'à ce que leurs connexions se stabilisent (figure 2.6). [60]

2.2.7.4 Le modèle ADALINE

L'ADALINE de Widrow et Hoff est un réseau à trois couches : une d'entrée, une couche cachée et une couche de sortie. Ce modèle est similaire au modèle de perceptron, seule la fonction de transfert change, mais reste toujours linéaire. Les modèles des neurones utilisés dans le perceptron et l'ADALINE sont des modèles linéaires. Séparation linéaire : on dit que deux classes A et B, sont linéairement séparables si on arrive à les séparer par une droite coupant le plan en deux (figure 2.10). Le problème est résolu avec les réseaux multicouches, car il peut résoudre toute sorte de problèmes qu'ils soient linéairement séparables ou non. [60]

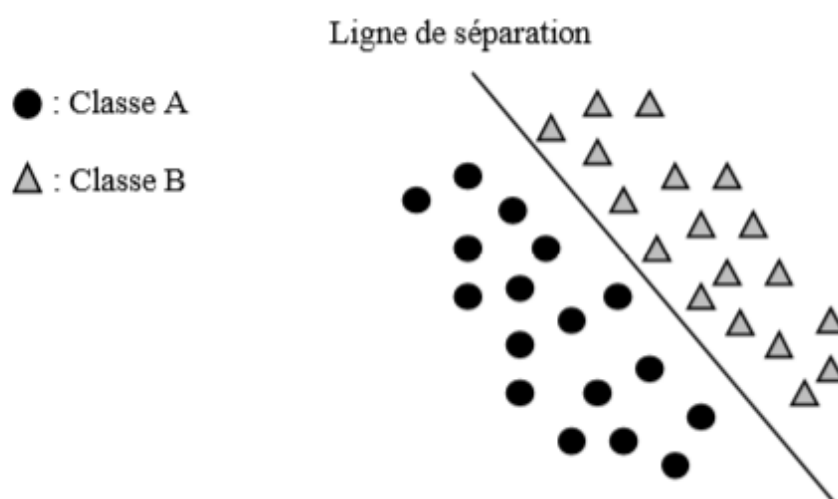


FIGURE 2.10 – La séparation linéaire entre la classe A et B. [60]

2.3 Classification de texte

La classification du texte est un exemple de reconnaissance des motifs, et peut se faire de deux façons différentes : la classification manuelle et automatique.[36]

Dans le premier, un annotateur humain interprète le contenu du texte et le classe en conséquence. Cette méthode peut fournir des résultats de qualité, mais il est coûteux, consomme beaucoup de temps, et vulnérable aux erreurs humaines. En outre, avec la dernière montée de taille des données et la variété des classes de documents, le processus manuel n'est certainement pas évolutif et peu pratique.[76]

Le dernier applique l'apprentissage automatique, le traitement du langage naturel et d'autres techniques pour classer automatiquement le texte de manière plus rapide et plus rentable.

En raison de la forte augmentation de la taille des données au cours des deux dernières décennies, un processus d'automatisation est nécessaire pour atteindre les objectifs d'extraction d'informations et de classification des données à des fins diverses.[76]

Ceux-ci incluent le filtrage et le routage des e-mails, observation des presses, filtrage des spams et moteurs de recherche.

La classification est l'une des applications d'exploration de données ou bien le Data Mining. L'exploration de données est un domaine informatique qui traite du développement de moyens d'extraire des informations de masses de données existantes. [76]

Certaines approches de classification automatique des textes ont été rapportées au cours des trois dernières décennies. De nombreux algorithmes ont été introduits et évalués, qui s'adressent aux algorithmes de classification du texte les plus courants. [76]

La classification peut être effectuée sur n'importe quel ensemble de données. La capacité de la classification de texte à travailler sur un ensemble de données étiqueté (dans le cas d'une automatisation de gestion de relation clientèle) ou non étiqueté (lecture des sentiments sociaux en ligne) élargit l'espace où cette technologie peut être mise en œuvre. [39]

Dans la terminologie de l'apprentissage automatique, la classification est considérée comme un exemple d'apprentissage supervisé, c'est-à-dire d'apprentissage où un ensemble de formation d'observations correctement identifiées est disponible. [36]

La classification de texte est le processus de classification des documents en catégories prédéfinies en fonction de leur contenu. [38] Si le nombre de catégories dépasse deux classes, il s'agit d'une classification multi-classes. La classification de texte est une exigence principale des systèmes de recherche d'information, qui récupèrent des textes en réponse à une requête de l'utilisateur, et des systèmes de compréhension de texte, qui transforment le texte d'une manière ou d'une autre, comme la production de résumés, la réponse à des questions ou l'extraction de données. Les algorithmes d'apprentissage supervisé existants pour classer le texte ont besoin de suffisamment de documents pour apprendre avec précision. [69]

Les données texte des documents peuvent être affectées à aucune, une ou même plusieurs catégories. La classification multi-étiquette (à ne pas confondre avec la classification multi-classes) est une variante de classification lorsque plusieurs étiquettes peuvent être affectées à chaque instance. La classification multi-étiquettes est une généralisation de la classification multi-classes, qui est l'instance de catégorisation à étiquette unique dans précisément une de plus de deux classes. Dans le problème multi-étiquettes, il n'y a aucune contrainte sur le nombre de classes auxquelles l'instance peut être affectée. [25]

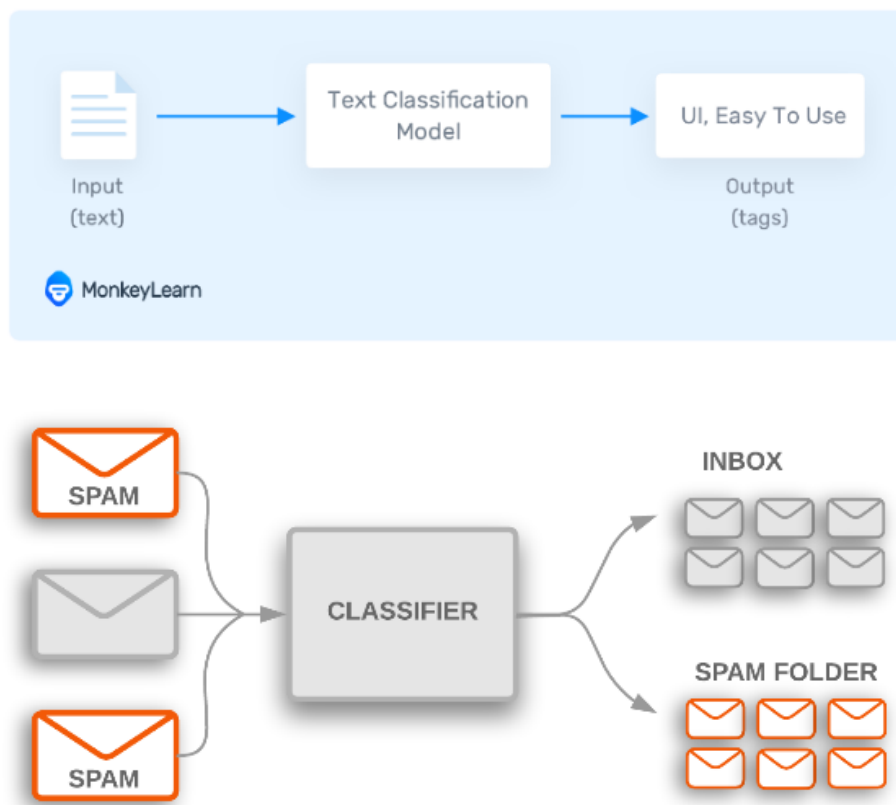


FIGURE 2.11 – Classification de texte. [37] [76]

2.3.1 Approches / techniques

La classification des textes peut être effectuée avec deux techniques / méthodologies principales, les techniques statistiques et les techniques d'apprentissage automatique. [76]

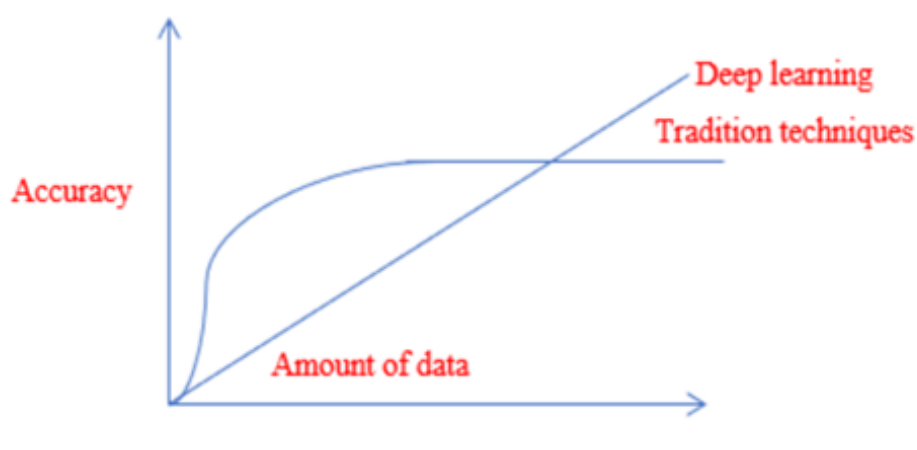


FIGURE 2.12 – Les techniques d'apprentissage. [76]

2.3.1.1 Techniques statistiques

Les techniques de classification des textes statistiques sont basées sur des fondements mathématiques. Ces techniques ont été développées relativement plus tôt que les techniques d'apprentissage automatique et conviennent mieux à des ensembles de données relativement petits. Certains d'entre eux conviennent également mieux aux classifications binaires qu'à la classification multi-classe. Des exemples de ces techniques sont : les procédures fréquentistes, les procédures bayésiennes et les procédures binaires et multi-classes. [76]

2.3.1.2 Techniques d'apprentissage automatique

La classification du texte avec l'apprentissage automatique apprend à effectuer des classifications basées sur des observations du passées. En utilisant des exemples pré-étiquetés comme données d'apprentissage, un algorithme d'apprentissage automatique peut apprendre les différentes associations entre des parties de texte, et qu'une sortie particulière (des étiquettes) est attendue pour une entrée particulière (du texte). [37]

La première étape vers la formation d'un classifieur avec l'apprentissage automatique est l'extraction des fonctionnalités : une méthode est utilisée pour transformer chaque texte en une représentation numérique sous la forme d'un vecteur. L'une des approches les plus fréquemment utilisées est le sac de mots (Bag of Words), où un vecteur représente la fréquence d'un mot dans un dictionnaire de mots prédéfini. [37]

Ensuite, l'algorithme d'apprentissage automatique est alimenté avec des données d'apprentissage qui se composent de paires d'ensembles de fonctionnalités (vecteurs pour chaque exemple de texte) et des étiquettes (par exemple, sports, politique) pour produire un modèle de classification : [37]

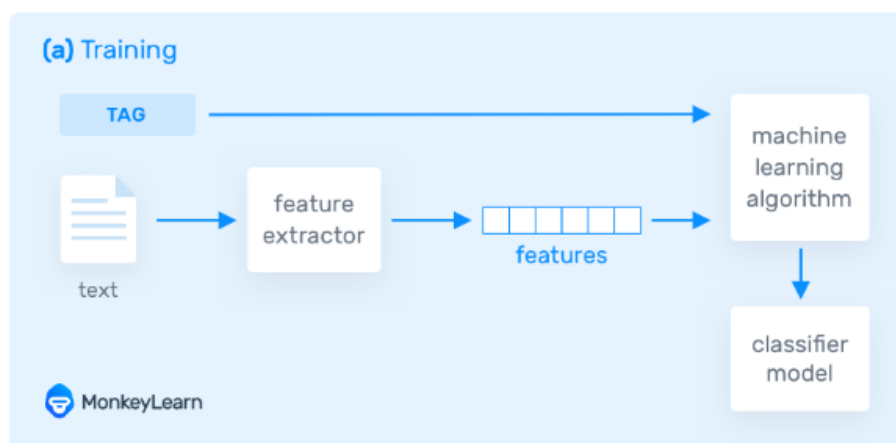


FIGURE 2.13 – Techniques d'apprentissage automatique. [37]

Une fois qu'il est formé avec suffisamment d'échantillons de formation, le modèle d'apprentissage automatique peut commencer à faire des prédictions précises. Le même extracteur de fonctionnalités est utilisé pour transformer le texte inaperçu en des ensembles de fonctionnalités qui peuvent être introduits dans le modèle de classification pour obtenir des prédictions sur les étiquettes. [37]

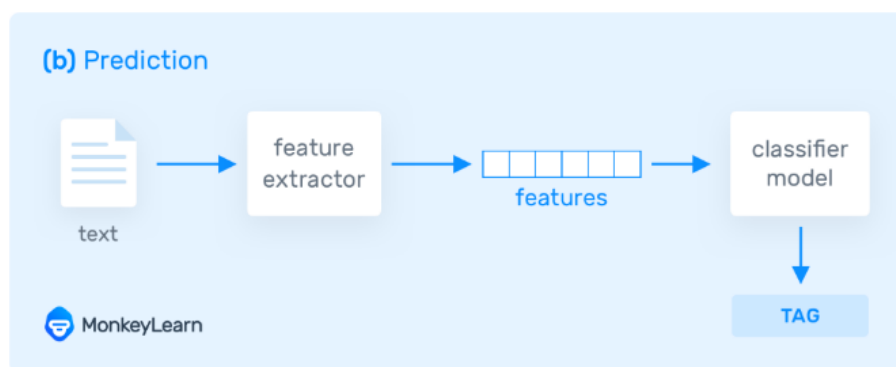


FIGURE 2.14 – Processus de formation d'un classifieur de texte d'apprentissage automatique. [37]

La classification de texte avec apprentissage automatique est généralement beaucoup plus précise que des techniques précédentes, en particulier pour les tâches de classification complexes. De plus, les classificateurs avec apprentissage automatique sont plus faciles à maintenir et vous pouvez toujours étiqueter de nouveaux exemples pour apprendre de nouvelles tâches.[37]

2.3.2 Étapes de classification du texte

La classification des textes en général comporte six étapes principales

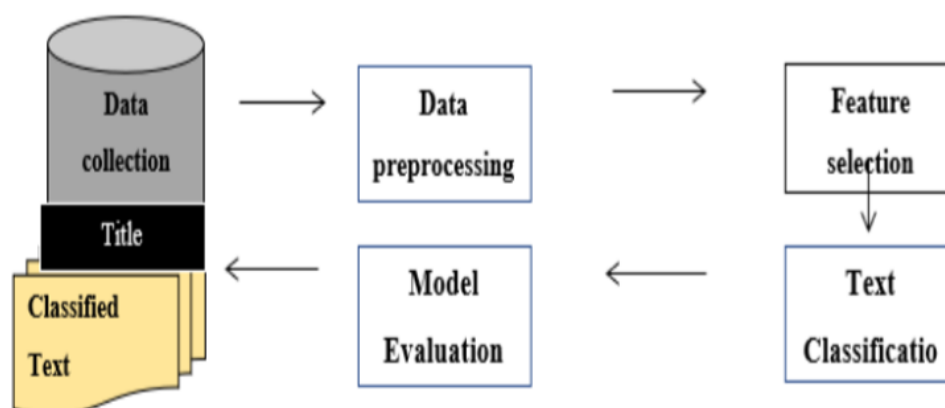


FIGURE 2.15 – Étapes de classification du texte. [76]

2.3.2.1 Collecte de données

La quantité de données nécessaires pour former les classificateurs est en effet une question ouverte. Néanmoins, les meilleures pratiques montrent que plus de données ont tendance à produire de meilleures performances, en particulier pour les modèles d'apprentissage profond. Cependant, un ensemble de données petit mais pertinent peut surpasser un ensemble plus grand mais moins pertinent. Il est sûr qu'il n'y a pas de règles claires et rapides pour ce type de compromis. Seulement, empiriquement, nous pouvons signaler un tel compromis.[50]

Il y a trois facteurs à considérer lors de la collecte de données pour une tâche de classification : [14]

- Nombre d'échantillons : nombre total d'exemples que vous avez dans les données.
- Nombre de classes : nombre total de sujets ou de catégories dans les données.
- Nombre d'échantillons par classe : Nombre d'échantillons par classe (sujet / catégorie).

Dans un ensemble de données équilibré, toutes les classes auront un nombre similaire d'échantillons; dans un ensemble de données déséquilibré, le nombre d'échantillons dans chaque classe variera considérablement. [14]

2.3.2.2 Prétraitement des données

Grâce à des décennies de recherche, nous avons accès à un large éventail d'options de prétraitement des données et de configuration des modèles. Cependant, la disponibilité d'un très large éventail d'options viables à choisir augmente considérablement la complexité et la portée du problème particulier à résoudre. Étant donné que les meilleures options peuvent ne pas être évidentes, une solution naïve serait d'essayer toutes les options possibles de manière exhaustive, en élaguant certains choix par intuition. Cependant, cela coûterait extrêmement cher.[6]

Le prétraitement de l'ensemble de données est une étape importante car il augmente la précision de la classification et réduit la taille de la mémoire requise pour le processus de classification. [76]

Le but de cette étape est de supprimer les données inutiles qui ne contribuent pas à la sémantique du document telles que les arrêts des mots (Stop Words). Les pronoms sont des exemples de ces mots. Cette étape supprime également les suffixes et les préfixes. Il combine également des mots de la même racine / origine, ce qui est fait par des algorithmes de tige (Stemming algorithms). L'objectif principal de cette étape est de réduire l'ensemble des fonctionnalités d'un document donné et de fournir une meilleure précision de classification. [76]

2.3.2.3 Extraction / sélection de fonctionnalités

L'extraction et la sélection d'entités sont deux étapes importants du prétraitement du texte avant la classification. Le processus d'extraction des caractéristiques vise à transformer le texte non structuré en une représentation structurée et à supprimer la redondance, ce qui facilite le traitement ultérieur et l'application de techniques d'apprentissage automatique. La sélection des fonctionnalités est une autre étape de prétraitement pour exclure les fonctionnalités non pertinentes et réduire la dimensionnalité élevée du résultat de l'étape précédente d'extraction des fonctionnalités. Il existe trois catégories principales d'algorithmes de sélection de fonctionnalités : l'emballage, le filtre et l'embarqué.[76]

2.3.2.4 Classification de texte

La classification du texte peut être effectuée à l'aide de nombreux algorithmes, comme suit :

Les techniques statistiques ont utilisé des algorithmes comme les modèles bayésiens naïfs qui ont présenté d'excellents résultats dans le domaine de la classification de texte. Le plus célèbre modèle bayésien naïf était connu comme le classificateur binaire indépendant. [76]

Dans la classification des textes distinctifs, des stratégies ont été développées pour trier les documents, par exemple, K-voisin le plus proche (KNN), les différents modèles KNN calculent les distances entre les termes d'index de document et les termes connus de chaque catégorie en appliquant des fonctions de distance, pour exemple, cosinus, similitude de dés ou fonctions euclidiennes, les classes retournées sont les k-èmes classes avec les scores les plus notables. [76]

Les techniques d'apprentissage automatique s'appuyaient sur des arbres de décision : qui sont utilisés pour classer les documents en construisant un arbre en calculant la fonction d'entropie des termes d'index sélectionnés. [76]

Le support des machines à vecteur (Support vector machines) sont considérées comme l'un des classificateurs de texte les plus connus. Les SVM sont l'une des techniques d'apprentissage automatique supervisé. Dans les SVM, un algorithme d'apprentissage est utilisé pour construire un modèle qui sera utilisé pour attribuer un nouveau document inconnu à une catégorie à partir d'un ensemble de catégories prédéfinies. Les SVM peuvent être utilisés pour effectuer une classification linéaire et non linéaire.[76]

2.3.2.5 Évaluation du classificateur

La classification des textes est évaluée en fonction de l'efficacité et de l'efficacité de la catégorisation. Certaines techniques ont été utilisées pour quantifier l'avancement du classificateur. L'une de ces techniques est F1, la précision et le rappel qui sont utilisées dans le domaine de la recherche d'informations et de l'apprentissage automatique. [76]

2.3.3 Algorithmes de classification du texte

2.3.3.1 La régression logistique

La régression logistique, malgré son nom, est un modèle linéaire de classification plutôt que de régression. La régression logistique est également connue dans la littérature sous le nom de régression logit, classification d'entropie maximale (MaxEnt) ou classificateur log-linéaire. Dans ce modèle, les probabilités décrivant les résultats possibles d'un seul essai sont modélisées à l'aide d'une fonction logistique. [18]

Cette implémentation peut s'adapter à une régression logistique binaire, un contre repos ou multinomiale avec une régularisation optionnelle l1,l2 ou Elastic-Net. [18]

Plusieurs solveurs sont implémentés dans la classe LogisticRegression de scikit-learn, le solveur « lbfgs » ne prend en charge que la régularisation l2 ou aucune régularisation, et il s'avère qu'il converge plus rapidement pour certaines données de grande dimension, il est utilisé par défaut pour sa robustesse. [18]

En tant que problème d'optimisation, la régression logistique pénalisée de classe binaire l2 minimise la fonction de coût suivante : [18]

$$\min_{w,c} \frac{1}{2} w^T w + C \sum_{i=1}^n \log(\exp(-y_i(X_i^T w + c)) + 1) \quad (2.10)$$

2.3.3.2 Bayes naïfs multinomiaux

Bayes naïfs multinomiaux est l'une des deux variantes Bayes naïves classiques utilisées dans la classification de texte. La distribution est paramétrée par les vecteurs $\Theta = (\Theta_{y1}, \dots, \Theta_{yn})$ pour chaque classe y , où n est le nombre d'entités (la taille du vocabulaire dans

notre cas) et Θ_{yi} est la probabilité $P(x_i | y)$ de l'entité i apparaissant dans un échantillon appartenant à la classe y . [26]

Les paramètres Θ_y est estimée par une version lissée du maximum de vraisemblance, c'est-à-dire le comptage de fréquence relative : [26]

$$\hat{\Theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha n} \quad (2.11)$$

Où $N_{yi} = \sum_{x \in T} x_i$ est le nombre de fois que la fonction i apparaît dans un échantillon de la classe y dans l'ensemble d'apprentissage T , et $N_y = \sum_{i=0}^n N_{yi}$ est le nombre total de toutes les fonctionnalités de la classe y . [26]

Le lissage a priori $\alpha \geq 0$ tient compte des caractéristiques non présentes dans les échantillons d'apprentissage et empêche les probabilités nulles dans les calculs ultérieurs. La définition de $\alpha = 1$ est appelée lissage de Laplace, tandis que $\alpha < 1$ est appelée lissage de Lidstone. [26]

2.3.3.3 Classificateur de forêt aléatoire

Dans le classificateur de forêt aléatoire chaque arbre de l'ensemble est construit à partir d'un échantillon prélevé avec remplacement de l'ensemble d'apprentissage. De plus, lors du fractionnement de chaque nœud lors de la construction d'un arbre, le meilleur fractionnement est trouvé soit à partir de toutes les entités en entrée, soit d'un sous-ensemble aléatoire de taille `max_features`. [16]

Le but de ces deux sources de hasard est de diminuer la variance de l'estimateur forestier. En effet, les arbres de décision individuels présentent généralement une variance élevée et ont tendance à s'adapter. Le caractère aléatoire injecté dans les forêts donne des arbres de décision avec des erreurs de prédiction quelque peu découplées. En prenant une moyenne de ces prévisions, certaines erreurs peuvent être annulées. Les forêts aléatoires atteignent une variance réduite en combinant divers arbres, parfois au prix d'une légère augmentation du biais. En pratique, la réduction de la variance est souvent significative, ce qui donne un meilleur modèle global. [16]

L'implémentation scikit-learn combine les classificateurs en faisant la moyenne de leur prédiction probabiliste, au lieu de laisser chaque classificateur voter pour une seule classe. [16]

2.3.3.4 Classificateur de Vecteur de Support Linéaire

Ce classificateur est capable d'effectuer une classification multi classe sur un ensemble de données. C'est une implémentation de Classification des Vecteurs de Support pour le cas d'un noyau linéaire. La fonction de décision des SVM dépend d'un sous-ensemble des données d'apprentissage, appelé vecteurs de support, une fois ajusté, le modèle peut être utilisé pour prédire de nouvelles valeurs. [7]

Le classificateur de Vecteur de Support Linéaire implémente une stratégie multi-classes « un contre le reste », entraînant ainsi des modèles `n_class`, s'il n'y a que deux classes, un seul modèle est formé. [7]

L'algorithme prend en entrée deux tableaux : un tableau X de taille $[n_{samples}, n_{features}]$ contenant les échantillons d'apprentissage, et un tableau y d'étiquettes de classe (chaînes ou entiers) de taille $[n_{samples}]$. [7]

Les attributs `coef__` et `intercept__` ont respectivement la forme $[n_{class}, n_{features}]$ et $[n_{class}]$, chaque ligne des coefficients correspond à l'un des nombreux classificateurs `n__class` « un contre le reste » et similaire pour les intersections, dans l'ordre de la classe « une ». Avec chaque ligne correspondant maintenant à un classificateur binaire. L'ordre des classes 0 à `n` est «0 vs 1 », «0 vs 2 », ... «0 vs `n` », «1 vs 2 », «1 vs 3 », «1 vs `n` », . . . « `N-1` vs `n` ». [7]

2.3.4 Classification des textes en arabe

Pour la langue arabe, il y a un manque d'études sur la classification des documents texte arabes avec une limitation de l'ensemble de données gratuit d'analyse comparative. D'un autre côté, la richesse morphologique de la langue arabe augmente considérablement la longueur du vecteur caractéristique et cela a considérablement influencé la recherche et les études dans le domaine de la classification des textes. [76]

Il y a 28 lettres en langue arabe, en plus du hamza arabe (ﺀ) qui est considéré comme une lettre par certains linguistes arabes et il est écrit de droite à gauche. Il a deux genres : féminin et masculin. Les nombres sont des nombres singuliers, doubles et pluriels. Grammatical sont trois cas : nominatif, accusatif et génitif. Un nom a trois cas linguistiques : cas nominatif lorsqu'il est sujet ; cas accusatif quand il fait l'objet d'un verbe ; et le cas génitif quand il fait l'objet d'une préposition. L'arabe n'utilise pas de majuscules / minuscules. Il emploie également des signes diacritiques qui représentent une petite lettre de voyelle comme « fatha, kasra, damma, sukun, shadda et tanween ». [76]

Il existe de nombreux ensembles de données standard pour la classification des textes en anglais librement accessibles. Malheureusement pour la langue arabe, The Open Source Arabic Corpus est disponible gratuitement mais n'est pas standardisé. La majorité des auteurs ont utilisé un ensemble de données arabe construit en interne avec différentes tailles et contenus, avec les sites Internet comme principale source de données. La portée des documents choisis qui ont été introduits dans l'ensemble de données variait de 240 documents répartis en six catégories à 17 658 documents répartis en sept genres. [76]

La taille des corpus utilisés est relativement petite par rapport au plus grand corpus anglais disponible (400 millions de mots). Le plus grand corpus arabe recensé contenait environ 310 millions de mots collectés sur trois sites Web uniquement et couvrant cinq catégories, dont une générale. Ceci n'est rapporté que récemment et n'est pas standardisé. En outre, le nombre de catégories est relativement faible par rapport à la nature et à la richesse de la langue arabe par rapport à une autre langue.[76]

Une différence significative entre la classification textuelle de la langue arabe et d'autres langues couramment utilisées telles que l'anglais est le fait que la racine pourrait affecter de manière significative le résultat de la classification. Certaines œuvres utilisaient le stemming basé sur les racines tandis que d'autres utilisaient des stemmers légers. Il a été démontré que l'utilisation de souches légères produit une meilleure mesure des performances que les souches basées sur les racines. L'arabe est une langue très flexionnelle et riche sur le plan morphologique et les mots arabes peuvent provenir d'une tige de mots de trois, quatre, cinq et six lettres. Près de 80% des mots arabes proviennent d'une racine à trois lettres. [76]

La majorité des travaux sur la classification de la langue arabe ont été effectués au cours de la décennie (2000 - 2010) à l'exception de quelques incidents. Ils ont appliqué les techniques de classification traditionnelles telles que SVM, NB, k-NN, Decision Trees

et ANN. Peu d'incidences ont proposé des classificateurs combinés tels que l'entropie maximale, maître-esclave et BSO-SVM. [76]

Une récente recherche intéressante concernant la classification des textes en arabe (publiée en 2020) a été une étude comparative sur neuf modèles d'apprentissage approfondi pour la catégorisation de textes arabes à la fois uniques et multi-étiquettes, et une autre étude de l'impact de l'utilisation des modèles d'intégration de word2vec pour améliorer les performances des tâches de classification. [50]

Alors que tous ont donné un haut niveau de précision (avec une précision minimale de 93,43 et des performances optimales de 95,81%) sur la classification à étiquette unique, il a diminué sur la classification à étiquettes multiples, donnant une précision de 70,34% pour un sous-ensemble maximum de 8 catégories, jusqu'à 88,68% pour un sous-ensemble maximum de 10 catégories. [50]

Les chercheurs ont construit deux nouveaux grands corpus bien annotés pour les tâches de classification arabe, collectés sur plusieurs portails d'actualités. À savoir, SANAD (ensemble de données d'articles d'actualités arabes en une seule étiquette) et NADIA (ensemble de données d'articles d'actualités multi-étiquettes en arabe). Chaque corpus se compose de plusieurs jeux de données. Les deux sont disponibles sur Mendely pour la communauté de recherche sur la linguistique informatique arabe. [50]

2.4 Apprentissage de similarité

Dans l'exploration de données, l'apprentissage automatique, la reconnaissance de formes, l'analyse statistique et d'autres applications informatiques, la similitude est utilisée pour présenter une mesure de ressemblance entre deux objets. [85]

Informellement, la similitude entre deux objets est une mesure numérique du degré de similitude des deux objets. Plus ils sont similaires, plus le degré de similitude est élevé. [85]

La dissimilarité entre deux objets est une mesure numérique du degré de différence entre les deux objets. Plus la dissimilarité entre les objets est faible, plus les paires d'objets sont similaires. Fréquemment, le terme distance est utilisé comme synonyme de dissimilarité, bien qu'il soit souvent utilisé pour désigner une classe spéciale de dissemblances. [85]

L'apprentissage de la similarité est étroitement lié à l'apprentissage à distance métrique. L'apprentissage métrique est la tâche d'apprendre une fonction de distance sur des objets. Une fonction métrique ou de distance doit vérifier quatre axiomes : la non-négativité, l'identité des indiscernables, la symétrie et la sous-additivité (ou l'inégalité du triangle). En pratique, les algorithmes d'apprentissage métrique ignorent la condition d'identité des indiscernables et apprennent une pseudo-métrique. [31]

2.4.1 Types des données

La définition de similitude et son calcul sont différents avec différents types de données. Par exemple, la mesure de similitude de deux objets numériques est souvent convertie en dissimilarité, celle-ci étant généralement une distance euclidienne. Il est utilisé pour présenter le degré de diversité entre deux objets ; et pour les données nominales, la similitude entre deux objets est liée au nombre de mêmes valeurs qui proviennent de leurs attributs correspondants. Selon les types de données des valeurs d'attribut, nous les divisons en trois types de données : type numérique, type non numérique et type mixte. [85]

2.4.1.1 Numérique

Également connu sous le nom de variables continues ou quantitatives, à savoir qu'il existe des valeurs infinies entre deux valeurs différentes des attributs numériques spécifiques. Souvent, les nombres naturels ou les unités de mesure sont utilisés pour mesurer directement les similitudes, telles que la température, la hauteur, etc. Les variables numériques peuvent être divisées en variables à échelle d'intervalle et variables d'échelle, tandis que les variables à échelle d'intervalle sont une variable d'échelle linéaire, et les variables d'échelle sont généralement non linéaires. [85]

2.4.1.2 Non numérique

Les valeurs d'attribut sont des données non quantitatives mais qualitatives, telles que le sexe, le nom, la citoyenneté, etc. d'une personne. Habituellement, ce type de valeurs d'attribut est un nombre fini d'états (lettre ou nombre ordinal). Les attributs non numériques peuvent être modifiés en attributs nominaux, attributs binaires et attributs ordinaux. Les attributs nominaux ne sont pas ordonnés, tandis que les ordinaux sont ordonnés. Le type mixte se réfère généralement au type mixte par type numérique et non numérique. [85]

2.4.2 Approches de calcul de similarité

L'apprentissage de la similarité est un domaine de l'apprentissage automatique supervisé en intelligence artificielle. Il est étroitement lié à la régression et à la classification, mais l'objectif est d'apprendre une fonction de similarité qui mesure la similitude ou la relation entre deux objets. [31]

De nombreuses approches d'apprentissage automatique reposent sur une métrique. Cela inclut l'apprentissage non supervisé tel que le clustering, qui regroupe des objets proches ou similaires. Il comprend également des approches supervisées comme l'algorithme K-plus proche voisin qui s'appuie sur les étiquettes des objets voisins pour décider de l'étiquette d'un nouvel objet. L'apprentissage métrique a été proposé comme étape de prétraitement pour bon nombre de ces approches. [31]

2.4.2.1 Mesures basées sur la similarité

Les méthodes basées sur la similarité déterminent les objets les plus similaires avec les valeurs les plus élevées car cela implique qu'ils vivent dans des quartiers plus proches. [5]

2.4.2.1.1 La similarité cosinus La similitude de cosinus mesure la similitude entre deux vecteurs d'un espace de produit intérieur. Il est mesuré par le cosinus de l'angle entre deux vecteurs et détermine si deux vecteurs pointent à peu près dans la même direction. Il est souvent utilisé pour mesurer la similitude des documents dans l'analyse de texte. Notez que à cause que la mesure de similitude cosinus ne vérifie pas toutes les propriétés de la définition des mesures métriques, elle est appelée mesure non métrique. [9]

La similitude cosinus calcule le cosinus de l'angle entre deux vecteurs, afin de calculer la similitude cosinus, nous utilisons la formule suivante :

$$s(x, y) = \frac{x \cdot y}{\|x\| \cdot \|y\|} = \frac{\sum_{i=0}^{n-1} x_i \cdot y_i}{\sqrt{\sum_{i=0}^{n-1} (x_i)^2} \cdot \sqrt{\sum_{i=0}^{n-1} (y_i)^2}} \quad (2.12)$$

Rappelez la fonction cosinus : à gauche les vecteurs rouges pointent sous différents angles et le graphique à droite montre la fonction résultante.

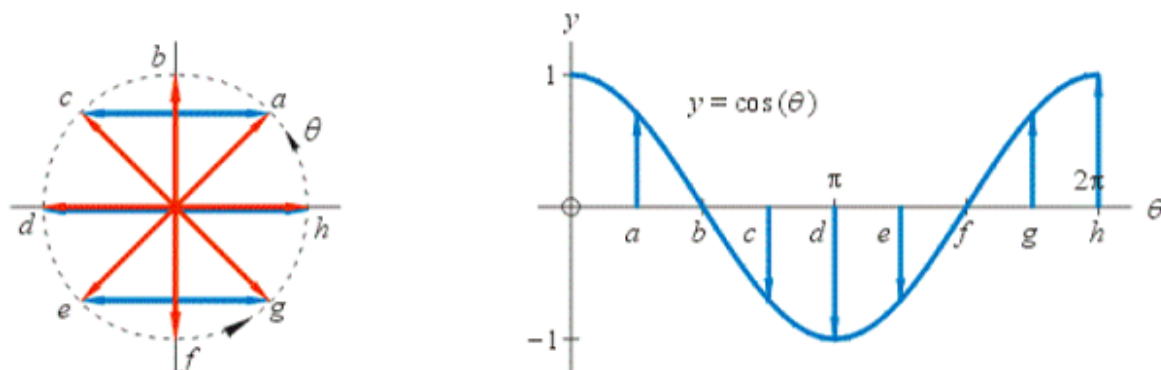


FIGURE 2.16 – La fonction de cosinus. [5]

Par conséquent, la similitude cosinus peut prendre des valeurs comprises entre -1 et +1. Si les vecteurs pointent dans la même direction, la similitude cosinus est +1. Si les vecteurs pointent dans des directions opposées, la similitude cosinus est -1. [5]

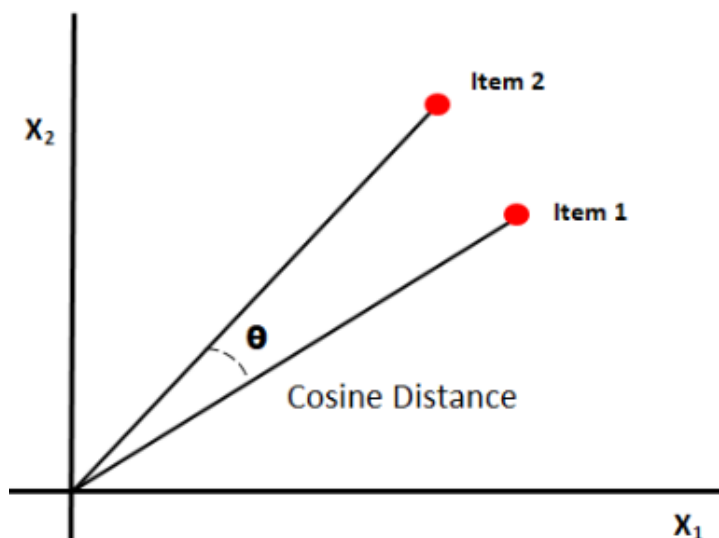


FIGURE 2.17 – La distance cosinus. [5]

La similitude cosinus est très populaire dans l'analyse de texte. Il est utilisé pour déterminer la similitude des documents entre eux, quelle que soit leur taille. La technique d'analyse de texte TF-IDF permet de convertir les documents en vecteurs où chaque valeur dans le vecteur correspond au score TF-IDF d'un mot dans le document. Chaque mot a son propre axe, la similitude cosinus détermine ensuite la similitude des documents. [5]

2.4.2.2 Mesures basées sur la distance

Les méthodes basées sur la distance priorisent les objets avec les valeurs les plus faibles pour détecter leur similitude. [5]

2.4.2.2.1 Distance euclidienne La distance euclidienne est une distance en ligne droite entre deux vecteurs, pour les deux vecteurs x et y , cela peut être calculé comme suit :

$$EUC = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.13)$$

Par rapport à la similitude cosinus, la distance euclidienne n'est pas très souvent utilisée dans le cadre des applications TAL. Il convient aux variables numériques continues. La distance euclidienne n'est pas invariante à l'échelle, il est donc recommandé de mettre à l'échelle les données avant de calculer la distance. De plus, la distance euclidienne multiplie l'effet des informations redondantes dans l'ensemble de données. [5]

2.4.3 L'évaluation des modèles de similarité

Notez qu'il n'y a aucun moyen d'évaluer les performances de ces modèles, une fois qu'ils sont formés, ils sont prêts à être déployés, seule l'inspection de leurs résultats peut prouver si les modèles ont réussi à apprendre les bonnes propriétés des différents objets.

2.4.4 Modèles de similarité

La prochaine étape est d'extraire les questions similaires à la question posée, cette fois on peut utiliser tous nos données car chaque class va posséder leur propre modèle, grâce aux nombreux modèles de Gensim, on a pu tester 4 modèles de similarité différents, y compris :

2.4.4.1 Fréquence du terme * Fréquence inverse du document

Tf-Idf attend un corpus de formation de sac de mots (valeurs entières) lors de l'initialisation. Lors de la transformation, il prendra un vecteur et retournera un autre vecteur de même dimensionnalité, sauf que les caractéristiques qui étaient rares dans le corpus d'entraînement verront leur valeur augmentée. Il convertit donc les vecteurs à valeur entière en vecteurs à valeur réelle, tout en laissant intact le nombre de dimensions. Il peut également éventuellement normaliser les vecteurs résultants à la longueur unitaire (euclidienne). [40]

2.4.4.2 L'indexation sémantique latente

LSI (ou parfois LSA) transforme les documents d'un espace de mots ou (de préférence) pondéré en TfIdf en un espace latent de dimensionnalité inférieure. Sur les corpus réels, une dimensionnalité cible de 200 à 500 est recommandée comme « norme d'or ». [40]

La formation LSI est unique, car nous pouvons la poursuivre à tout moment, simplement en fournissant plus de documents de formation. Cela se fait par des mises à jour

incrémentielles du modèle sous-jacent, dans un processus appelé formation en ligne. En raison de cette fonctionnalité, le flux de documents d'entrée peut même être infini. [40]

2.4.4.3 Allocation Dirichlet Latente

LDA est une autre transformation du sac de mots, compte dans un espace thématique de dimensionnalité inférieure. LDA est une extension probabiliste de LSA (également appelée PCA multinomiale), de sorte que les sujets de LDA peuvent être interprétés comme des distributions de probabilité sur les mots. Ces distributions sont, tout comme avec LSA, déduites automatiquement d'un corpus de formation. Les documents sont à leur tour interprétés comme un mélange (doux) de ces sujets (encore une fois, comme avec LSA). [40]

2.4.4.4 Doc2Vec (Modèle Vectoriel de Paragraphe)

L'algorithme Doc2Vec surpasse généralement une telle moyenne simple des vecteurs Word2Vec, l'idée de base est d'agir comme si un document avait un autre vecteur flottant semblable à un mot, qui contribue à toutes les prédictions d'apprentissage, et est mis à jour comme les autres vecteurs de mots, mais nous l'appellerons un doc-vecteur. La classe Doc2Vec de Gensim implémente cet algorithme, qui fait référence au modèle vectoriel de paragraphe. [12]

Conclusion

Tout au long de ce chapitre, nous avons détaillé l'apprentissage profond et ses concepts, les RNA avec leurs architectures et leurs modèles ainsi que les techniques de classification de texte, ses algorithmes et ses étapes de traitement. Nous avons aussi présenté les différentes techniques utilisées dans le calcul de similarité distributionnelle. Dans ce qui suit, nous détailleront notre approche pour la réalisation d'un système de question/réponse pour les Fatwa islamiques.

Chapitre 3

Contribution et implémentation

Introduction

L'approche que nous avons proposée à travers ce projet consiste en deux étapes essentielles, le routage des requêtes (demande de Fatwa) et la recherche des Fatwas adéquates en calculant la similarité sémantique avec la banque des Fatwas existantes. Sachant que la réponse aux demandes de Fatwas est un sujet critique dans la charia islamique, la raison pour laquelle, le système proposé est ainsi divisé. Le routage des requêtes consiste à classer la question dans la bonne catégorie de la charia (Salat, Sawm,...). La similarité avec les Fatwas existantes est calculée en utilisant des algorithmes de calcul de similarité distributionnelle. Il est à noter que pour chacune des deux étapes (routage ou calcul de similarité), un ensemble de modèles d'apprentissage ont été implémentés. La figure 1 représente l'architecture globale de notre approche.

3.1 Architecture globale (fonctionnement)

Notre architecture de déploiement de système est détaillée dans la figure suivante :

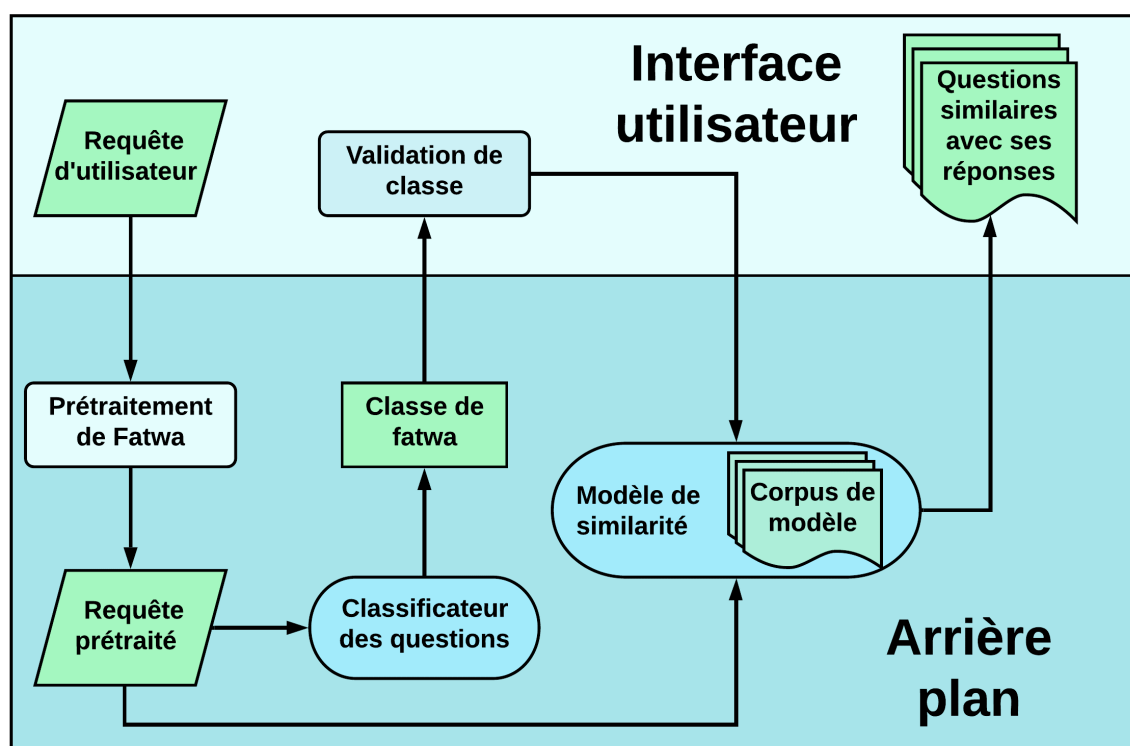


FIGURE 3.1 – Architecture de fonctionnement.

La première étape a le but de transformer la requête de format textuelle au format plus adéquate pour les modèles de l'apprentissage automatique.

Les questions des fatwas consistent d'une norme éthique, car le questionneur s'adresse à un religieux (connu sous le nom de Mufti) que nous avons dû supprimer à cette étape.

Après la reformulation des fatwas, un modèle de classification prédire la classe associée à la requête, et retourne le résultat vers l'utilisateur pour la confirmer ou la changer.

La requête ensuite va être comparé avec tous les données de la classe associé afin de distinguer les questions les plus similaires. Cette étape implémente le module de recherche d'information dans un système de question réponse, et nous l'avons implémenté avec les modèles de similarité.

Seulement les 5 premiers résultats de chaque modèle sont affichés d'une manière descendante de classement de similitude, avec un seuil de 50 % de similitude.

3.2 Architecture de formation

La formation des modèles d'apprentissage automatique comporte trois étapes essentielles, le prétraitement, l'entraînement, et le teste.

Le prétraitement comprend diverses tâches de traitement automatique de la langue, telles que la tokenisation et la suppression de mots vides, afin d'avoir les meilleures performances par les modèles.

L'entraînement est le corps de l'apprentissage automatique, les modèles se transforme de niveau totalement naïf vers un niveau qui peut dépasser celle des humains dans le domaine étudié.

Le teste est aussi important pour déterminer le niveau des modèles entraînés, c'est la phase de jugement des compétences des modèles.

Dans ce chapitre on va présenter deux premières étapes, tandis que la dernière étape va être présenté dans le dernier chapitre.

3.2.1 Préparation de corpus

Avant toutes les étapes mentionnées ci-dessus, la première étape consiste en fait à collecter les données nécessaires pour alimenter les modèles d'apprentissage. Le corpus utilisé extrait du corpus collecté par Setti et Belaribi [84].

Le manque des travaux similaires a empêché d'avoir un corpus prêt à utiliser, donc ils ont collecté leur propre corpus manuellement à partir des sites web de trois muftis très connus dans le domaine des fatwas islamiques, "Abd Elaziz Ibn Abdallah Ibn Baz" et "Mohammed Ibn Salah Elotheimeen" (que la miséricorde d'Allah soit sur eux) et "Mohammed Ali Ferkous" (qu'Allah le protège).

Les fatwas sont écrites en arabe classique et moderne, et sont rassemblées dans différentes catégories (chapitres) de la jurisprudence islamique. Le nombre total de fatwas collectées est de 3000 fatwas extraites de 12 catégories différentes. [84]

Nous avons pris environ la moitié de corpus, 1224 fatwas distribué en 4 types des fatwas notamment : 'Hadj' avec 119 fatwas, 'Salat' avec 554 fatwas, 'Sawm' avec 245 fatwas, et 'Zakat' avec 306 fatwas.

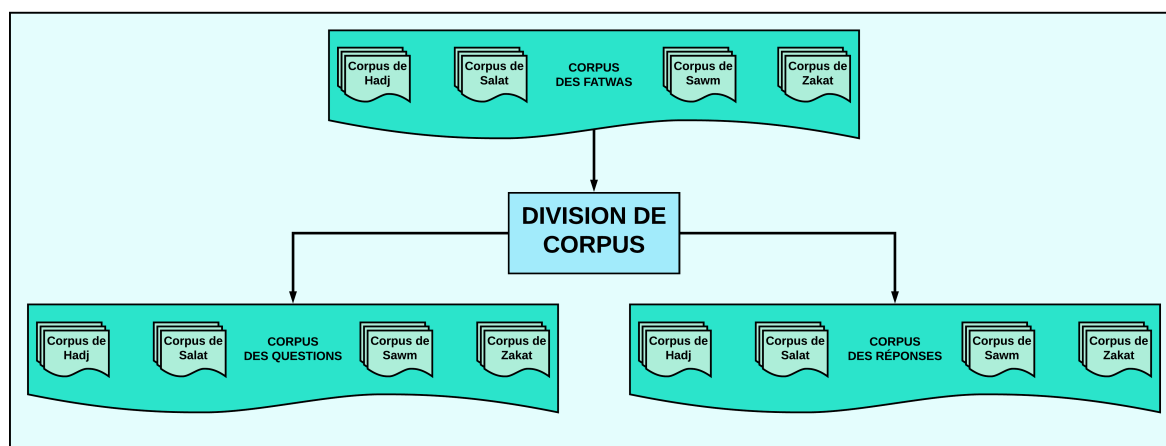


FIGURE 3.2 – Préparation de corpus.

Le but de cette étape est de générer depuis le corpus des fatwas, deux corpus contenant les questions et les réponses, la figure 2 démontre le processus.

Cette partie a été implémenté en Java, tandis que le reste du projet a été développé en python. L'approche utilisé était naïve, mais efficace a un certain degré, en cherchant des indices (expressions, points d'interrogation, ...) qui signifie la fin de la question, et le début de la réponse, ensuite créer deux nouveaux documents dans des chemins séparés contenant la question et la réponse.

3.2.2 Prétraitement

Après la division des corpus, on est besoin de prétraiter le corpus des questions avant l'alimenter aux modèles de routage et de similarité, en supprimant les mots et expressions inutiles dans les questions, ce type de mots sont appelés les mots vides ou bien les 'stopwords', ils donnent un sens au pour le lecteur (humain) sous un aspect lexical et cognitif, mais ils ne servent qu'un bruit pour les modèles d'apprentissage automatique.

Le processus de prétraitement vise à donner des tokens uniques aux mots de la même racine (stemming), et minimiser le nombre des mots dans le dictionnaire de corpus, qui diminue leur dimension, d'où diminuer la complexité de la tâche. Il est si important de le faire correctement car cela influence grandement la prochaine étape.

La bibliothèque NLTK (kit d'outils de langage naturel) comprend une petite liste contenant près de 800 mots vides arabes, ce n'était pas suffisant et nous avons donc obtenu une liste plus longue, combinés avec notre liste spéciale de mots vides, nous avons atteint 1897 mots uniques. On a aussi utilisé l'outil ISRIStemmer pour la racinisation des mots.

L'algorithme de Stemming Arabe de l'Institut de Recherche en Sciences de l'Information (ISRI) a été développé à l'université du Nevada, aux États-Unis, en 2005. Il s'agit d'un radical arabe sans dictionnaire racine. Il n'utilise pas de dictionnaire racine. De plus, si une racine n'est pas trouvée, le stemmer ISRI a renvoyé une forme normalisée, plutôt que de renvoyer le mot d'origine non modifié. [28]

Le corpus contenait de nombreux documents vides, hors sujet et inéligibles qui ont été éliminés dans cette phase, pour cette raison le nombre de documents de formation et de test était un peu ambigu, mais pour donner un pourcentage approximatif :

Les modèles de classification ont été formés avec 67% du corpus et testés avec le reste, tandis que les modèles de similitude ont été formés avec 70% du corpus et testés avec le reste.

3.2.3 Entraînement des modèles

La dernière étape de formation est l'entraînement des modèles, tandis que le modèle de routage s'entraîne avec tous les classes des fatwas pour classifier les requêtes, les modèles de similarité s'entraîne uniquement avec une partie de corpus qui représente sa spécialité, la partie suivante va exposer les architectures de formations des modèles.

3.2.3.1 Routage des requêtes

Les modèles de routage de notre système sont en fait des modèles de classification multi classes à étiquette unique pour classer les questions des utilisateurs, et sert à un moyen de déterminer la classe de fatwa adéquate qui devrait contenir la réponse à cette question, d'où les bons modèles de similarité continueront le processus de fonctionnement de système.

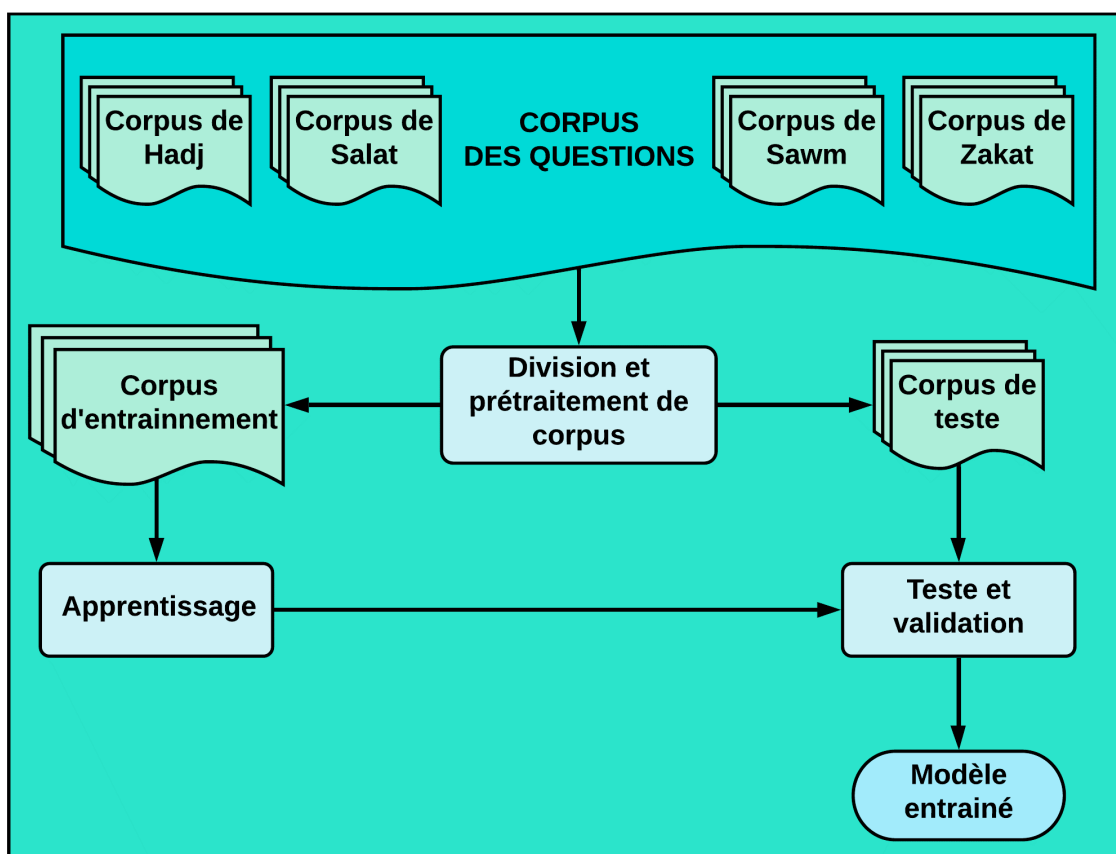


FIGURE 3.3 – Architecture de formation des modèles de routage.

3.2.3.1.1 Équilibrage des classes Les tâches de classification sont fortement influencées par le nombre de données de formation par classe comme indiqué précédemment

dans le deuxième chapitre, un ensemble de données déséquilibré amènera le classificateur à souvent prédire les documents des classes minoritaires comme étant ceux de la classe majoritaire, ce qui réduit la précision.

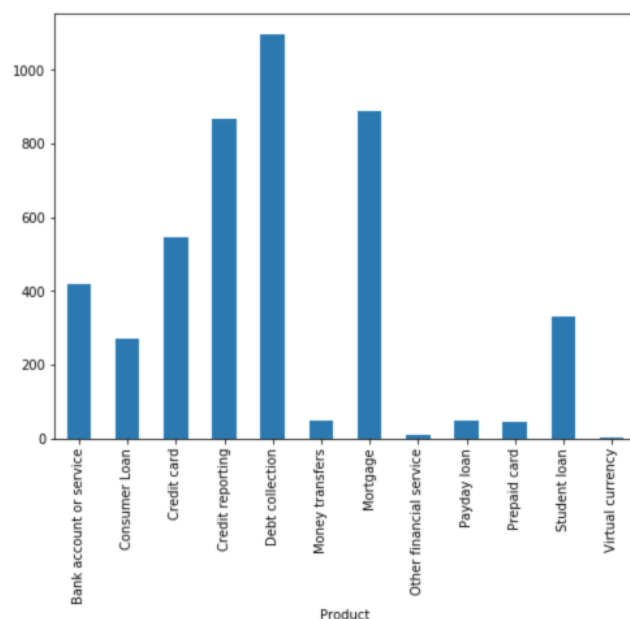


FIGURE 3.4 – Exemple d'un corpus déséquilibré. [24]

L'exclusion de certains documents est la solution à ce problème, on a borné le nombre des documents à 300 par classe qui réduit l'écart entre eux, le graph à barres suivant montre la distribution des classes utilisée pour les modèles de classification.

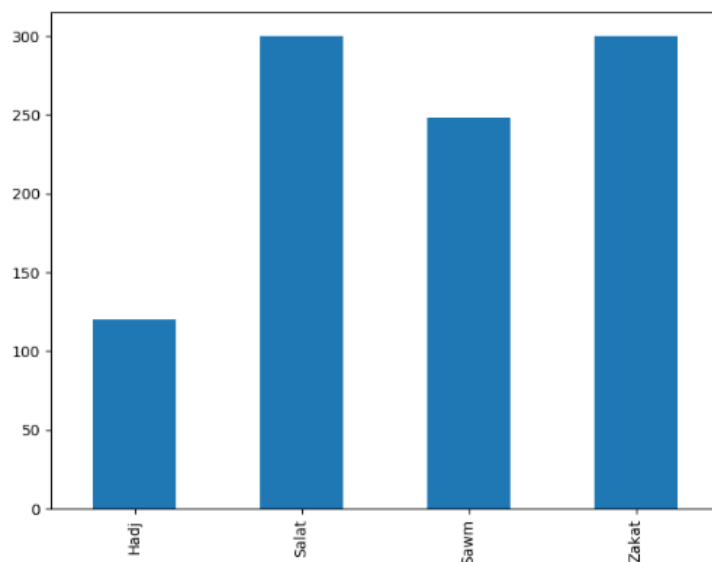


FIGURE 3.5 – Distribution de données par classe.

Grace à la bibliothèque Scikit-learn et pour assurer les meilleurs résultats on a testé les performances de 4 modèles différents, y compris la régression logistique, le bayes naïf multinomiaux, le classificateur de forêt aléatoire et de vecteur de support linéaire.

3.2.3.1.2 Paramètres des algorithmes de classification Les paramètres généraux comprennent les outils de représentation des données, pour cela on a utilisé le vectoriseur TfidfVectorizer avec les paramètres suivants :

Paramètres	Valeurs
Représentation des données	TF-IDF (TfidfVectorizer)
(sublinear_f) Applique une mise à l'échelle TF linéaire	Vrai
(min_f) ignorer les termes dont la fréquence de document est strictement inférieure au seuil donné .	5
(norm) Chaque ligne de sortie aura une norme d'unité : l2 ou l1	l2
(ngram_range) Limites inférieure et supérieure de la plage de valeurs n pour différents n grammes à extraire.	(1, 2)
Données d'entraînement	67%
Données de teste	33%
Couche d'entrée	968 (features = labels)
Couche de sortie	4

TABLE 3.1 – Paramètres de vectoriseur des modèles de classification.

Ce vectoriseur apprend tout le vocabulaire de corpus et compte les fréquences des mots, ensuite il inverse ce dernier pour donner une importance plus élevée aux mots plus rare dans le corpus, après le prétraitement des questions, elles sont représentées comme montré ci-dessous :

Question	ما حكم من اعتمر او حج من غير لباس الاحرام				
Question prétraité	حكم	عمر	حج	لبس	حرم
Identité de chaque mot	(0.625)	(0.518)	(0.204)	(0.194)	(0.184)
Valeur de vectoriseur	0.43	0.42	0.26	0.41	0.63

TABLE 3.2: Représentation d'une question avec le vectoriseur TF-IDF.

Les tableaux suivants montrent les paramètres utilisés pour entraîner les modèles de routage, seulement les deux premiers paramètres de classificateur de forêt aléatoire ne sont pas dans ses valeurs par défaut parmi tous les paramètres cités.

Paramètres	Valeurs
(Penalty) norme utilisée dans la pénalisation	l2
(loss) Spécifie la fonction de perte.	squared_log
(dual) Sélectionne l'algorithme pour résoudre le problème d'optimisation double ou primaire.	Vrai
(tol) Tolérance pour les critères d'arrêt.	0.0001
(C) Paramètre de régularisation.	1.0
(multi_class) Détermine la stratégie multi-classes	one-vs-rest (forme des classificateurs un contre rest)

(fit_intercept) Indique s'il faut calculer l'ordonnée à l'origine pour ce modèle.	Vrai
(intercept_scaling) une caractéristique synthétique à valeur constante	1
(class_weight) Poids associés aux classes	Équilibré
(verbose) Activer la sortie détaillée.	0
(random_state) Contrôle la génération de nombres pseudo aléatoires pour mélanger les données.	0
(max_iter) Le nombre maximal d'itérations à exécuter.	1000
(Stratify) conserver la même proportion de classes dans le train et les ensembles de tests selon quoi	Labels (classe)

TABLE 3.3 – Paramètres d'algorithme de Support de vecteur linéaire. [35]

Paramètres	Valeurs
(Solver) Algorithme à utiliser dans le problème d'optimisation	lbfgs
(Pénalité) norme utilisée dans la pénalisation	l2
(dual) Formulation double ou primale.	Faux
(tol) Tolérance pour les critères d'arrêt.	0.0001
(C) Inverse de la force de régularisation	1.0
(fit_intercept) Spécifie si une constante doit être ajoutée à la fonction de décision.	Vrai
(class_weight) Poids associés aux classes	Équilibré
(random_state) Instance pour mélanger les données.	0
(max_iter) Nombre maximum d'itérations prises pour que les solveurs convergent.	100
(multi_class) la perte minimisée	Auto (multinomiale)
(verbose) pour la verbosité.	0
(warm_start) réutiliser la solution de l'appel précédent pour l'adapter comme initialisation	Faux
(n_jobs) Nombre de cœurs de processeur utilisés lors de la parallélisation sur des classes	1
(l1_ratio) Le paramètre de mélange Elastic-Net	Aucun

TABLE 3.4 – Paramètres d'algorithme de régression logistique. [33]

Paramètres	Valeurs
(Alpha) Paramètre de lissage additif	1.0
(fit_rior) Apprendre les probabilités a priori de la classe.	Vrai
(class_rior) Probabilités antérieures des classes.	Aucun

TABLE 3.5 – Paramètres d’algorithme Naïf bayes multinomiaux. [17]

Paramètres	Valeurs
(N_stimators) Le nombre d’arbres dans la forêt.	200
(Max_epth) La profondeur maximale de l’arbre.	3
(criterion) La fonction pour mesurer la qualité d’un split.	Gini
(min_amples_plt) Le nombre minimum d’échantillons requis pour diviser un nœud interne	2
(min_amples_eaf) Le nombre minimum d’échantillons requis pour être à un nœud de feuille.	1
(min_eight_raction_eaf) La fraction pondérée minimale du total des poids requis pour être à un nœud de feuille.	0.0
(max_eatures) Le nombre de fonctionnalités à considérer lors de la recherche de la meilleure répartition.	Auto (n_eatures)
(max_eaf_odes) Faites pousser des arbres avec cet attribut de la meilleure façon. Les meilleurs nœuds sont définis comme une réduction relative de l’impureté.	Aucun
(min_mpurity_ecrease) Un nœud sera divisé si ce fractionnement induit une diminution de l’impureté supérieure ou égale à cette valeur.	0.0
(min_mpurity_plt) Seuil pour l’arrêt précoce de la croissance des arbres.	Aucune
(bootstrap) Si des échantillons de bootstrap sont utilisés lors de la construction d’arbres.	Vrai
(oob_core) Utiliser ou non des échantillons hors du sac pour estimer la précision de la généralisation.	Faux
(n_obs) Nombre de travaux à exécuter en parallèle.	Aucun (1)

(Random_tate) Contrôle à la fois le caractère aléatoire du bootstrap des échantillons utilisés lors de la construction d'arbres et l'échantillonnage des fonctionnalités à prendre en compte lors de la recherche du meilleur fractionnement à chaque nœud	0
(verbose) Contrôle la verbosité lors de l'ajustement et de la prévision.	0
(warm_tart) s'il faut réutiliser la solution de l'appel précédent pour ajuster et ajouter plus d'estimateurs à l'ensemble.	Faux
(class_eight) Poids associés aux classes	Aucun
(ccp_lpha) Paramètre de complexité utilisé pour l'élagage à coût-complexité minimal.	0.0
(max_amples) Si le bootstrap est vrai, le nombre d'échantillons à tirer de X pour former chaque estimateur de base.	Aucun

TABLE 3.6 – Paramètres d'algorithme de classificateur de forêt aléatoire. [32]

3.2.3.2 Calcul de similarité distributionnelle

L'utilisation des modèles de similarité dans notre système vise à trouver les questions les plus similaires à la question posée, ces modèles dépendent totalement des questions de corpus entraîné, et ne génère aucun type de fatwas par eux-mêmes, ils ne prennent pas le rôle des muftis mais plutôt facilitent et automatisent leur travail en assurent la livraison des fatwas au grande publique.

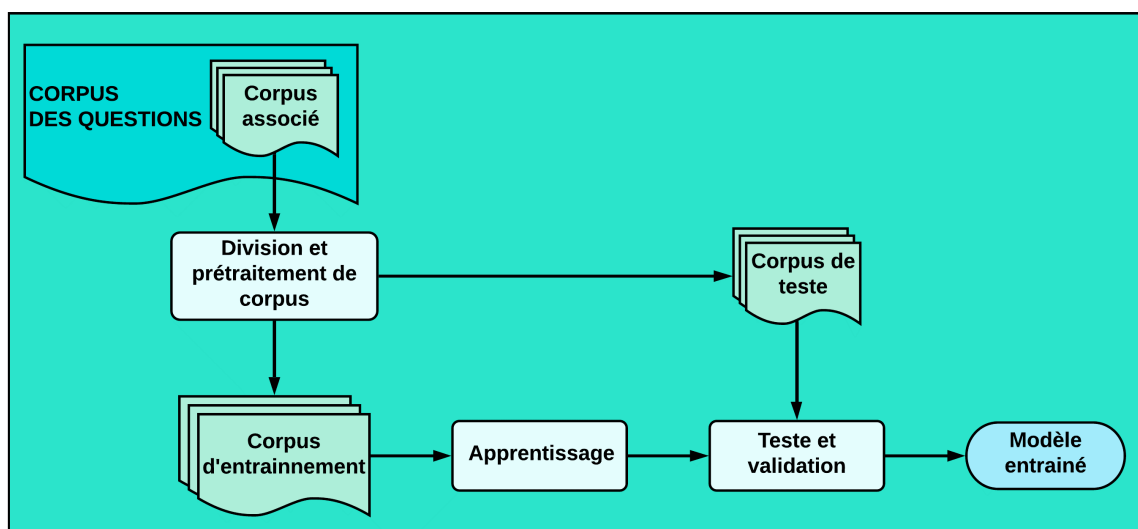


FIGURE 3.6 – Architecture de formation des modèles de similarité.

La prochaine étape est d'extraire les questions similaires à la question posée, cette fois

nous ne devons exclure aucun document de l'ensemble de données car chaque classe va posséder ses propres modèles.

Cette méthodologie augmente le nombre des modèles mais a l'avantage de donner des meilleurs résultats, en excluant les documents non liés au sujet dans le classement des documents similaires à la requête, ce qui est faux à bien des égards.

Grâce aux nombreux modèles de Gensim, on a pu tester 4 modèles de similarité différents, y compris TF-IDF, L'indexation sémantique latente, Allocation Dirichlet Latente et Doc2Vec.

3.2.3.2.1 Paramètres des algorithmes de similarité distributionnelle Les paramètres généraux comprennent les outils de représentation des données, pour cela on a utilisé le modèle Bow avec les paramètres suivants :

Paramètres	Valeurs
Taille de vocabulaire	Taille de dictionnaire
Représentation des données	BoW (Sac des mots)
Données d'entraînement	70%
Données de teste	30%
Couche d'entrée	Selon la classe (voir la taille des dictionnaires)
Couche de sortie	Selon le corpus (70% de taille totale)

TABLE 3.7: Paramètres généraux des modèles de similarité.

Le tableau suivant montre comment le modèle Bow représente les questions :

Question	ما حكم من اعتمر او حج من غير لباس الاحرام؟				
Question prétraité	لبس	عمر	حكم	حرم	حج
Identité de chaque mot dans le corpus	197	71	60	47	12
Occurrence de chaque mot dans la question	1	1	1	1	1

TABLE 3.8: Représentation d'une question avec le modèle de sac des mots.

Remarquant que le modèle de sac des mots ne respecte pas l'ordre des mots dans la question comme le vectoriseur TF-IDF, et les trie par ordre alphabétique. Le tableau suivant représente les tailles des dictionnaires dans chaque classe de corpus.

Classes	Taille des dictionnaires
Hadj	554
Salat	1347
Sawm	960
Zakat	1444
Total	4305

TABLE 3.9: Taille des dictionnaires des modèles de similarité.

Les tableaux suivants sont les paramètres utilisés pour entraîner les modèles de similarité, seulement les 7 premiers paramètres des modèles Doc2Vec, et une de LDA ne sont pas dans ses valeurs par défaut entre tous les paramètres cités, qui est les passes du modèle LDA (200 au lieu de 1).

Paramètres	Valeurs
(corpus) Corpus d'entrée	1 pour chaque modèle (total=4)
(id2word) Jeton de mappage utilisé pour convertir les données d'entrée au format sac de mots.	Aucun
(dictionary) - utilisé pour construire directement le mappage de fréquence de document inverse	Selon la classe (voir le tableau des dictionnaires)
(global) Fonction pour la pondération globale	df2idf
(wlocals) Fonction de pondération locale	identity
(smartirs) un schéma mnémorique pour désigner les variantes de pondération tf-idf dans le modèle d'espace vectoriel.	bic (voir les 3 prochaines paramètres)
Pondération de fréquence de terme	brut
Pondération de fréquence du document	idf
Normalisation du document	cosinus
(pivot) Normalisation de la longueur du document pivoté	Déterminé automatiquement à partir des propriétés du corpus ou du dictionnaire.
(slope) effet de la normalisation de la longueur du document pivoté.	0,25

TABLE 3.10 – Paramètres d'algorithme tf-idf. [23]

Paramètres	Valeurs
(corpus) Flux de vecteurs de documents ou matrice clairsemée.	1 pour chaque modèle
(num_opics) Nombre de facteurs demandés (dimensions latentes)	200
(id2word) Mappage ID vers mot.	Aucun
(chunksize) Nombre de documents à utiliser dans chaque bloc de formation.	20000
(decay) Poids des observations existantes par rapport aux nouvelles.	1.0
(distributed). mode distribué (exécution parallèle sur plusieurs machines)	Faux
(onepass) Si l'algorithme à un passage doit être utilisé pour la formation.	Vrai
(power_ters) Nombre d'étapes d'itération de puissance à utiliser.	2
(extra_amples) Des échantillons supplémentaires à utiliser en plus du rang k.	100
(dtype) Applique un type pour les éléments de la matrice décomposée.	numpy.float64

TABLE 3.11 – Paramètres d'algorithme LSI. [22]

Paramètres	Valeurs
(corpus) Flux de vecteurs de documents ou matrice clairsemée	1 pour chaque modèle
(num_opics) Le nombre de sujets latents demandés à extraire du corpus de formation.	100
(id2word) Mappage des ID de mot aux mots. utilisé pour déterminer la taille du vocabulaire	Aucun
(distributed) si le calcul distribuée doit être utilisée pour accélérer la formation	Faux
(chunksize) Nombre de documents à utiliser dans chaque bloc de formation.	2000
(passes) Nombre de passes à travers le corpus pendant l'entraînement.	200
(update_very) Nombre de documents à parcourir pour chaque mise à jour.	1
(alpha) Apprend un a priori asymétrique à partir du corpus	Symétrique
(eta) Croyance a priori sur la probabilité des mots	Aucun
(decay) quel pourcentage de la valeur lambda précédente est oublié lorsque chaque nouveau document est examiné.	0.5
(offset) contrôle combien nous allons ralentir les premières étapes des premières itérations.	1.0
(eval_very) La perplexité des journaux est estimée à chaque fois que de nombreuses mises à jour.	10
(iterations) Nombre maximum d'itérations à travers le corpus lors de la déduction de la distribution des rubriques d'un corpus.	50
(gamma_hreshold) Variation minimale de la valeur des paramètres gamma pour continuer l'itération.	0.001
(minimum_probability) Les sujets avec une probabilité inférieure à ce seuil seront filtrés.	0.01
(random_state) soit un objet randomState, soit une graine pour en générer un. Utile pour la reproductibilité.	Aucun
(minimum_probability) cela représente une borne inférieure du terme probabilités.	0.01
(per_order_opics) s'il faut calculer une liste de sujets pour chaque mot	Faux
(callbacks) Rappels métriques pour consigner et visualiser les métriques d'évaluation du modèle pendant la formation.	Aucun

(dtype) Type de données à utiliser lors des calculs à l'intérieur du modèle. Toutes les entrées sont également converties.	numpy.float32
--	---------------

TABLE 3.12 – Paramètres de modèle LDA. [21]

Paramètres	Valeurs
(vector_ize) Dimensionnalité des vecteurs caractéristiques	50
(window) Distance maximale entre le mot actuel et le mot prédit dans une phrase.	10
(min_ount) Ignore tous les mots dont la fréquence totale est inférieure à celle-ci.	1
(workers) Utilisez ce nombre de threads de travail pour former le modèle	8
(alpha) Le taux d'apprentissage initial.	0.025
(min_lpha) Le taux d'apprentissage diminuera linéairement vers cet attribut à mesure que la formation progresse.	0.015
(epochs) Nombre d'itérations sur le corpus.	100
(documents) itérable de la liste des documents balisés	Aucun
(corpus_ile) Chemin vers un fichier corpus au format LineSentence.	1 pour chaque modèle
(dm) Définit l'algorithme d'apprentissage.	(0) sac de mots distribué (PV-DBOW).
(dbow_ords) s'il faut former des vecteurs de mots simultanément avec la formation DBOW doc-vecteur	(0) ne forme que des doc-vecteurs
(dm_oncat) soit utilisez la concaténation des vecteurs de contexten soit la somme / la moyenne	(0) (la somme / la moyenne)
(dm_ag_ount) Nombre constant attendu de balises de document par document, lors de l'utilisation du mode dm_oncat.	1
(docvecs) liste des vecteurs à clé Doc2Vec Représentations vectorielles des documents dans le corpus.	Aucun
(trim_ule) Règle d'ajustement du vocabulaire	Aucun (min_ount sera utilisé)
(callbacks) Liste des rappels qui doivent être exécutés / exécutés à des étapes spécifiques pendant la formation.	Vide

TABLE 3.13 – Paramètres d'algorithme du modèle doc2vec. [20]

3.3 Interface graphique

L'interface de l'application a été réalisé à l'aide de Qt Designer, cet outil est basé sur glisser et déposer, la génération du code python nécessite l'exécution d'une seule commande, ces facteurs ont facilité la création de l'interface.

La fenêtre principale est composée de 3 parties principales :

- La première partie s'occupe à formuler les requêtes d'utilisateur, et c'est fait en 3 manières différents (écrire la question manuellement, parcourir les documents ou aléatoirement).
- La deuxième partie s'occupe de la classification des requêtes en leur sujet, avec la capacité de changer la classe en cas de fausse prédiction.
- La dernière partie comporte l'affichage des résultats, en forme d'un tableau pour mieux ordonner les réponses.

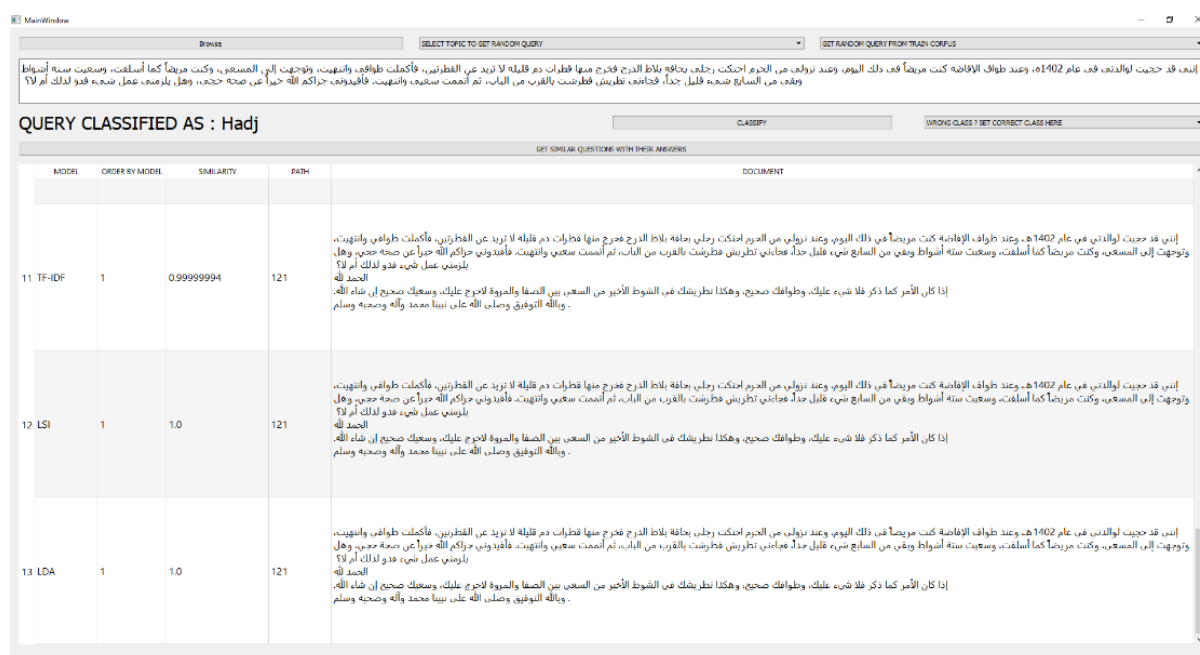


FIGURE 3.7 – Interface graphique représentant le résultat d'une requête aléatoire.

Conclusion

Dans ce chapitre, nous avons présenté les deux architectures de formation et de fonctionnement de notre système, le fonctionnement se base sur deux types de modèles en apprentissage automatique, les modèles de classification (routage) et les modèles de similarité. Ce chapitre ainsi introduit notre architecture proposée pour la réalisation d'un système de question réponse de fatwas islamiques, et une implémentation d'une interface graphique simple pour visualiser le fonctionnement du système.

Chapitre 4

Expérimentations et déploiement

Introduction

Afin d'évaluer la performance de l'approche proposée, un ensemble de tests ont été menés. L'évaluation concerne les modèles proposés dans les deux aspects de notre système QA, le routage et la recherche de similarité.

Les évaluations des modèles de routage ont été faites en utilisant les métriques rappel, précision, f-mesure et Justesse, avec un corpus de 320 questions divisés en 4 classes, notamment 'Hadj' avec 40 fatwas, 'Salat' avec 99 fatwas, 'Sawm' avec 82 fatwas, et 'Zakat' avec 99 fatwas. Cependant, d'autres méthodes d'évaluation ont été utilisées pour évaluer les modèles de similarité distributionnelle.

Les expérimentations des modèles de similarité distributionnelle ont été menées sur un corpus de Fatwas constitué de 368 questions distribuées en quatre catégories, notamment 'Hadj' avec 33 fatwas, 'Salat' avec 167 fatwas, 'Sawm' avec 74 fatwas, et 'Zakat' avec 94 fatwas.

4.1 Métriques d'évaluation

Il existe de nombreuses métriques qui peuvent être utilisées pour mesurer les performances d'un classificateur. Pour les tâches de classification, les termes vrais positifs, vrais négatifs, faux positifs et faux négatifs comparent les résultats du classificateur testé avec des jugements externes fiables. Les termes positifs et négatifs se réfèrent à la prédiction du classifieur, et les termes vrai et faux se réfèrent à savoir si cette prédiction correspond au jugement externe.

4.1.1 Matrice de confusion

Une matrice de confusion également connue sous le nom de matrice d'erreur, est un type spécial de tableau de contingence, avec deux dimensions actuelle et prévues, et des ensembles de classes identiques dans les deux dimensions, elle permet de résumer et de visualiser les performances d'un algorithme de classification, en comparaison avec une autre classification de référence. [8]

Un classificateur de classe N formé produit un résultat qui peut être organisé en une matrice $N \times N$. La matrice de confusion d'un classificateur binaire ressemblerait à celle présentée ci-dessous :

		Classe réelle	
		Positif	Négatif
Classe prédite	Positif	Vrai Positif	Faux Positif
	Négatif	Faux Négatif	Vrai Négatif

TABLE 4.1: Matrice de confusion. [8]

En plus de matrice de confusion, il existe également des mesures uniques qui donnent un numéro unique pour évaluer le test.

4.1.2 La précision

La précision (également appelée valeur prédictive positive) est la fraction des instances pertinentes parmi les instances récupérées. Dans une tâche de classification, la précision d'une classe est le nombre de vrais positifs divisé par le nombre total d'éléments étiquetés comme appartenant à la classe positive. [29]

Dans une tâche de classification, un score de précision de 1,0 pour une classe C signifie que chaque élément étiqueté comme appartenant à la classe C appartient bien à la classe C. [29]

4.1.3 Le rappel

Le rappel, la sensibilité ou le vrai taux positif est la proportion de données qui se sont révélées positives et sont positives (vrai positif) de toutes les données qui appartiennent réellement à la classe positive. [29]

Dans une tâche de classification, un rappel de 1,0 signifie que chaque article de la classe C a été étiqueté comme appartenant à la classe C. [29]

La précision peut être considérée comme une mesure de justesse ou de qualité, tandis que le rappel est une mesure d'exhaustivité ou de quantité. Souvent, il existe une relation inverse entre précision et rappel, où il est possible d'augmenter l'un au prix de réduire l'autre. [29]

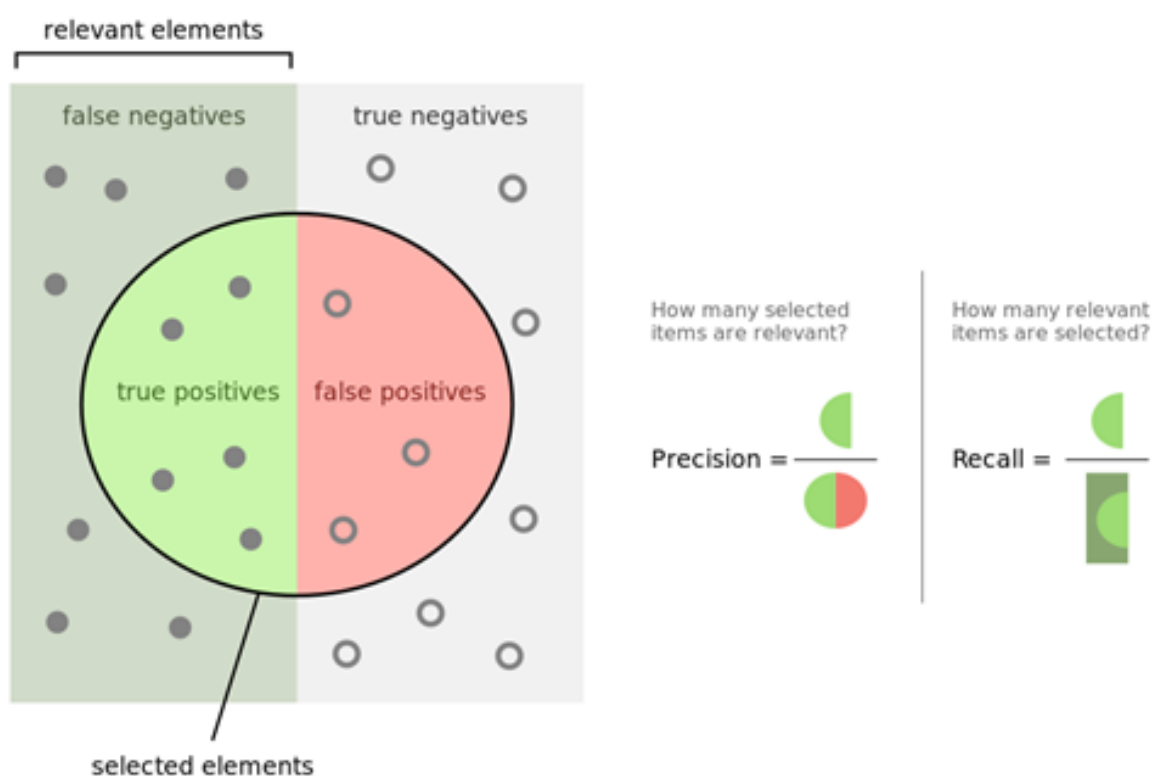


FIGURE 4.1 – Précision et rappel. [29]

4.1.4 La justesse

La justesse est utilisée comme une mesure statistique de la mesure dans laquelle un test de classification identifie ou exclut correctement une certaine classe. C'est-à-dire que la justesse est la proportion de prédictions correctes (à la fois vrais positifs et vrais négatifs) parmi le nombre total de cas examinés. [1]

Une faible justesse indique une différence entre un résultat et une valeur souhaitée. [1] La justesse donnera des résultats trompeurs si l'ensemble de données est déséquilibré, c'est-à-dire lorsque le nombre d'observations dans différentes classes varie considérablement. [13]

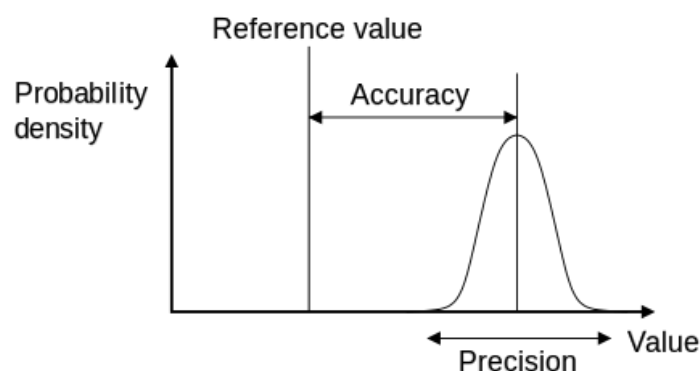


FIGURE 4.2 – Précision et justesse. [1]

4.1.5 Score F-1

Habituellement, les scores de précision et de rappel ne sont pas discutés isolément. Au lieu de cela, soit les valeurs d'une mesure sont comparées pour un niveau fixe à l'autre mesure, soit les deux sont combinées en une seule mesure. [29]

Un score F est une combinaison de la précision et du rappel, fournissant un score unique. Le score F traditionnel ou équilibré (score F1) est la moyenne harmonique de la précision et du rappel, un score F1 atteint sa meilleure valeur (1,0) signifie une précision et un rappel parfaits. [15]

Le score F1 est plus peu fiable que l'exactitude lorsque l'ensemble de données est déséquilibré, il produit des résultats trompeurs. [13]

Sklearn offre une variété de paramètres dans leurs implémentations, l'un d'eux a été utilisé dans l'évaluation. Ce paramètre est requis pour les cibles multi classes / multi étiquettes. Cela détermine le type de moyennage effectué sur les données. [34]

4.1.5.1 Macro

Calculez les métriques pour chaque étiquette et trouvez leur moyenne non pondérée. Cela ne tient pas compte du déséquilibre des étiquettes. [34]

4.1.5.2 La macro F1

Calcule la F1 séparée par classe mais n'utilise pas de poids pour l'agrégation, ce qui entraîne une pénalisation plus importante lorsque le modèle n'est pas suffisamment efficace avec les classes minoritaires. [19]

4.1.5.3 Pondéré

Le score F1 pondéré calcule le score F1 pour chaque classe indépendamment, mais lorsqu'il les additionne, il utilise un poids qui dépend du nombre d'étiquettes vraies de chaque classe. [19]

Cela modifie la macro pour tenir compte du déséquilibre des étiquettes; il peut en résulter un score F qui n'est pas entre la précision et le rappel.[34]

4.2 Evaluation des algorithmes

4.2.1 Modèles de classification

Les modèles de classification ont été testé avec les mêmes données en utilisant la validation croisée à 5 itérations, après plusieurs tests, le modèle de classificateur de vecteur de support linéaire s'est avéré être le meilleur classificateur avec une justesse moyenne de 95 % suivi par la régression logistique qui avait 94 % et Bayes naïfs multinomiaux avec une justesse de 92%, le Classificateur de forêt aléatoire avait la plus faible justesse de 80 %.

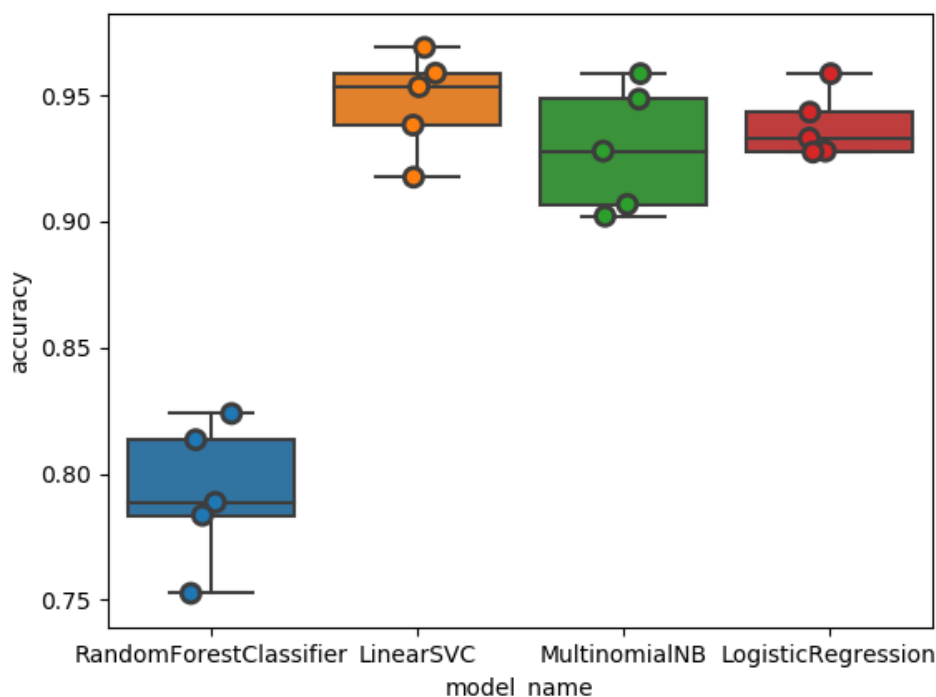


FIGURE 4.3 – Les justesses des modèles de classification.

Modèle \ Justesse	Justesse	Justesse moyenne
Classificateur de vecteur de support linéaire	0.95	0.945275
Régression logistique	0.95	0.935981
Naïf bayes multinomiaux	0.92	0.921505
Classificateur de forêt aléatoire	0.80	0.797527

TABLE 4.2: Comparaison des justesses par modèle.

Les matrices de confusion résument beaucoup d'informations dans un tableau des axes actuelle et prévues. Relativement à chaque classe, les numéros dedans représente :

- Colonne : faux positive (toutes les lignes d'une colonne à part le diagonal).
- Ligne : faux négative (toutes les colonnes d'une ligne à part le diagonal).
- Diagonal : vrai positive / vrai négative.

Cette matrice montre l'efficacité de l'algorithme, la seule faille remarquable est de 'Sawm' qui a une fausse négative de 8, associé au 'Salat', tandis que les autres failles sont entre 0, 1 et 2, ces résultats se traduisent vers un recul de précision de 'Salat' et de rappel de 'Sawm'.

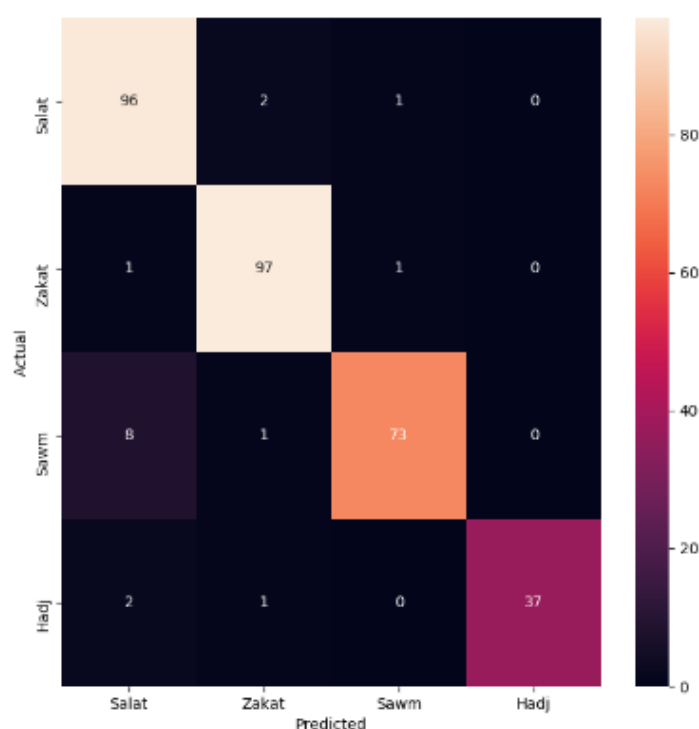


FIGURE 4.4 – Matrice de confusion de Classificateur de vecteur de support linéaire.

Classe \ Métrique	Précision	Rappel	F1-score	Support
Hadj	1.00	0.93	0.96	40
Salat	0.90	0.97	0.93	99
Sawm	0.97	0.89	0.93	82
Zakat	0.96	0.98	0.97	99
Macro moyenne	0.96	0.94	0.95	320
Moyenne pondérée	0.95	0.95	0.95	320

TABLE 4.3: Métriques d'évaluation de Classificateur de vecteur de support linéaire.

Les matrices de classificateur de vecteur de support linéaire et celle de régression logistique sont presque identiques, et nécessite un examen approfondi pour détecter leur petite différence.

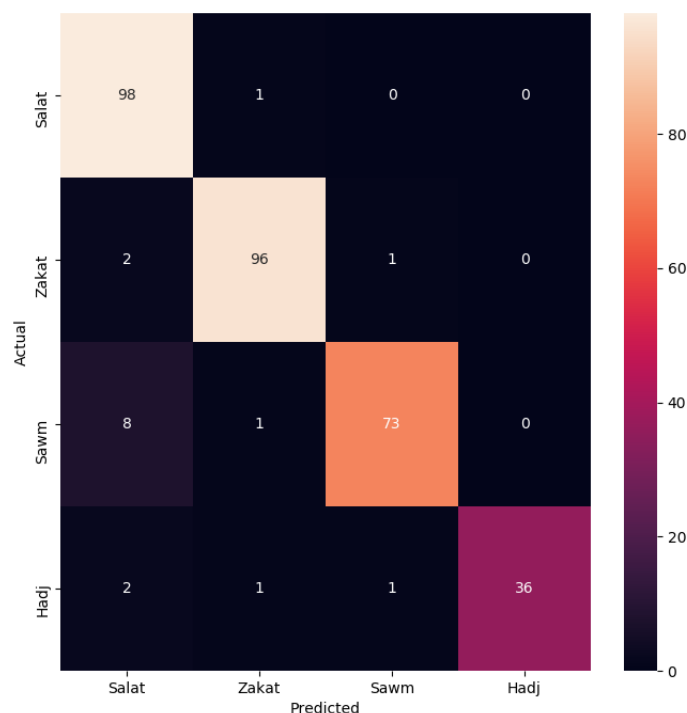


FIGURE 4.5 – Matrice de confusion de Régression logistique.

Métrique \ Classe	Précision	Rappel	F1-score	Support
Hadj	1.00	0.90	0.95	40
Salat	0.89	0.99	0.94	99
Sawm	0.97	0.89	0.93	82
Zakat	0.97	0.97	0.97	99
Macro moyenne	0.96	0.94	0.95	320
Moyenne pondérée	0.95	0.95	0.95	320

TABLE 4.4: métriques d'évaluation de Régression logistique.

La matrice de confusion de bayes naïf multinomiaux montre un peu de recul par rapport aux deux premiers algorithmes avec trois failles notables de 'Salat' dont deux fausses positives qui appartiennent en fait à 'Hadj' et 'Sawm', et une fausse négative prédit d'être de 'Zakat'.

Ces résultats sont traduits vers un recul des performances de l'algorithme par rapport aux deux premiers algorithmes.

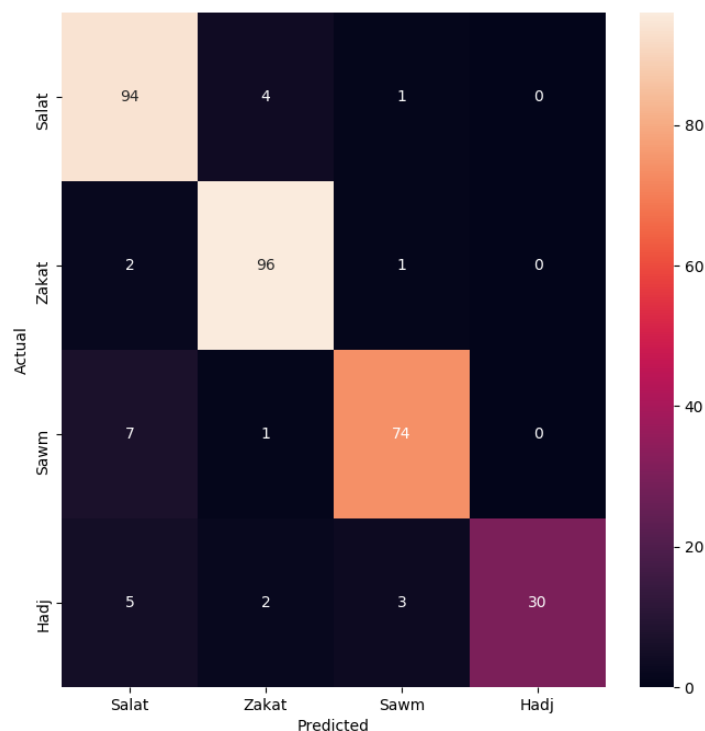


FIGURE 4.6 – Matrice de confusion de Naïf bayes multinomiaux.

Métrique \ Classe	Précision	Rappel	F1-score	Support
Hadj	1.00	0.75	0.86	40
Salat	0.87	0.95	0.91	99
Sawm	0.94	0.90	0.92	82
Zakat	0.93	0.97	0.95	99
Macro moyenne	0.93	0.89	0.91	320
Moyenne pondérée	0.92	0.92	0.92	320

TABLE 4.5: Métriques d'évaluation de Naïf bayes multinomiaux.

La matrice de confusion de classificateur de forêt aléatoire expose sa mauvaise compétence avec les autres algorithmes, les erreurs sont clairement visibles, les questions de 'Hadj' et 'Sawm' ont été mal prédit d'être des questions de 'Salat' en ce qui représente une fausse négative pour les premiers et une fausse positive pour le dernier.

Tandis qu'il a bien performé dans les classes majoritaires ('Salat', 'Zakat'), il a mal prédit les classes minoritaires ('Hadj', 'Sawm'), cette faille a été aussi exposé par le rappel des classes minoritaires.

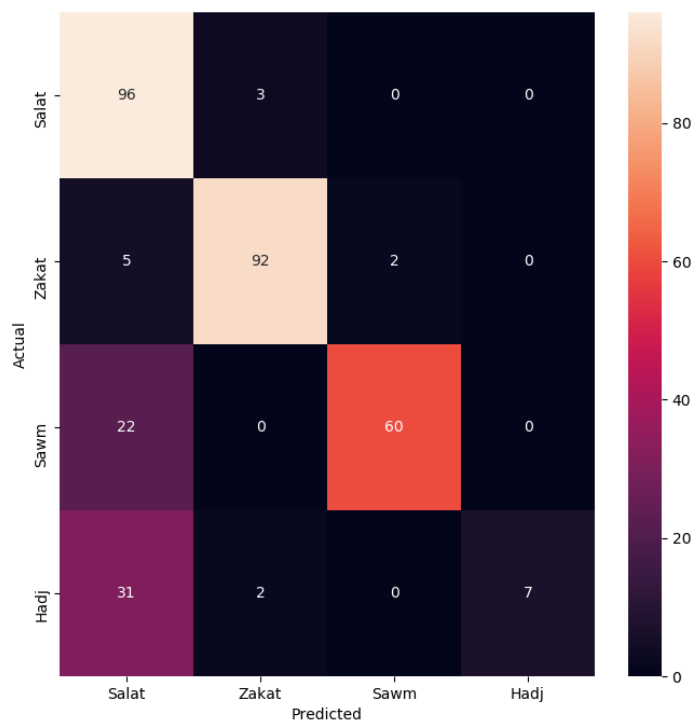


FIGURE 4.7 – Matrice de confusion de Classificateur de forêt aléatoire.

Métrique \ Classe	Précision	Rappel	F1-score	Support
Hadj	1.00	0.17	0.30	40
Salat	0.62	0.97	0.76	99
Sawm	0.97	0.73	0.83	82
Zakat	0.95	0.93	0.94	99
Macro moyenne	0.88	0.70	0.71	320
Moyenne pondérée	0.86	0.80	0.78	320

TABLE 4.6: Métriques d'évaluation de modèle classificateur de forêt aléatoire.

Tandis que les performances de dernier algorithme sont un peu en dessous du niveau acceptable, les autres algorithmes ont bien performé, comme le prouvent les mesures d'évaluation que nous avons utilisées.

Les performances sont excellentes étant donné la longueur de notre corpus, et bien qu'ils ne s'agissent pas de modèles d'apprentissage profond, cela prouve que même les modèles de classification conventionnelles peuvent atteindre des résultats satisfaisants avec le prétraitement et les paramètres appropriés. L'amélioration de cette étape au cours de l'implémentation a augmenté la justesse de 3 % jusqu'à près de 7 % pour certains modèles de classification.

Néanmoins, les performances des modèles ont montré que nous n'avons besoin que d'un modèle pour cette tâche (le modèle le plus performant) et qui est dans notre cas le classificateur de vecteur de support linéaire.

4.2.2 Modèles de similarité

On a entraîné les modèles de similarité avec 70 % des corpus et gardé le reste pour les tester, les résultats suivants montrent la précision de reconnaissance des questions de formation par les modèles de similarité. Une précision parfaite est la première position, tandis que les autres positions signifient un manque de précision.

Modèle \ Prédiction	Position correcte (Première occurrence)	Position incorrecte		Total
		2ème	3ème	
LDA	64	19	5	88
LSI	86	2	0	
TF-IDF DOC2VEC				

TABLE 4.7: Comparaison de précision des modèles de similarité (classe de Hadj).

Modèle \ Prédiction	Position correcte (Première occurrence)	Position incorrecte					Total
		2ème	3ème	4ème	5ème	6ème	
LDA	256	67	42	16	5	4	390
LSI	388	2	0				
TF-IDF							
DOC2VEC							

TABLE 4.8: Comparaison de précision des modèles de similarité (classe de Salat).

Modèle \ Prédiction	Position correcte (Première occurrence)	Position incorrecte					Total
		2ème	3ème	4ème	5ème	6ème	
LDA	103	42	19	6	3	1	174
LSI	171	3	0				
TF-IDF							
DOC2VEC							

TABLE 4.9: Comparaison de précision des modèles de similarité (classe de Sawm).

Modèle \ Prédiction	Position correcte (Première occurrence)	Position incorrecte					Total
		2ème	3ème	4ème	5ème	6ème	
LDA	124	49	22	16	7	6	224
LSI	213	11	0				
TF-IDF							
DOC2VEC							

TABLE 4.10: Comparaison de précision des modèles de similarité (classe de Zakat).

En interrogeant les modèles avec leurs données de formation, ils doivent retourner la même question en première sortie (la question plus similaire), ce test de performances a exposé la faiblesse de modèle LDA, tandis que les autres modèles avaient le même niveau de précision avec un score presque parfait.

Les résultats ci-dessus peuvent être repris dans le tableau suivant :

Position	Position correcte				Position incorrecte			
Classe Modèle	Hadj	Salat	Sawm	Zakat	Hadj	Salat	Sawm	Zakat
LDA	0.73	0.66	0.59	0.55	0.27	0.34	0.41	0.45
LSI	0.98	0.99	0.98	0.95	0.2	0.005	0.02	0.05
TF-IDF								
DOC2VEC								

TABLE 4.11: Comparaison de précision des modèles de similarité.

La différence est claire maintenant, les modèles DOC2VEC, TF-IDF et LSI ont bien identifié leurs données de formation malgré la taille modeste du corpus. Par contre le modèle LDA n'a pas pu identifier les questions de formation en premier lieu, malgré l'augmentation des passes jusqu'à 200 (par défaut c'est 1), l'algorithme n'a pas montré une grande différence de performances, les résultats des classes 'Hadj' en fait a diminué, mais les autres classes ont eu une petite amélioration.

En tant que verdict de leurs performances, nous pensons que c'est acceptable malgré la mauvaise précision de LDA, car les résultats soumis contiennent la question de voulu.

Les données de test représentent 30 % de taille total de chaque corpus, les tableaux suivants comprennent une comparaison des premiers résultats de chaque modèle.

Similarité	Similarité maximum				Total
Classe Modèle	Hadj	Salat	Sawm	Zakat	
TF-IDF	0	1	0	0	1
LSI	8	15	5	16	44
LDA	17	114	53	73	257
DOC2VEC	8	37	16	5	66
Total	33	167	74	94	368

TABLE 4.12: Comparaison de combien la première sortie d'un modèle avait la plus grande similitude entre les autres modèles.

Similarité	Similarité maximum			
Classe Modèle	Hadj	Salat	Sawm	Zakat
TF-IDF	0.77	0.99	0.95	0.50
LSI	0.79	0.98	0.95	0.83
LDA	1.0	1.0	1.0	1.0
DOC2VEC	0.85	0.98	0.96	0.89

TABLE 4.13: Comparaison de taux maximum de similarité eu dans chaque corpus.

Les résultats montrent que LDA avait la majorité des taux de similarité maximum, avec un total de 257 sur 368 questions. Cependant, le deuxième tableau expose une anomalie de taux maximum qui atteint jusqu'à 100 % dans tous les corpus, ces niveaux de similitude sont remarquables par rapport aux autres modèles.

Les tableaux suivants montrent une comparaison opposée de précédent, en comparant les similarités minimums.

Similarité Classe Modèle	Similarité minimum				Total
	Hadj	Salat	Sawm	Zakat	
TF-IDF	32	161	71	94	358
LSI	1	3	3	0	7
LDA	0	0	0	0	0
DOC2VEC	0	3	0	0	3
Total	33	167	74	94	368

TABLE 4.14: Comparaison de combien la première sortie d'un modèle avait la plus faible similitude entre les autres modèles.

Similarité Classe Modèle	Similarité minimum			
	Hadj	Salat	Sawm	Zakat
TF-IDF	0.15	0.17	0.12	0.03
LSI	0.43	0.33	0.38	0.03
LDA	0.41	0.49	0.52	0.41
DOC2VEC	0.45	0.34	0.43	0.36

TABLE 4.15: Comparaison de taux minimum de similarité eu dans chaque corpus.

Les résultats montrent que TF-IDF avait la majorité des taux de similarité minimum, avec un total de 358 sur 368 questions. Cependant, le deuxième tableau montre une différence remarquable de taux minimum qui atteint 35 % en moyenne.

En comparant la différence des résultats entre les modèles en un seuil de différence supérieur ou égal à 4 (sur les 5 premiers résultats), les tableaux suivants sont symétriques, et montrent le taux de différence dans les résultats des modèles de similarité.

Modèle Modèle	Modèle				
	Doc2vec	TF-IDF	LSI	LDA	Total
DOC2VEC	0	4	7	13	24
TF-IDF	4	0	0	7	11
LSI	7	0	0	6	13
LDA	13	7	6	0	26

TABLE 4.16: Comparaison des différents résultats selon le corpus de test de Hadj.

Le corpus de test de Hadj contient 33 questions, la comparaison montre une affinité dans les résultats, la plus grande différence est entre Doc2vec et LDA de 39%, la différence moyenne entre tous les modèles est 19%.

Modèle Modèle	Modèle				
	Doc2vec	TF-IDF	LSI	LDA	Total
DOC2VEC	0	57	105	124	286
TF-IDF	57	0	65	119	241
LSI	105	65	0	125	295
LDA	124	119	125	0	368

TABLE 4.17: Comparaison des différents résultats selon le corpus de test de Salat.

Le corpus de test de Salat contient 167 questions différents, la comparaison montre une dissemblance des résultats tel que la moindre différence est de 34% entre Doc2vec et TF-IDF, en moyenne les modèles ont 59% des résultats différents.

Modèle \ Modèle	Doc2vec	TF-IDF	LSI	LDA	Total
DOC2VEC	0	30	49	51	130
TF-IDF	30	0	20	33	83
LSI	49	20	0	39	108
LDA	51	33	39	0	123

TABLE 4.18: Comparaison des différents résultats selon le corpus de test de Sawm.

Le corpus de test de Sawm contient 74 questions, la comparaison montre un résultat équilibré de différence et similitude, la moyenne est de 50%.

Modèle \ Modèle	Doc2vec	TF-IDF	LSI	LDA	Total
DOC2VEC	0	39	73	74	186
TF-IDF	39	0	30	68	137
LSI	73	30	0	75	178
LDA	74	68	57	0	217

TABLE 4.19: Comparaison des différents résultats selon le corpus de test de Zakat.

Le corpus de test de Zakat contient 94 questions, la comparaison montre un peu de dissemblance entre les modèles, les résultats en gros sont 64% différents.

En général, les modèles Doc2vec et LSI ont été bornés par le modèle TF-IDF qui avait le moindre taux de différence, et le modèle LDA qui avait la majorité de différents résultats, néanmoins, leurs résultats n'étaient pas si similaires.

4.3 Proposition d'un modèle de déploiement

Les modèles d'apprentissage automatique sont pratiquement entraînés sur une grande quantité de données, nous en parlons jusqu'à 1000 fois la longueur de notre ensemble de données, quant à une solution à ce problème, nous déploierons nos modèles dans une application web afin d'obtenir infrastructure nécessaire, cette application est un prototype à améliorer en augmentant la taille du corpus et même étendre les domaines des fatwas prises en charge.

En fourniront cette occasion, l'ensemble de données peut élargir considérablement, qui assure la continuité du projet.

4.3.1 Avantages de déploiement en ligne

Le déploiement des modèles dans une application web sera un excellent ajout au projet pour plusieurs raisons :

- Internet est aujourd'hui le moyen le plus simple d'obtenir les informations, les applications bureautiques ne sont pas si populaires, surtout en ce qui concerne les fatwas.

- Le projet bénéficiera à de nombreuses personnes, ce qui devrait être notre objectif final.

Il assurera la continuité du projet à long terme :

- Le projet vivra et prospérera au lieu d'être oublié.
- Le manque de données (fatwas) pourrait être dépassé avec l'aide des bonnes personnes.
- Des nouvelles questions (inexistantes dans le corpus) pourraient être posées par des utilisateurs et répondues par des muftis, de cette façon les modèles continueront à s'améliorer avec le temps.

4.3.2 Exposition de système en ligne

La fonctionnalité fondamentale de l'application web est le système de réponse aux questions des fatwas islamiques, les questions peuvent être transmises au système par toute personne naviguant sur le site web, dans une interface simple comme présenté ci-dessous :

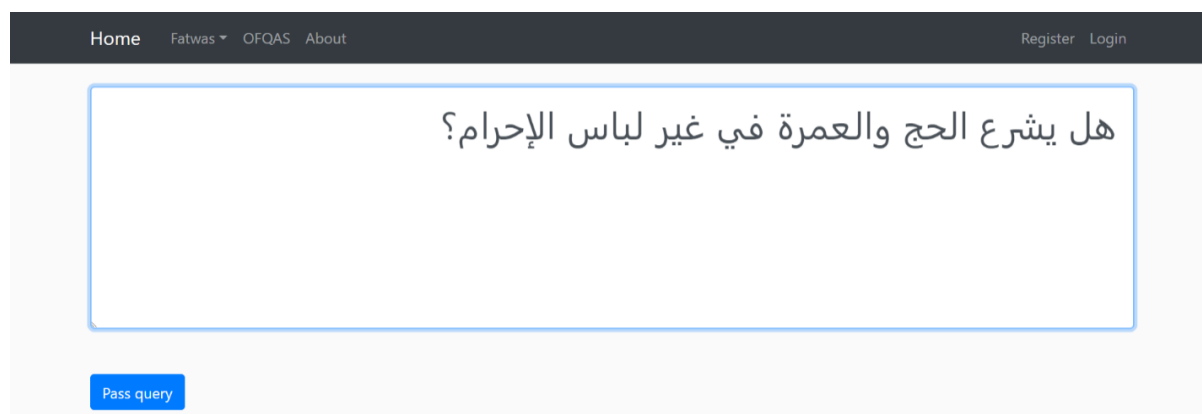


FIGURE 4.8 – Page de soumission des questions.

Après la soumission de question, le questionneur va connaître le résultat de classification, et a l'option de la changer en cas de fausse prédiction, la fenêtre est comme suite :

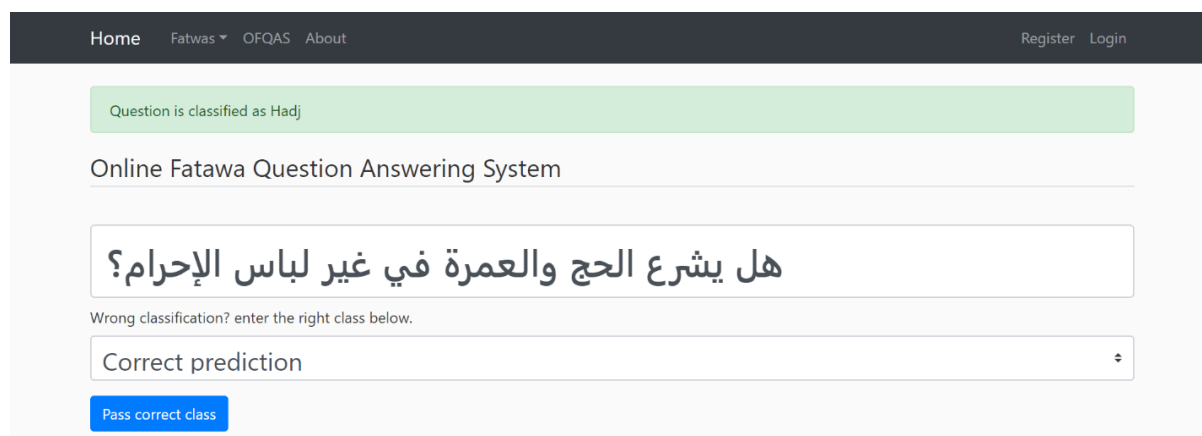


FIGURE 4.9 – Page de confirmation de classification et choix des modèles.

La dernière étape est de présenter les résultats ordonnés par taux de similarité, comme suite :



FIGURE 4.10 – Page de présentation des résultats.

L'application web contient beaucoup de fonctionnalités à côté de système de questions réponses pour enrichir le site, et faciliter l'amélioration de projet au fil du temps :

- Une gestion des fatwas maintenu par les muftis.
- Une gestion des questions posés par les utilisateurs, qui peuvent être répondu par les muftis.
- Une gestion des modèles pour les entrainer (mise à jour), les télécharger, et surveiller leurs performances.

Néanmoins ces fonctionnalités, l'application web n'est pas encore à sa forme définitive, il existe de nombreuses autres classes de fatwas qui peuvent être ajoutées pour élargir leur domaine de travail.

Conclusion

Dans ce dernier chapitre, nous avons présenté les différentes expérimentations menées aux modèles de routage et de similarité. Les résultats obtenus sont assez prometteurs vis-à-vis le corpus de Fatwa utilisé.

Le corpus utilisé contenait 1224 fatwas divisés en 4 classes, suite à la bonne architecture proposée on a pu avoir des résultats satisfaisants les normes de performances agréables d'un projet d'apprentissage automatique. Le déploiement des modèles en ligne devrait énormément influencer la taille actuelle de corpus.

Conclusion générale

Les recherches en systèmes de question/réponse en langue naturelle sont devenus de plus en plus nécessaires. Le texte religieux ne fait pas exception et la communauté musulmane s'attend à des outils technologiques performants pour répondre à leurs besoins. Le projet de cette thèse s'inscrivait dans ce cadre et visait à utiliser les méthodes d'intelligence artificielle et d'apprentissage profond pour faciliter la recherche, les questions et les réponses aux demandes des Fatwas de la charia islamique.

Ce travail a été un succès à bien des égards, nous sommes arrivés à construire non seulement un système accessible aux grand public à travers le web, mais aussi un système qui peut s'améliorer au fil du temps.

La réalisation de ce projet nous a conduit à explorer et à apprendre de nombreux domaines de l'informatique dont les systèmes de question-réponse, l'intelligence artificielle et les applications web.

Les Fatwas sont divisées en de nombreux sujets doctrinaux, pour un début nous en avons exploité quatre qui sont : (le pèlerinage (Hadj), la prière (Salat), le jeûne (Sawm) et la dîme (Zakat)), plus de sujets seront inclus plus tard.

Le corpus collecté par Setti et Belaribi [84] a fourni une ressource fondamentale pour la formation et l'évaluation des modèles d'apprentissage automatique. Une variété des modèles de routage et de similarité ont été entraînés et testés afin d'obtenir les meilleurs résultats.

Les performances de tous les modèles de routage étaient d'un niveau satisfaisant avec un minimum de 80% et un maximum de 95% de justesse. Les différentes expérimentations des modèles de similarité ont aussi montré que la majorité avait une excellente précision de 95% jusqu'à 99% dans le corpus de formation.

Comme travaux futures, nous envisageons enrichir le corpus de Fatwa et la création de nouveaux modèles de routage et de recherche de similarité distributionnelle de façon dynamique et participative.

Bibliographie

- [1] Accuracy and precision. URL https://en.wikipedia.org/wiki/Accuracy_and_precision.
- [2] Apprentissage automatique et deep learning. URL <https://www.stemmer-imaging.com/fr-ch/conseil-technique/apprentissage-automatique-et-apprentissage-profond/>.
- [3] Apprentissage profond ou deep learning. URL <https://www.universalis.fr/encyclopedie/apprentissage-profond-deep-learning/>.
- [4] Base de connaissance. URL https://fr.wikipedia.org/wiki/Base_de_connaissance.
- [5] Calculate similarity — the most relevant metrics in a nutshell. URL <https://towardsdatascience.com/calculate-similarity-the-most-relevant-metrics-in-a-nutshell-9a43564f533e>.
- [6] Choose a model. URL <https://developers.google.com/machine-learning/guides/text-classification/step-2-5>.
- [7] Classification. URL <https://scikit-learn.org/stable/modules/svm.html#svm-classification>.
- [8] Confusion matrix. URL https://en.wikipedia.org/wiki/Confusion_matrix.
- [9] Cosine similarity. URL <https://www.sciencedirect.com/topics/computer-science/cosine-similarity>.
- [10] Deep learning. URL <https://www.futura-sciences.com/tech/definitions/intelligence-artificielle-deep-learning-17262/>.
- [11] Deep learning ou apprentissage profond : définition, concept. URL <https://www.lebigdata.fr/deep-learning-definition>.
- [12] Doc2vec model. URL https://radimrehurek.com/gensim/auto_examples/tutorials/run_doc2vec_lee.html.
- [13] Evaluation of binary classifiers. URL https://en.wikipedia.org/wiki/Evaluation_of_binary_classifiers.
- [14] Explore your data. URL <https://developers.google.com/machine-learning/guides/text-classification/step-2>.
- [15] F1 score. URL https://en.wikipedia.org/wiki/F1_score.

- [16] Forests of randomized tree. URL <https://scikit-learn.org/stable/modules/ensemble.html#forests-of-randomized-trees>.
- [17] https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html. URL https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html.
- [18] Logistic regression. URL https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression.
- [19] macro average and weighted average meaning in classification_report. URL <https://datascience.stackexchange.com/questions/65839/macro-average-and-weighted-average-meaning-in-classification-report>.
- [20] models.doc2vec – doc2vec paragraph embeddings. URL <https://radimrehurek.com/gensim/models/doc2vec.html>.
- [21] models.ldamodel – latent dirichlet allocation. URL <https://radimrehurek.com/gensim/models/ldamodel.html>.
- [22] models.lsimodel – latent semantic indexing. URL <https://radimrehurek.com/gensim/models/lsimodel.html>.
- [23] models.tfidfmodel – tf-idf model. URL <https://radimrehurek.com/gensim/models/tfidfmodel.html>.
- [24] Multi-class text classification with scikit-learn. URL <https://towardsdatascience.com/multi-class-text-classification-with-scikit-learn-12f1e60e>.
- [25] Multi-label classification. URL https://en.wikipedia.org/wiki/Multi-label_classification.
- [26] Multinomial naive bayes. URL https://scikit-learn.org/stable/modules/naive_bayes.html#multinomial-naive-bayes.
- [27] Nlp — question answering system using deep learning. URL <https://medium.com/@akshaynavalakha/nlp-question-answering-system-f05825ef35c8>.
- [28] Nltk 3.5 documentation. URL <https://www.nltk.org/api/nltk.stem.html>.
- [29] Precision and recall. URL https://en.wikipedia.org/wiki/Precision_and_Recall.
- [30] Répertoire. URL <https://fr.wikipedia.org/wiki/R%C3%A9pertoire>.
- [31] Similarity learning. URL https://en.wikipedia.org/wiki/Similarity_learning.
- [32] sklearn.ensemble.randomforestclassifier. URL <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>.
- [33] sklearn.linear_model.logisticregression. URL https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html.

- [34] sklearn.metrics.f1_score. URL https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html.
- [35] sklearn.svm.linearsvc. URL <https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html>.
- [36] Statistical classification. URL https://en.wikipedia.org/wiki/Statistical_classification.
- [37] Text classification. URL <https://monkeylearn.com/text-classification/>.
- [38] Text classification. URL <http://www.big-data.tips/text-classification>.
- [39] Text classification : Applications and use cases. URL <https://towardsdatascience.com/text-classification-applications-and-use-cases-beab4bfe2e6>
- [40] Topics and transformations. URL https://radimrehurek.com/gensim/auto_examples/core/run_topics_and_transformations.html#sphx-glr-auto-examples-core-run-topics-and-transformations-py.
- [41] World wide web. URL https://fr.wikipedia.org/wiki/World_Wide_Web.
- [42] Zoom sur un type particulier d'extraction des connaissances : Les systèmes question réponse. URL <https://extractiondesconnaissances.wordpress.com/2012/03/19/61/>.
- [43] Y. H. ABAKER et M. RSHWAN : Semantic-based arabic question answering : Core and recent techniques. *International Journal of u- and e- Service*, 2017.
- [44] B. ABDELGHANI : Exploitation des données liées :système question-réponse. *these de doctorat*, 2019.
- [45] R. M. M. R. M. A.-F. B. E. N. . T. M. ABDELNASSER, H. : Al-bayan : an arabic question answering system for the holy quran. *the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, p. 57–64, 2014.
- [46] B. K. . R. P. ABOUENOUR, L. : Idraaq : New arabic question answering system based on query expansion and passage retrieval. *CLEF (Online Working Notes/Labs/Workshop)*, 2012.
- [47] C. N. M. AJITKUMAR MESHAM PUNDGE, Sunil Khillare : Question answering system, approaches and techniques : A review. *International Journal of Computer Applications*, 141(3), 2016.
- [48] M. K. AKOUR M., Abufardeh S. et A.-R. Q. : Qarabpro : A rule based question answering system for reading comprehension tests in arabic. *American Journal of Applied Sciences*, 8(6) :652–661, 2011.
- [49] H. M. AL CHALABI : Question processing for arabic question answering system. *Doctoral dissertation, The British University in Dubai (BUiD)*, 2015.
- [50] O. E. ASHRAF ELNAGAR, Ridhwan Al-Debsi : Arabic text classification using deep learning models. *International Journal of Information Processing & Management*, 2020.

- [51] H. H. ATHENIKOS S.J. : Biomedical question answering : A survey. *Computer Methods and Programs in Biomedecine*, 2010.
- [52] W. BAKARI : Une approche vers la comprehension automatique des textes arabes destinee pour les systemes de question-reponse. *Thèse de doctorat*, 2018.
- [53] B. P.-P. P. B. S.-. G. A. F. BANERJEE, S. : Multiple choice question (mcq) answering system for entrance examination. *CLEF (Working Notes)*, Septembre 2013.
- [54] . A.-H. M. BEKHTI, S. : Aquasys : A question-answering system for arabic. *In WSEAS International Conference. Proceedings. Recent Advances in Computer Engineering Series*, 2013.
- [55] A. BEN-ABACHA : Questions-réponses dans le domaine médical : une approche sémantique. *MajecSTIC 2009 Avignon*, novembre 2009.
- [56] A. BEN-ABACHA : Recherche de réponses précises à des questions médicales : le système de questions-réponses means. *PhD thesis. Université PARIS-SUD 11 LIM-SICNRS.*, JUIN 2012.
- [57] R. P.-. L. A. BENAJIBA, Y. : Implementation of the arabia question answering systems components. *Proc. Workshop on Arabic Natural Language Processing, 2nd Information Communication Technologies Int. Symposium, ICTIS2007*, p. 3–5, Avril 2007.
- [58] E. M.-. H. B. L. BRINI, W. : Qasal : Un système de question-réponse dédié pour les questions factuelles en langue arabe. *9ème Journées Scientifiques des Jeunes Chercheurs en Génie Electrique et Informatique, Tunisia*, 2009.
- [59] R. D et H. E. : Learning surface text patterns for a question answering system. *The 40th Annual Meeting on Association of Computational Linguistics* :41–47, 2002.
- [60] Y. DJERIRI : Les réseaux de neurones artificiels. *Journal of Theoretical and Applied Information Technology*, 2017.
- [61] I. ELHALWANY : Using textual case-based reasoning in intelligent fatawa qa system. *Journal of Information Technology*,, 2015.
- [62] K.-M. H. . E.-S. Y. EZZELDIN, A. M. : Alqasim : Arabic language question answer selection in machines. *International Conference of the Cross-Language Evaluation Forum for European Languages*, p. 100–103, Septembre 2013.
- [63] Z. L.-. E. O. FADER, A. : Open question answering over curated and extracted knowledge bases. *In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014.
- [64] . G.-B. GRAPPY, A. : Validation du type de la réponse dans un système de questions réponses. *Journal of Information Technology*,, p. 125–147, 2011.
- [65] A. GRAPPY : Validation de réponses dans un système de questions. *Mémoire de thèse de doctorat*, 2011.
- [66] . G.-V. GUPTA, P. : A survey of text question answering techniques. *International Journal of Computer Applications*, 2012.

- [67] A. S.-L. S. . E.-M. HAMMO, B. : Experimenting with a question answering system for the arabic language. *Computers and the Humanities*, p. 397–415, 2004.
- [68] Z. W. R. A. ITTYCHERIAH A, Franz M et M. R.J. : Ibm’s statistical question answering system. *Text Retrieval Conference TREC-9*, 2000.
- [69] S. M. KAMRUZZAMAN : Text classification using artificial intelligence. *International Journal*, 2010.
- [70] D. S. I. S. KOCALEVA, M. et Z. ZDRAVEV : Pattern recognition and natural language processing : State of the art. *TEM Journal*, p. 236–240, 2016.
- [71] . M. M. F. KOLOMIYETS, O. : A survey on question answering technology from an information retrieval perspective. *Information Sciences*, p. 5412–5434, 2011.
- [72] A. S. . A. N. KURDI, H. : Development and evaluation of a web based question answering system for arabic language. *Computer Science & Information Technology (CS & IT)*, p. 187–202, 2014.
- [73] M. E. B. LAURENT GILLARD, Patrice Bellot : Quelles combinaisons de scores et de critères numériques pour un système de questions/réponses ? *Bibliothèque Universitaire Déposants Hal-Avignon*, 2008.
- [74] U. V. S. M. . M.-E. LOPEZ, V. : Is question answering fit for the semantic web ? : a survey. *Semantic Web*, p. 125–155, 2011.
- [75] . J. S. K. MISHRA, A. : A survey on question answering systems with classification. *Journal of King Saud University-Computer and Information Sciences*, 2015.
- [76] A. E. S. S. MOHAMMAD ABDEEN, Sami Albouq : A closer look at arabic text classification. *International Journal of Advanced Computer Science and Applications*, 2019.
- [77] N. K. e. H. H. M. MOHAMMED, F. A. : A knowledge based arabic question answering system (aqas). *ACM SIGART Bulletin*, 4(4) :21–30, 1993.
- [78] P. M. H. S. . S. M. MOLDOVAN, D. : Performance issues and error analysis in an open-domain question answering system. *ACM Transactions on Information Systems (TOIS)*, p. 133–154, 2003.
- [79] B. W. N et G. N. K. : Development of yes/no arabic question answering system. *International Journal of Artificial Intelligence & Applications (IJAIA)*, 4(1), Janvier 2013.
- [80] H. G. M. G. . R.-G. P. NIU, Y. : Answering clinical questions with role identification. *Proceedings of the ACL 2003 workshop on Natural language processing in biomedicine*, 13:73–80, Juillet 2003.
- [81] V. M. PHO : Génération de réponses pour un système de questions-réponses. *In CORIA*, p. 449–454, 2012.
- [82] D. J. S. G. . P.-A. RINALDI, F. : Answering questions in the genomics domain. *the ACL 2004 Workshop on Question Answering in Restricted Domains*, p. 46–53, Juillet 2004.

- [83] A. E. SAIDALAVI KALADY et R. DAS. : Natural language question generation using syntax and keywords. *The 3rd Workshop on Question Generation*.
- [84] Z. SETTI et W. BELARIBI : Classification automatique des fatwas islamiques : une approche à base d'apprentissage approfondi. *Mémoire de Master Université de Khemis Miliana*, Juin 2019.
- [85] R. L. W. Z. SHAOHUA TENG, Junlei Li : The calculation of similarity and its application in data mining. *Joint International Conference on Pervasive Computing and the Networked World ICPCA/SWS 2013*, 8351, 2014.
- [86] M. SHEKER, S. SAAD, R. ABOOD et M. SHAKIR : Domain-specific ontology-based approach for arabic question answering. *Journal of Theoretical and Applied Information Technology*, 2015.
- [87] B. SILVA, C. et Ribeiro : Inductive inference for large scale text classification : Kernel approaches and techniques. 255, 2009.
- [88] K. SÉJOURNÉ : Questions réponses et interactions. *Thèse de doctorat*, 2009.
- [89] B. L. H. u. R. P. TRIGUI, O. : Defarabicqa : Arabic definition question answering system. *In Workshop on Language Resources and Human Language Technologies for Semitic Languages*, p. 40–45, 2010.
- [90] G. P. USUNIER N., Amini M. : Boosting weak ranking functions to enhance passage retrieval for question answering. *IR4QA workshop of SIGIR*, 2004.
- [91] . P. P. WYSE, B. : Generating questions from openlearn study units. *Proceedings of the 2nd Workshop on Question Generation, AIED*, 2009.

