

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Benyoucef BENKHEDDA-Alger1



Faculté des Sciences
Département des Mathématiques et Informatique

Projet de fin d'étude pour l'obtention du diplôme
de Master en Informatique

Option : Ingénierie des Systèmes Informatiques Intelligents

Thème

Deep Question Answering Systems for
Information Retrieval

Encadré par :

- D^r ZIANI Amel
- D^r BOUADJENEK Mohamed Reda

Réalisé par :

- AGABI Rayane Younes
- TIDAFI Asma

2019/2020

Remerciements

Dédicaces

A. Tidaï, Y. Agabi

Résumé

Abstract

Table des matières

Table des matières	i
Table des figures	iii
Liste des tableaux	iv
Liste des abréviations	v
1 Introduction	1
1.1 Introduction	1
1.2 Problématique et motivation	2
1.3 Contribution	3
1.4 Plan du mémoire	3
2 État de l’art	4
2.1 Introduction	4
2.2 Classification selon le domaine d’application	4
2.2.1 Closed-domain QA Systems	5
2.2.2 Open-domain QA Systems	6
2.2.3 Social QA Systems	7
2.3 Classification selon d’autres critères	9
2.4 Conclusion	12
3 Background	13
3.1 Introduction	13
3.2 Recherche d’Information	13
3.3 Natural Language Processing	13
3.4 Deep Learning	13
3.5 Conclusion	13
4 Présentation de la solution	14
4.1 Introduction	14
4.2	14
4.3 Conclusion	14

5	Discussion	15
5.1	Introduction	15
5.2	15
5.3	Conclusion	15
6	Conclusion	16
	Bibliographie	17
A	Titre de l'annexe	22
B	Titre de l'annexe	23

Table des figures

2.1	La taxonomie de l'état de l'art des systèmes QA.	4
2.2	Classification des systèmes QA.	10

Liste des tableaux

2.1	Classification des systèmes selon plusieurs axes	11
-----	--	----

Liste des abréviations

QAS	Question-Answering Systems
QA	Question-Answering
ODQAS	Open-Domain Question-Answering Systems
CDQAS	Closed-Domain Question-Answering Systems
KB	Knowledge Bases
IR	Information Retrieval

Chapitre 1

Introduction

1.1 Introduction

L'un des principaux défis de l'informatique est de construire des systèmes plus intelligents et capables de comprendre les êtres humains sans qu'on leur dise explicitement ce qu'ils doivent faire. Depuis les années 60, une percée majeure dans ce domaine se présente sous la forme de systèmes Questions-Réponses (Question-Answering Systems ou QAS). Le système QA est, comme son nom l'indique, un système qui peut répondre à des questions au lieu d'encombrer l'utilisateur avec des documents ou même des passages correspondants, comme le fait la plupart des systèmes de recherche d'informations basiques.

Dès leur début, les majeurs défis des systèmes QA sont la précision, l'habileté à répondre à toutes les questions complexes correctement avec une performance semblable à celle des humains. Dans ce document, notre objectif est de mettre en œuvre un modèle qui garantit des performances comparables à l'état de l'art actuel. Pour avoir une vision plus claire sur les systèmes QA actuels, prenons d'abord un moment pour comprendre la structure du problème et pourquoi les solutions existantes ne sont pas tout à fait suffisantes pour répondre à des questions complexes. Les systèmes QA sont généralement classés en deux grandes catégories : les QAS pour le domaine ouvert (ODQAS) et les QAS pour le domaine fermé (CDQAS).

Les systèmes QA du domaine fermé sont des systèmes aptes pour répondre à des questions dans un domaine spécifique (médecine, informatique,...etc), ce qui implique un ensemble de questions assez limité et restreint. Nous citons par exemple Joost [1], AQUA [2], Start [3] qui sont des systèmes Question-Answering du domaine fermé. La mise en œuvre de ce type de systèmes peut être considérée comme une tâche plus facile, car les systèmes peuvent utiliser des connaissances spécifiques à un domaine fréquemment formalisées dans des ontologies.

Pour les systèmes Question-Answering du domaine ouvert, de tels systèmes permettent de répondre à des questions qui ne sont pas limitées à des domaines prédéfinis, ainsi, les questions peuvent être sur quasiment n'importe quel sujet. Pour arriver à cette fin, le système devrait être capable de passer au crible une très grande quantité de documents textuels pour trouver la réponse, ce qui est une tâche plus complexe et défiante que les QAS du domaine fermé.

En ce qui concerne la source de connaissances¹ des systèmes QA et la façon avec laquelle ces derniers s'en servent, plusieurs approches ont vu le jour durant l'évolution des techniques et des sources de données. Parmi ces approches, nous trouvons les systèmes QA basés sur le texte, les faits, le Web, la recherche d'information (Information Retrieval ou IR) et les règles [4]. Pour notre modèle, nous allons opter pour l'approche de la recherche d'information dans une collecte de données basée sur les articles de Wikipedia² seulement. La recherche d'information, contrairement aux autres approches, sollicite et fait la recherche l'information dans des sources qui ne sont pas forcément structurées ce qui permet une meilleure flexibilité dans le cas d'ajout et d'extension des sources de recherche.

1.2 Problématique et motivation

De nos jours, suite à l'utilisation croissante des appareils mobiles, tels que les smartphones, pour accéder à l'information et recevoir une réponse directe à une question pour laquelle les requêtes traditionnelles consistant à spécifier des mots-clés ne sont pas très conviviales, c'est donc devenu l'une des fonctions les plus désirables pour les consommateurs d'information.

La majorité des connaissances humaines qui représentent les besoins d'information détaillés d'un utilisateur sont uniquement représentées par le langage naturel. Ils sont accessibles aux humains, qui peuvent comprendre les textes en langage naturel et répondre à des questions relatives à leur contenu, mais ne sont pas accessibles et compréhensibles pour les machines. Ces dernières ne peuvent donc pas comprendre et interpréter les énoncés des requêtes en langage naturel.

La tâche de l'extraction automatisée d'informations spécifiques à partir d'une source de connaissances, en tant que réponse à une question en langage naturel, n'est pas simple, même pour des ressources d'informations relativement réduites. La question doit être représentée comme une requête et la réponse doit être courte et précise. Nous pouvons extraire des informations factuelles explicites à partir d'un texte, mais l'extraction d'informations conceptuelles qui nécessitent également une compréhension du discours reste un objectif lointain. Pour obtenir des réponses précises, il faut formuler le besoin d'information de manière exacte et bien exprimée, au-delà d'un petit ensemble de termes vagues, comme c'est généralement le cas pour la recherche de documents, car d'une part, les requêtes en langage naturel sont réduites à des recherches basées sur des mots-clés. D'autre part, les bases de connaissances sont interrogées avec des requêtes structurées ou logiques obtenues à partir des questions en langage naturel, et les réponses sont obtenues par raisonnement.

Le langage naturel est ambigu (une phrase peut avoir un ou plusieurs sens) et syntaxiquement riche (un seul et même sens peut être véhiculé par de nombreuses expressions du langage naturel). La tâche de trouver une réponse à une question, lorsque les deux sont en langage naturel, est traditionnellement traitée par le domaine de la réponse aux questions ou Question Answering (QA). Les QAS reposent d'abord sur une analyse syntaxique de la question et une

1. **Source de connaissances** ("Knowledge source" en anglais) : C'est la source dans laquelle les systèmes QA fouinent à la recherche d'une réponse à une question donnée

2. <https://www.wikipedia.org/>

utilisation des techniques de recherche d'informations (RI) pour traiter le texte. Ensuite extraire des passages courts (snippets) suite à une classification par rapport à la probabilité de leur pertinence et de l'existence de l'information recherchée. Le système doit retourner uniquement les informations qui ont été spécifiquement demandées. Or, les demandes peuvent être complexes et narratives, ce qui signifie qu'il sera plus difficile pour le QA d'y répondre avec précision. De plus, les passages peuvent provenir de différents documents, nous devons donc les combiner pour fournir des réponses pertinentes, il se peut alors que nous ayons besoin d'un raisonnement complexe. Il sera donc difficile de formuler des réponses en langage naturel.

1.3 Contribution

Durant ce travail, nous nous concentrons sur les interactions entre l'extraction des réponses à l'aide du Deep Learning, le traitement du langage naturel et la recherche d'informations. Nous allons mettre en œuvre une architecture générale d'un QA en utilisant des collections et des ensembles de données de référence. Le but est que les utilisateurs reçoivent des réponses directes extraites de documents à leurs requêtes en langage naturel au lieu des documents pertinents pour enfin mettre en forme la réponse et la présenter à l'utilisateur.

1.4 Plan du mémoire

Chapitre 2

État de l’art

2.1 Introduction

Le Question-Answering est un domaine de recherche qui a connu un intérêt remarquable durant ces dernières années, ce qui a permis une avancée majeure par les chercheurs de ce domaine. Ce chapitre résume l’étude bibliographique effectuée. Il porte sur la qualité des systèmes existants en général et les multiples dimensions qui permettent de caractériser, d’évaluer et de classer ces systèmes afin de connaître l’état de l’art de ce domaine.

2.2 Classification selon le domaine d’application

Dans le domaine des QAS, nous faisons souvent référence à la classification de ces derniers en fonction du domaine dans lequel ils opèrent et répondent aux questions de ce dernier. Comme le montre la taxonomie illustrée dans la figure 2.1. Les QAS peuvent être classés selon leur domaine d’application. Il existe trois classes qui sont : Closed-domain QAS, Open-domain QAS et Social QAS.

Dans cette section, nous allons discuter les caractéristiques de ces classes en fournissant des exemples des QAS selon certains critères afin de comparer et extraire les avantages et les limites de chaque domaine d’application.

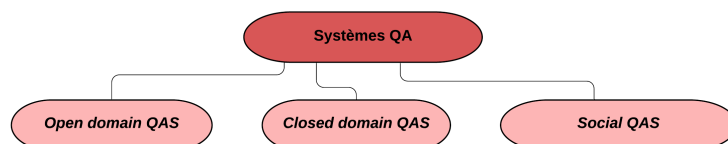


FIGURE 2.1 – La taxonomie de l’état de l’art des systèmes QA.

2.2.1 Closed-domain QA Systems

Les systèmes QAS du domaine fermé ou (domaine restreint) permettent de répondre aux questions relatives à un domaine particulier (médecine, cinématographie, aquariophilie, etc) en se basant sur les connaissances spécifiques aux domaines souvent formalisés dans des ontologies. Par ailleurs, des types limités de questions sont acceptés dans ces systèmes [5]. Ce domaine nécessite une disposition linguistique pour comprendre le texte en langue naturelle afin d'apporter une réponse précise aux requêtes [6].

Les premiers QAS créés sont LUNAR [7] en 1961 et BASEBALL [8] en 1972 qui ont été les premiers exemples d'interfaces en langage naturel avec des bases de données (NLIDB), i.e. interroger une base de données structurée à l'aide de questions en langage naturel [9]. LUNAR répond aux questions en langage naturel sur l'analyse géologique des roches renvoyées par les missions lunaires Apollo. Il est capable de répondre à 90% des questions dans son domaine posées par des utilisateurs non formés au système. Tandis que BASEBALL, répond aux questions sur les dates, les lieux et les résultats des matchs de baseball. Ces premiers systèmes encodaient de grandes quantités de connaissances spécifique au domaine dans des bases de données. Les QAS du domaine restreint modernes sont moins dépendants des grandes KB et ne se concentrent pas dans la compréhension des langues comme le faisaient les anciens systèmes. Nous citons dans le domaine de la médecine, le premier système nommé MYCIN [10] qui explique les concepts médicaux. MedQA [11] qui a pour objectif de répondre à tous les types de questions médicales. Ce système intègre des techniques de recherche, d'extraction et de résumé des informations afin de générer automatiquement un texte au niveau des paragraphes pour les questions de définition (eg : "Qu'est-ce que X?"). Toujours dans le domaine de la médecine, HONqa [12], EAGLi¹ et askHERMES [13] qui ont utilisé MEDLINE² comme source majeure de réponses. HONqa³ est un système multilingue (anglais, français et italien) qui a été conçu. Il utilise des sites web de santé certifiés qui permettent d'orienter les informations vers des personnes ayant des niveaux différents de connaissances en matière de santé [14]. Tandis que askHERMES⁴ réponds à tous les types de questions. Les réponses sont présentées de trois façons : réponses regroupées par termes, liste de réponses simples classées et réponses regroupées par contenu [14]. D'autre part, loin du domaine médical, le site web KAAS [15] qui a été développé pour être utilisé dans un environnement d'apprentissage (Advanced Interactive Environnement de découverte pour l'ingénierie Education ou AIDE) a pour but de répondre aux questions des étudiants de premier cycle de deux universités spécialisées dans l'aéronautique l'ingénierie. WEBCOOP [16] qui a été conçu et développé pour le domaine de tourisme. Il utilise une représentation logique des données et facilite leur interrogation avec des requêtes en langage naturel. De même, Diekema et al. [] qui ont contruit un QAS pour la recherche d'information dans le domaine de l'ingénierie aérospatiale.

Malgré la précision que ces systèmes ont atteinte puisqu'ils sont spécialisés dans des domaines précis, leur restriction les rend moins utiles au moment de vouloir avoir des réponses à des questions dans divers spécialités. Mais les QAS du domaine fermé peuvent être combinés pour créer des QAS de domaine ouvert (Indurkhy et al., 2010 ; Lopeza et al., 2011) [6].

1. eagl.unige.ch/EAGLi

2. <https://www.nlm.nih.gov/bsd/medline.html>

3. services.hon.ch/cgi-bin/QA10/qa.pl

4. <https://www.askhermes.org/>

2.2.2 Open-domain QA Systems

D'autre part, les QAS du domaine ouvert ne sont pas limités à un domaine spécifique et fournissent une réponse courte à une question, traitée en langage naturel. En outre, les questions peuvent être sur quasiment n'importe quel sujet [5]. Ces systèmes recherchent généralement des réponses au sein d'une vaste collection de documents. Il existe un grand nombre de questions qui peuvent être posées par des utilisateurs occasionnels dans les systèmes QA du domaine ouvert, et afin de répondre à ces questions, ce type de systèmes exploite l'ontologie générale et la connaissance du monde dans leurs méthodologies pour générer des réponses. En général, la qualité des réponses fournies par ces systèmes n'est pas aussi précise que les systèmes QA du domaine fermé. Les systèmes QA du domaine ouvert ne nécessitent pas de vocabulaire spécifique au domaine. Ils recherchent des réponses dans des grandes collections de documents [17]. En contrepartie, ils permettent aux utilisateurs la possibilité de poser des questions sans connaître les mots clés du domaine spécifique pour formuler des questions, ce qui favorise l'utilisation de systèmes QA du domaine ouvert par tous les utilisateurs des différents niveaux d'instruction et des différents domaines de spécialisation. De plus, Ces systèmes ne nécessitent pas le dictionnaire d'un domaine spécifique, ce qui veut dire que Wikipédia peut être utilisée comme source d'information.

Comme instance de ces systèmes, nous citons le plus ancien nommé START⁵ qui est disponible et accessible au public sur internet depuis 1993. Les réponses données par START sont extraites d'une très large liste de sources, telles que World Book, The World Factbook 2008, START KB, Internet Public Library, et bien d'autres [18]. Un autre système du domaine ouvert est le QuALiM [19] financé par Microsoft. Lorsqu'une question est posée, QuALiM recherche dans toutes les descriptions de questions des modèles et retient celles qui correspondent à la question.

De nos jours, les QAS du domaine ouvert sont les plus courants, nous citons celui de Seunghyun Yoon et ses collègues [20] qui est basé sur la sélection de réponses et entraîné à l'aide du jeu de données TrecQA. HyperQA [21] proposé par Yi Tay et qui est basé sur les réseaux de neurones convolutifs. Il a été évalué sur quatre ensembles de données ; YahooCQA, WikiQA, SemEvalCQA et TrecQA. DeepQA proposé par les chercheurs d'IBM et qui repose sur une architecture massivement parallèle. DeepQA a été appliqué avec succès à la fois aux datasets Jeopardy et TrecQA. BERT est aussi un système proposé par Google, c'est un encodeur bidirectionnel basé sur les transformateurs⁶. De nombreux Systèmes QA utilisent le modèle BERT comme modèle de base. Comme systèmes assez récents, nous avons ceux ALBERT [22] (A Lite BERT) proposé par l'équipe de chercheurs de Google Research⁷ et Toyota Technological Institute of Chicago⁸ qui vient améliorer le modèle BERT [23]. Les chercheurs ont établi de nouveaux résultats sur les datasets GLUE, SQuAD et RACE pour la compréhension des langues naturelles. De même, Zhilin Yang et son équipe, ont proposé le système à domaine ouvert XLNet [24], qui est un modèle basé sur BERT. En un mot, XLNet est une méthode généralisée d'autorégressive. Cela permet de prédire le mot suivant en utilisant le contexte du mot actuel en misant sur une seule direction qui peut être soit vers l'avant (forward context) ou vers l'arrière (backward context) contrairement au modèle BERT qui utilise une approche bidirectionnelle.

5. <http://start.csail.mit.edu/index.php>

6. **Transformer-based Models** : Une architecture pour transformer une séquence en une autre à l'aide de deux parties (encodeur et décodeur).[36]

7. <https://research.google/>

8. <https://www.ttic.edu/>

En Janvier 2020, des chercheurs de 4 départements de l'université de Shanghai Jiao Tong⁹ située à Shanghai en Chine, ont contribué à la conception d'un lecteur rétrospectif (Retrospective Reader [25]) qui est capable d'effectuer une vérification des réponses de manière suffisante et efficace au lieu de simplement empiler les vérificateurs dans les lecteurs existants. Le Retrospective Reader a été testé et évalué sur les 2 benchmarks SQuAD2.0 et NewsQA¹⁸ et a donné des résultats meilleurs que le modèle ALBERT. Nous citons aussi le modèle BiDAF [26] proposé par Seo et al.[39]. C'est un réseau de neurones récurrent hiérarchique à plusieurs niveaux end-to-end¹⁰ qui prend des entrées de granularité différente (caractères, mots et phrases) pour obtenir une représentation contextuelle à l'aide d'une attention sans mémoire. Un autre système qui est aussi basé sur les réseaux de neurones récurrents, nous citons DrQA [27]. Sa mise en œuvre comporte des couches d'encodage de paragraphes et de questions, ainsi qu'une couche de sortie. Contrairement aux deux systèmes précédents, QAnet [28] ne nécessite pas de réseaux récurrents. L'objectif principal des chercheurs ici, est de rendre la compréhension de la machine rapide. Cette architecture accélère considérablement le modèle sur l'ensemble de données SQuAD. Et enfin, TANDA [29] Transfer and Adaptation, qui, en l'évaluant TANDA sur les ensembles de données TREC-QA et WikiQA ainsi que sur trois ensembles de données industrielles différents, les résultats ont montré qu'elle améliore l'état de l'art dans AS2¹¹. TANDA a atteint une précision beaucoup plus élevée que les modèles traditionnels, en particulier dans le cas de données sur le bruit.

2.2.3 Social QA Systems

Malgré la précision et la complétude que les QAS ont atteint de nos jours, on se trouve toujours face à plusieurs problèmes [30], dont la mauvaise expression de la question et l'absence des réponses pertinentes. Pour cela, un nouveau type de QAS est apparu qui est le Social QAS. Il met en relation les utilisateurs afin collecter le contenu nécessaire pour répondre aux questions. Ce type de QAS permet de fournir un moyen pour répondre à plusieurs types de questions telles que la recommandation (eg. Recherche d'opinions, connaissance factuelle, résolution de problèmes...) en prenant en compte la fréquence de ces questions et les motivations des utilisateurs qui interrogent leur réseau social plutôt que d'utiliser un moteur de recherche traditionnel [31]. D'une part, social QAS incluent un composant de réseautage social (social networking), par lequel les utilisateurs peuvent créer des liens avec d'autres utilisateurs en posant et en répondant à des questions et en établissant une réputation dans la communauté pour leurs contributions de contenu. D'autre part, ils offrent une vaste base de connaissances évolutive de questions et réponses. Ils constituent une ressource précieuse pour les utilisateurs ayant des besoins d'informations similaires [32].

Le concept des QAS social est apparu dans les débuts des années 2000, lorsque The AnswerBank¹² a été réalisé qui représente une plateforme où les utilisateurs posent des questions et obtiennent de vraies réponses de vraies personnes et donnent quelques réponses en retour.

9. <http://en.sjtu.edu.cn/>

10. **L'apprentissage de bout en bout ("end-to-end" en anglais) :** Une approche d'apprentissage qui optimise les poids du réseau en considérant directement les entrées et les sorties.

11. **AS2 :** Given a question and a set of answer sentence candidates, consists in selecting sentences (e.g., retrieved by a search engine) correctly answering the question.

12. <https://www.theanswerbank.co.uk/>

Ensuite, en 2001, Wikipedia¹³ a créé un nouveau service presque similaire nommé Wikipedia Reference Desk¹⁴. Dans ce QAS, les utilisateurs posent leurs questions, et des volontaires chez Wikipedia offrent des solutions/réponses. En 2002, des anciens employés chez Samsung (Jeon et al.) ont développé Naver's KiN [33] qui a été lancé la première fois en Corée du Sud par NHN Corporation¹⁵. Contrairement aux autres QAS Social, sur NKIN les utilisateurs peuvent non seulement poser et répondre à plusieurs questions mais aussi communiquer entre eux via email ou messages. Un an après, la concurrence a pris de l'ampleur, trois social QAS ont pris place sur internet; AnswerBag¹⁶, FunAdvice¹⁷ et Ask Me Help Desk¹⁸. AnswerBag créé par Joel Downs, donne la possibilité à ses utilisateurs de poser et répondre à des questions dans plus de 4000 catégories. Il ne contient pas la fonctionnalité de sélection de la meilleure réponse mais comporte un système de notation qui permet de trier les réponses selon leurs notes [34]. Sur AnswerBag, on peut attacher une image ou une vidéo avec la question/réponse. En 2005, deux systèmes sont apparus Answers.com¹⁹ et Yahoo! Answers²⁰. Ce dernier autorise les utilisateurs à publier une question en détail avec le titre de la question dans une catégorie relative à laquelle la question appartient. D'autres bénévoles de la communauté sont autorisés à répondre. L'utilisateur qui pose la question a la possibilité de sélectionner la meilleure réponse qui a résolu son problème ou a répondu à sa question [35]. En 2006 et 2007, construire un QAS Social est devenu l'intention de tous les chercheurs. On cite; Blurtit²¹, AOL Answers²², Chacha²³, Fluther²⁴, LinkedInAnswers, Wiki Answers²⁵ et AskVille²⁶. Celui qui a marqué le plus de succès pendant ces deux années est le Wiki Answers. Il a été lancé autant que FAQ Farm en 2002 par Chris Whitten ensuite acheté par Answer Corporation en Novembre 2006. Wiki Answers permet aux utilisateurs de modifier les question et réponses existantes pour améliorer la qualité du contenu. Les questions redondantes sont identifiées, fusionnées et généralisées. Wiki Answers contient plus de 7,000 categories []. Avant la fin de la première décennie, la première version de StackOverflow²⁷ créée par Jeff Atwood et Joel Spolsky a été lancée; un site de QAS pour les programmeurs professionnels et passionnés. Il propose des questions et réponses sur un large éventail de sujets en programmation informatique. Durant la deuxième décennie, pleins de sites web qui tentent à leur tour contribuer à ce domaine nous citons Ask.fm²⁸,

13. <https://www.wikipedia.org/>

14. https://en.wikipedia.org/wiki/Wikipedia:Reference_desk

15. <https://www.nhn.com/>

16. <https://www.answerbag.com/>

17. [funadvice.com](https://www.funadvice.com/)

18. <https://www.askmehelpdesk.com/>

19. <https://www.answers.com/>

20. <https://answers.yahoo.com/>

21. <https://www.blurtit.com/>

22. <https://www.aol.com/>

23. <http://www.chacha.com/>

24. <https://www.fluther.com/>

25. <https://wiki-answers.com/>

26. <https://askville.com.cutestat.com/>

27. <https://stackoverflow.com/>

28. <https://ask.fm/>

Quora²⁹, Zhihu.com³⁰, Aardvark³¹, Brilliant.org³² et Spring.me lancé en 2006 sous le nom de formspring³³.

Les systèmes cités dans cette section ne sont pas assez complets par rapport au temps de réponse et à la qualité des réponses. Il est donc important de classer correctement des réponses dans les systèmes QA. Pour cela, Dalip et Al. [36] a proposé une méthode d'apprentissage qui consiste à l'utilisation des caractéristiques spécifiques du domaine des questions-réponses. Cette approche a donné des résultats extraordinaires et a produit un nouvel état de l'art.

2.3 Classification selon d'autres critères

A l'origine, on désignait par un système Question-Answering la recherche de réponses dans des assemblages de documents non structurés. L'évolution de la quantité des données ont poussé de nombreuses nouveautés récentes à apparaître telles que : WebQuestions [37] et SimpleQuestions [38] fondées sur le KB Freebase [39], ou sur des bases de connaissances (Knowledge bases ou KB) extraits automatiquement, par exemple, OpenIE triples et NELL [40]. Néanmoins, les bases de connaissances KB exposent des limites attachées (incomplétude, schémas fixes) qui ont fait appel aux chercheurs pour revenir au cadre original de réponses à partir du texte brut.

Une deuxième motivation qui avait pour but d'attirer le regard à nouveau sur ce problème est celle de la compréhension automatique du texte, autrement dit répondre à des questions après avoir lu un court texte ou une histoire. Ce sous-domaine a progressé considérablement dernièrement grâce à de nouvelles structures d'apprentissage approfondi comme les réseaux de neurones basés sur l'attention (Attention mechanism³⁴) et la mémoire augmentée [41] et à la publication de nouveaux jeux de données d'entraînement et d'évaluation comme QuizBowl [42], CNN/DailyMail [43] basé sur des articles de presse, CBT [44] basé sur des livres pour enfants, ou SQuAD [45] et WikiReading [46], tous deux basés sur Wikipédia.

Aujourd'hui, les systèmes QA sont de plus en plus nombreux et qui offrent une diversité imposante. En conséquence, la classification par domaine d'application n'est plus suffisante pour comparer et définir toutes les caractéristiques des systèmes QA. Pour cela, nous avons établi d'autres critères de classification (classification par type de question, source de données et par source de connaissance) qui permettent de mettre en valeur les différences de ces systèmes et extraire les avantages et les limites de chacun d'eux.

Dans la figure 2.2, nous illustrons les différents axes de classification selon lesquels nous allons classer et comparer les systèmes QA existants.

29. <https://quora.com/>

30. <https://www.zhihu.com/>

31. [https://en.wikipedia.org/wiki/Aardvark\(searchengine\)](https://en.wikipedia.org/wiki/Aardvark(searchengine))

32. <https://brilliant.org/>

33. <https://fr.wikipedia.org/wiki/Spring.me>

34. **Le mécanisme d'attention :** Un mécanisme qui permet au modèle de se concentrer et d'accorder plus d'attention aux parties pertinentes de la séquence d'entrée selon les besoins.

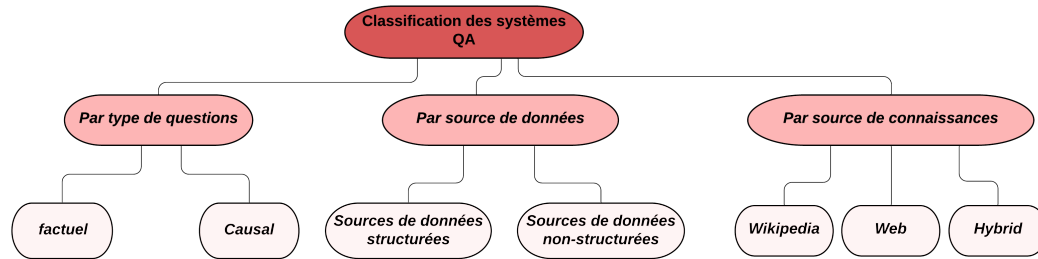


FIGURE 2.2 – Classification des systèmes QA.

En outre, le tableau 1 résume les différents axes selon lesquels nous allons classer les systèmes QA tout en expliquant ces axes et des exemples de chaque catégorie de système QA.

Système	Domaine d'application			Type de question						Source de données		Type de réponse	
	Ouvert	Fermé	Social	Factuelle	Causale	Confirmation	Liste	Définition	Procédurale	Structuré	Non Structurée	Passage	Réponse Générée
BASEBALL		X		X		X				X		X	
LUNAR		X		X			X	X		X			X
MedQA		X						X		X		X	
MYCIN		X											
HONqa		X		X				X	X	X		X	
EAGLi		X						X		X		X	
askHERMES		X		X	X	X	X	X	X	X		X	
KAAS		X		X									
WEBCOOP		X			X	X		X	X				
START	X												
QuALIM	X												
HyperQA	X												
DeepQA	X												
BiDAF	X												
BERT	X												
ALBERT	X												
DrQA	X												
QAnet	X												
TANDA	X												
XLNet	X												

TABLE 2.1 – Classification des systèmes selon plusieurs axes

Comme nous pouvons le voir dans le tableau 2.1, les systèmes QA peuvent être vus et classés selon plusieurs critères. Parmi ces critères, nous trouvons tout d'abord le domaine d'application du système qui peut être un domaine de type ouvert ou de type fermé. Une autre vision de la classification nous a permis de classer les systèmes selon le type de questions prises en charge par ces derniers. Aussi, nous avons utilisé une autre approche de classification qui est la source de connaissances. Cette classification permet de classer les systèmes selon leurs sources sur lesquelles ils se basent pour extraire les informations afin de répondre à une question. Dans notre étude bibliographique, nous avons constaté différentes approches de recherche d'une réponse, ce qui nous a incité à classer les systèmes selon différentes approches de recherche. Enfin, chaque système fournit une réponse d'une façon différente. Pour cela, nous avons établi une classification selon les types de réponses de ces systèmes.

2.4 Conclusion

L'objectif de ce chapitre était de donner un aperçu global sur les systèmes réalisés jusqu'à présent dans le domaine du Question-Answering, ainsi que les problèmes et les faiblesses de ce que les chercheurs ont réussi à produire de mieux jusque là. De plus, nous avons vu les différentes approches appliquées lors de l'extraction des réponses ainsi que les diverses sources de connaissances des systèmes QA. Par ailleurs, nous avons vu les différents axes de classification selon lesquels nous pouvons classer et évaluer un système QA.

Chapitre 3

Background

3.1 Introduction

3.2 Recherche d'Information

3.3 Natural Language Processing

3.4 Deep Learning

3.5 Conclusion

Chapitre 4

Présentation de la solution

4.1 Introduction

4.2

4.3 Conclusion

Chapitre 5

Discussion

5.1 Introduction

5.2

5.3 Conclusion

Chapitre 6

Conclusion

Bibliographie

- [1] van Noord G. Bouma G., Mur J. Reasoning over dependency relations. 2005.
- [2] Maria Vargas-Vera and Enrico Motta. Aqua – ontology-based question answering system. In Raúl Monroy, Gustavo Arroyo-Figueroa, Luis Enrique Sucar, and Humberto Sossa, editors, *MICA I 2004 : Advances in Artificial Intelligence*, pages 468–477, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.
- [3] Boris Katz, Sue Felshin, Deniz Yuret, Jimmy Lin, Gregory Marton, Alton McFarland, and Baris Temelkuran. Omnibase : Uniform access to heterogeneous data for question answering. pages 230–234, 06 2002.
- [4] R. Mervin. An overview of question answering system. 1, 10 2013.
- [5] Passent M. ElKafrawy, Amr M. Sauber, and Nada A. Sabry. Semantic question answering system using dbpedia. In *Lecture Notes in Computer Science*, pages 821–832. Springer International Publishing, 2018.
- [6] Bolanle Ojokoh and Emmanuel Adebisi. A review of question answering systems. *Journal of Web Engineering*, 17 :717–758, 01 2019.
- [7] W. A. Woods. Progress in natural language understanding. In *Proceedings of the June 4-8, 1973, national computer conference and exposition on - AFIPS 73*. ACM Press, 1973.
- [8] Bert F. Green, Alice K. Wolf, Carol Chomsky, and Kenneth Laughery. Baseball. In *Papers presented at the May 9-11, 1961, western joint IRE-AIEE-ACM computer conference on - IRE-AIEE-ACM 61 (Western)*. ACM Press, 1961.
- [9] Oleksandr Kolomiyets and Marie-Francine Moens. A survey on question answering technology from an information retrieval perspective. *Information Sciences*, 181(24) :5412–5434, dec 2011.
- [10] Edward Shortliffe. Mycin : A knowledge-based computer program applied to infectious diseases*. *Proceedings / the ... Annual Symposium on Computer Application [sic] in Medical Care. Symposium on Computer Applications in Medical Care*, 10 1977.
- [11] Minsuk Lee, James Cimino, Hai Zhu, Carl Sable, Vijay Shanker, John Ely, and Hong Yu. Beyond information retrieval—medical question answering. *AMIA Annu Symp Proc*, pages 469–473, 02 2006.
- [12] María-Dolores Olvera-Lobo and Juncal Gutiérrez-Artacho. Multilingual question-answering system in biomedical domain on the web : An evaluation. In *Multilingual and Multimodal Information Access Evaluation*, pages 83–88. Springer Berlin Heidelberg, 2011.
- [13] YongGang Cao, Feifan Liu, Pippa Simpson, Lamont Antieau, Andrew Bennett, James J. Cimino, John Ely, and Hong Yu. AskHERMES : An online question answering system for complex clinical questions. *Journal of Biomedical Informatics*, 44(2) :277–288, apr 2011.

- [14] Michael A Bauer and Daniel Berleant. Usability survey of biomedical question answering systems. *Human Genomics*, 6(1), sep 2012.
- [15] Anne R. Diekema, Ozgur Yilmazel, and Elizabeth D. Liddy. Evaluation of restricted domain question-answering systems. In *Proceedings of the Conference on Question Answering in Restricted Domains*, pages 2–7, Barcelona, Spain, July 2004. Association for Computational Linguistics.
- [16] Farah Benamara and Patrick Saint-Dizier. Webcoop : A cooperative question-answering system on the web. 04 2003.
- [17] A Chandra Obula Reddy and K Madhavi. A survey on types of question answering system. *IOSR-JCE*, 19(6) :19–23, 2017.
- [18] María Dolores Olvera-Lobo and Juncal Gutiérrez-Artacho. Open- vs. restricted-domain qa systems in the biomedical field. *Journal of Information Science*, 37 :152–162, 04 2011.
- [19] Michael Kaisser. Qualim at trec 2005 : Web-question answering with framenet. In *TREC*, 2005.
- [20] Seunghyun Yoon, Franck Deroncourt, Doo Kim, Trung Bui, and Kyomin Jung. A compare-aggregate model with latent clustering for answer selection. pages 2093–2096, 11 2019.
- [21] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. Hyperbolic representation learning for fast and efficient neural question answering. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining - WSDM 18*. ACM Press, 2018.
- [22] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert : A lite bert for self-supervised learning of language representations, 2019.
- [23] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert : Pre-training of deep bidirectional transformers for language understanding, 2018.
- [24] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet : Generalized autoregressive pretraining for language understanding, 2019.
- [25] Zhuosheng Zhang, Junjie Yang, and Hai Zhao. Retrospective reader for machine reading comprehension, 2020.
- [26] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension, 2016.
- [27] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv :1704.00051*, 2017.
- [28] Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. Qanet : Combining local convolution with global self-attention for reading comprehension, 2018.
- [29] Siddhant Garg, Thuy Vu, and Alessandro Moschitti. Tanda : Transfer and adapt pre-trained transformer models for answer sentence selection, 2019.
- [30] Dan Moldovan, Marius Paşca, Sanda Harabagiu, and Mihai Surdeanu. Performance issues and error analysis in an open-domain question answering system. *ACM Transactions on Information Systems (TOIS)*, 21(2) :133–154, apr 2003.

- [31] Mohamed Reda Bouadjenek, Hakim Hacid, and Mokrane Bouzeghoub. Social networks and information retrieval, how are they converging? a survey, a taxonomy and an analysis of social information retrieval approaches and platforms. *Information Systems*, 56, 03 2016.
- [32] Eduarda Mendes Rodrigues and Natasa Milic-Frayling. Socializing or knowledge sharing? In *Proceeding of the 18th ACM conference on Information and knowledge management - CIKM 09*. ACM Press, 2009.
- [33] Kevin Kyung Nam, Mark S. Ackerman, and Lada A. Adamic. Questions in, knowledge in? In *Proceedings of the 27th international conference on Human factors in computing systems - CHI 09*. ACM Press, 2009.
- [34] Rich Gazan. Seven words you cant say on answerbag. In *Proceedings of the 27th ACM Conference on Hypertext and Social Media - HT 16*. ACM Press, 2016.
- [35] Lada A. Adamic, Jun Zhang, Eytan Bakshy, and Mark S. Ackerman. Knowledge sharing and yahoo answers. In *Proceeding of the 17th international conference on World Wide Web - WWW 08*. ACM Press, 2008.
- [36] Daniel Hasan Dalip, Marcos André Gonçalves, Marco Cristo, and Pavel Calado. Exploiting user feedback to learn to rank answers in q&a forums. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval - SIGIR 13*. ACM Press, 2013.
- [37] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.
- [38] Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. Large-scale simple question answering with memory networks. *ArXiv*, abs/1506.02075, 2015.
- [39] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase : A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, page 1247–1250, New York, NY, USA, 2008. Association for Computing Machinery.
- [40] Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. Open question answering over curated and extracted knowledge bases. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, page 1156–1165, New York, NY, USA, 2014. Association for Computing Machinery.
- [41] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2015.
- [42] Mohit Iyyer, Jordan Boyd-Graber, Leonardo Claudino, Richard Socher, and Hal Daumé III. A neural network for factoid question answering over paragraphs. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 633–644, Doha, Qatar, October 2014. Association for Computational Linguistics.
- [43] Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. *CoRR*, abs/1506.03340, 2015.
- [44] Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. The goldilocks principle : Reading children’s books with explicit memory representations. *CoRR*, abs/1511.02301, 2015.

- [45] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad : 100, 000+ questions for machine comprehension of text. *CoRR*, abs/1606.05250, 2016.
- [46] Daniel Hewlett, Alexandre Lacoste, Llion Jones, Illia Polosukhin, Andrew Fandrianto, Jay Han, Matthew Kelcey, and David Berthelot. WikiReading : A novel large-scale language understanding task over Wikipedia. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 1535–1545, Berlin, Germany, August 2016. Association for Computational Linguistics.
- [47] Pum-Mo Ryu, Myung-Gil Jang, and Hyun-Ki Kim. Open domain question answering using wikipedia-based knowledge model. *Inf. Process. Manage.*, 50(5) :683–692, September 2014.
- [48] David Ahn, Valentin Jijkoun, Gilad Mishne, Karin Müller, Maarten Rijke, and Stefan Schlobach. Using wikipedia at the trec qa track. 01 2004.
- [49] Davide Buscaldi and Paolo Rosso. Mining knowledge from wikipedia for the question answering task. In *In Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC’06)*, page 727–730, 2006.
- [50] Huan Sun, Hao Ma, Wen-tau Yih, Chen-Tse Tsai, Jingjing Liu, and Ming-Wei Chang. Open domain question answering via semantic enrichment. In *Proceedings of the 24th International Conference on World Wide Web, WWW ’15*, page 1045–1055, Republic and Canton of Geneva, CHE, 2015. International World Wide Web Conferences Steering Committee.
- [51] Eric Brill, Susan Dumais, and Michele Banko. An analysis of the AskMSR question-answering system. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 257–264. Association for Computational Linguistics, July 2002.
- [52] Petr Baudiš and Jan Šedivý. Modeling of the question answering task in the yodaqa system. In *Proceedings of the 6th International Conference on Experimental IR Meets Multilinguality, Multimodality, and Interaction - Volume 9283, CLEF’15*, page 1156–1165, Berlin, Heidelberg, 2015. Springer-Verlag.
- [53] Seunghyun Yoon, Franck Dernoncourt, Doo Soon Kim, Trung Bui, and Kyomin Jung. A compare-aggregate model with latent clustering for answer selection. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM ’19*, page 2093–2096, New York, NY, USA, 2019. Association for Computing Machinery.
- [54] Weijie Bian, Si Li, Zhao Yang, Guang Chen, and Zhiqing Lin. A compare-aggregate model with dynamic-clip attention for answer selection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM ’17*, page 1987–1990, New York, NY, USA, 2017. Association for Computing Machinery.
- [55] Cody Kwok, Oren Etzioni, and Daniel S. Weld. Scaling question answering to the web. *ACM Trans. Inf. Syst.*, 19(3) :242–262, July 2001.
- [56] Diego Molla Aliod and José González. Question answering in restricted domains : An overview. *Computational Linguistics*, 33 :41–61, 03 2007.
- [57] Vanessa Lopez, Enrico Motta, Marta Sabou, and Miriam Fernandez. Poweraqua : A multi-ontology based question answering system–v1.
- [58] Zhiping Zheng. Question answering using web news as knowledge base. In *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Lin-*

guistics - Volume 2, EACL '03, page 251–254, USA, 2003. Association for Computational Linguistics.

- [59] Wilson Wong. Practical approach to knowledge-based question answering with natural language understanding and advanced reasoning, 2007.

Annexe A

Titre de l'annexe

Vous pouvez mettre ici, par exemple, l'implémentation d'un algorithme qui a été présenté dans le corps du travail ou une description de la syntaxe des langages de programmation utilisés dans le texte.

Annexe B

Titre de l'annexe

Vous pouvez mettre ici la présentation de l'organisme d'accueil par exemple.