UniversitÃ¤t Heidelberg
Institut fÃ¼r Informatik
Arbeitsgruppe Datenbanksysteme

Master-Arbeit

Building an adaptable and resource constrained Conversational Information Search System

Name:	Stephan Lenert
Matrikelnummer:	Matrikelnummer der Autorin/des Autors
Betreuer:	Name der Betreuerin / des Betreuers
Datum der Abgabe:	September 21, 2023

Ich versichere, dass ich diese Master-Arbeit selbststÃ¤ndig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe und die GrundsÃ¤tze und Empfehlungen “Verantwortung in der Wissenschaft” der UniversitÃ¤t Heidelberg beachtet wurden.

Abgabedatum: September 21, 2023

Zusammenfassung

Die Zusammenfassung muss auf Deutsch **und** auf Englisch geschrieben werden. Die Zusammenfassung sollte zwischen einer halben und einer ganzen Seite lang sein. Sie soll den Kontext der Arbeit, die Problemstellung, die Zielsetzung und die entwickelten Methoden sowie Erkenntnisse bzw. Ergebnisse übersichtlich und verständlich beschreiben.

Abstract

The abstract has to be given in German **and** English. It should be between half a page and one page in length. It should cover in a readable and comprehensive style the context of the thesis, the problem setting, the objectives, and the methods developed in this thesis as well as key insights and results.

Contents

List of Acronyms

List of Figures

List of Tables

1 Introduction

This chapter is an introduction to the topic of this thesis. It starts with a brief overview of the current state of the art in the field of question answering and chatbots. Then, it describes the motivation behind this thesis and the goals that are to be achieved. Finally, it gives an overview of the structure of this thesis.

2 Background and Related Work

This chapter provides essential background information and reviews relevant prior research. It commences with an introduction to the sub-task of Question Answering (QA), as presented in Section 2.1. As previously mentioned in the Introduction (Chapter 1), this chapter maintains a clear distinction between QA and Conversational Question Answering (Conv QA). Consequently, Section 2.2 extends upon the foundational knowledge of QA and introduces the requisite concepts for the transformation of a QA-System into a Conv QA-System.

2.1 Question Answering

The evolution of QA as a research field provides a solid foundation for understanding current research initiatives and methodologies. Among the early contributions is BASEBALL, an automated QA system developed by researchers at Massachusetts Institute of Technology (MIT) in 1961. This QA system demonstrated its capability to answer questions related to baseball using natural English language [2].

In 1999, Text REtrieval Conference (TREC) (Text Retrieval Conference) initiated the TREC-8 Question Answering track, which marked "the first large-scale evaluation of domain-independent question-answering systems" [3]. A more well-known QA system is *Watson* by IBM, an open-domain QA system that won a the TV show Jeopardy! in 2011 [4]. It is evident that an evolutionary process has occurred between the early research in 1961 and today's systems like *ChatGPT* by OpenAI. To understand the dimensions in which these systems differ, their components, and how to distinguish them will be introduced in Section 2.1.1, while subsequent sections will delve deeper into specific components.

In 1999, the TREC initiated the TREC-8 Question Answering track, marking "the first large-scale evaluation of domain-independent question-answering systems" [3]. A more renowned QA system is IBM's *Watson*, an open-domain QA system that famously triumphed on the television game show Jeopardy! in 2011 [4]. It is evident that an evolutionary process has transpired between the early research in 1961 and contemporary systems such as OpenAI's *ChatGPT*. In section 2.1.1 we will lay the groundwork by introducing the fundamental aspects of QA-Systems and the techniques used to differentiate and categorize them. Following that, subsequent sections will delve deeper into the examination of specific system components.

2.1.1 Basics

Jurafsky and Martin define a QA-System as a system "designed to satisfy human information needs" [5]. Hence, it primarily functions as an Information Retrieval System, with its primary objective being to provide users with the desired and accurate information in response to natural language requests.

The research community has yet to establish a universally accepted classification framework for Question Answering (QA) systems. For instance, Hao et al. and Farea et al. [6, 1] take a comprehensive approach to classify QA systems but differ in certain aspects, such as their treatment of question types and knowledge sources. On the other hand, other researchers [7, 5, 8, 9] employ a similar classification methodology but often focus solely on retrieval-based approaches, thereby lacking a holistic perspective.

The classification proposed by Farea et al. [1] goes a step further by distinguishing between the **QA-Framework** and **QA-Paradigms**, enhancing its versatility for comparing classical and modern QA systems. An adaptation of this classification will be utilized in this thesis. The originally proposed QA algorithms have been extended to include the Retrieval-based approach, and the Question Types have been revised based on the typology introduced by Mishra et al. in their 2016 survey [10], which was further elaborated upon by Etezadi et al. [8]. In this context, a crucial distinction is made between a **QA** and **ConvQA** system, guided by the criteria outlined in [11]: a QA system exclusively handles standalone questions, while any inquiry exceeding a single question and involving conversational context falls within the domain of a **ConvQA** system.

The **QA-Framework** encompasses external factors such as Question and Answer Types, while also considering system-related factors like the QA Algorithm and Knowledge Source [1, 6]. Conversely, the **QA-Paradigm** defines the fundamental underlying concept of a system and can be seen as a subset of possible combinations within the **QA Framework**. Currently, three dominant paradigms prevail:

1. **Information Retrieval (IR)-Based QA**: This paradigm involves searching through extensive multi-modal data based on a user's question and using the retrieved passages to generate an answer.

2. **Knowledge Base (KB) QA**: In this approach, a semantic representation of the question is constructed, and a knowledge base is queried using this representation.

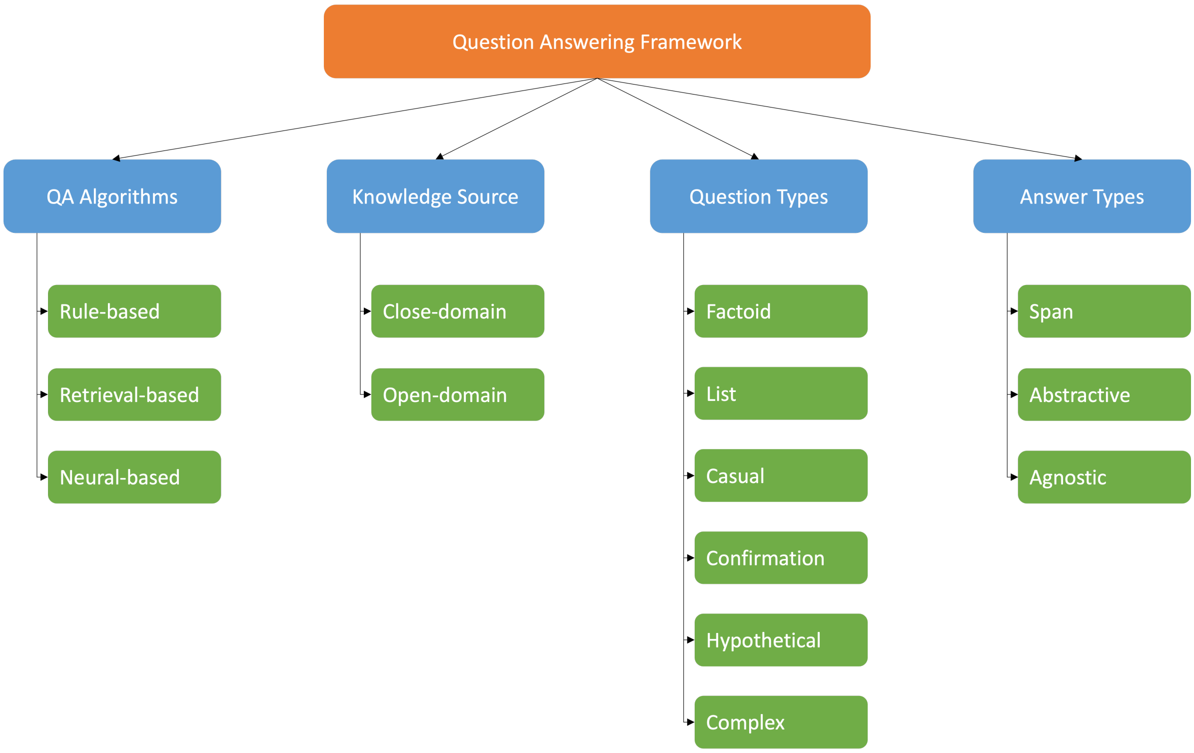

Figure 2.1: Adjusted QA Framework Classification by Farea et al. [1]

The returned results are then used to generate an answer.

3. **Generative Question Answering**: Here, knowledge is fully explicit, and a neural network (NN) generates answers based on its trained parameters.

For visual clarity, a diagram illustrating the adjusted QA Framework Classification by Farea et al. is provided in Figure 2.1.

Figure 2.1 illustrates the aforementioned classification. The primary distinguishing factor is the employed **QA Algorithm**. Rule-based approaches involve the manual crafting of feature extractions from user questions, which are then compared to the knowledge base. Rule-based approaches are typically employed in closed-domain QA systems exclusively [8].

Retrieval-based approaches are the classic Information Retrieval (IR)-based QA systems, comprising two key components: an intent classifier and a retriever. The intent classifier's objective is to discern the question's intent and identify important entities. Subsequently, the retriever searches the knowledge source and identifies the most relevant passages [1, 7].

The Neural-based approach, often referred to as the generative approach, utilizes a Sequence-to-Sequence (S2S) model to generate accurate answers to given questions. In this paradigm, the information is stored directly in the neural network's parameters, otherwise the neural network is part of a Retrieval-based approach. Most datasets in these contexts consist of triples of question, context, and answer pairs [5]. Notably, widely used datasets such as SQuAD and QASPER originally emerged from the field of machine reading comprehension, representing a foundational step in the evolution of QA systems [12, 13, 7].

In addition to the **QA Algorithms**, the **Knowledge Source** plays a pivotal role in distinguishing various aspects of Question Answering (QA) systems. The nature of the knowledge source can range from structured to unstructured or semi-structured, and it may encompass diverse data modalities, including text, audio, and video. A common point of comparison in the QA landscape is between closed and open-domain systems.

In the broad sense, a **closed-domain** QA system operates within the confines of a specific knowledge domain, which means it has limited access to information. In contrast, **open-domain** QA systems grapple with an extensive array of knowledge sources, necessitating a more versatile approach [1].

Furthermore, a closed-domain setup often entails limitations on the types of questions it can handle, primarily focusing on factoid questions or predefined templates. Additionally, it frequently relies on structured knowledge bases like graphs or logically organized repositories [6].

Conversely, open-domain QA systems are designed to tackle a wide spectrum of user queries, ranging from factoids to more complex inquiries. They typically deal with unstructured knowledge sources, which can be substantial and diverse in content [7, 1, 5].

An alternative perspective for distinguishing QA-Systems lies in the **Question Types** that users can input into the system. Questions can fall into various categories, such as *factoid, list, casual, confirmation, hypothetical* [10], or *complex* [8].

- *Factoid questions*, the most common type, are typically signaled by question words (what, when, which, who, how) and yield a concise factual answer.
- *List questions* represent a specialized subset of factoid questions, where the answer comprises a list of facts.
- *Casual questions* encompass inquiries that deviate from the factoid format, often involving words like *how* or *why* and requiring more advanced reasoning.
- *Confirmation questions* seek simple yes or no responses, frequently employed in personal assistant applications.
- *Hypothetical questions* delve into hypothetical scenarios (e.g., "what would happen if"), aiming for plausible rather than definitive answers.

- *Complex questions* can be further categorized into *answer-retrieval-complex* and *question-understanding-complex*. In the case of question-understanding-complex questions, the complexity arises from nuances like multiple constraints, making the question itself intricate to comprehend. In contrast, answer-retrieval-complex questions involve complexities in finding the correct answer, often requiring the combination of information from multiple documents or similar sources. This is commonly referred to as long-form QA.

Lastly, a QA-System can be characterized by the **Answer Types** it offers, a concept closely intertwined with Question Types. Farea et al. [1] delineate three categories of answers: *span, abstractive,* and *agnostic*.

- *Span answers* represent the most common type, where the answer is a specific factual excerpt presented as a span of tokens.
- *Abstractive answers* often arise in response to confirmation questions and can be a system-generated reaction based on the user's provided answer.
- *Agnostic answers* typically correspond to complex questions that necessitate the system to consider multiple documents and information sources to formulate a response. In such cases, no predefined or annotated answer exists.

2.1.2 Indexing Approaches

2.1.3 Retrieval Approaches

2.1.4 User Interaction

2.1.5 Example Architectures

2.1.6 Limitations

2.2 Conversational Question Answering

3 Open-domain QA Chatbot over PDFs

This chapter is the main achievment of the Thesis. It consits of laying out different possible solutions to the given problem.

4 Experimental Evaluation

This chapter is the evaluation of the proposed solution. It consists of laying out different possible solutions to the given problem.

5 Conclusions and Future Work

This chapter is the conclusion of the thesis. It starts with a brief overview of the current state of the art in the field of question answering and chatbots. Then, it describes the motivation behind this thesis and the goals that are to be achieved. Finally, it gives an overview of the structure of this thesis.

Bibliography

[1] Amer Farea, Zhen Yang, Kien Duong, Nadeesha Perera, and Frank Emmert-Streib. Evaluation of question answering systems: Complexity of judging a natural language.

[2] Bert F. Green, Alice K. Wolf, Carol Chomsky, and Kenneth Laughery. Baseball: an automatic question-answerer. In *Papers presented at the May 9-11, 1961, western joint IRE-AIEE-ACM computer conference*, IRE-AIEE-ACM '61 (Western), pages 219–224. Association for Computing Machinery.

[3] E. Voorhees. The TREC-8 question answering track report.

[4] D. A. Ferrucci. Introduction to this is watson. 56(3):1:1–1:15. Conference Name: IBM Journal of Research and Development.

[5] Dan Jurafsky and James H. Martin. *Speech and Language Processing*. 3 edition.

[6] Tianyong Hao, Xinxin Li, Yulan He, Fu Lee Wang, and Yingying Qu. Recent progress in leveraging deep learning methods for question answering. 34(4):2765–2783.

[7] Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. Retrieving and reading: A comprehensive survey on open-domain question answering.

[8] Romina Etezadi and Mehrnoush Shamsfard. The state of the art in open domain complex question answering: a survey. 53(4):4124–4144.

[9] Qin Zhang, Shangsi Chen, Dongkuan Xu, Qingqing Cao, Xiaojun Chen, Trevor Cohn, and Meng Fang. A survey for efficient open domain question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14447–14465. Association for Computational Linguistics.

[10] Amit Mishra and Sanjay Kumar Jain. A survey on question answering systems with classification. 28(3):345–361.

[11] Hamed Zamani, Johanne R. Trippas, Jeff Dalton, and Filip Radlinski. Conversational information seeking.

[12] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text.

[13] Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner. A dataset of information-seeking questions and answers anchored in research papers.