

# An Interactive e-Government Question Answering System

Malte Schwarzer<sup>1</sup>, Jonas Düver<sup>1</sup>, Danuta Ploch<sup>2</sup>, and Andreas Lommatzsch<sup>2</sup>

<sup>1</sup> Technische Universität Berlin, Straße des 17. Juni, D-10625 Berlin, Germany  
{malte.schwarzer, jonas.duever}@campus.tu.berlin.de}

<sup>2</sup> DAI-Labor, TU Berlin, Ernst-Reuter-Platz 7, D-10587 Berlin, Germany  
{danuta.ploch, andreas.lommatzsch}@dai-labor.de

**Abstract.** Services for citizens provided by the government are often complex and related with various requirements. Citizens usually have a lot of questions traditionally answered by human experts. In this paper, we describe an information retrieval-based question answering (QA) system for the e-government domain. The QA system is capable of giving direct answers to questions in German concerning governmental services. The system successfully handles ambiguous questions by combining retrieval methods, task trees and a rule-based approach. We evaluate our system in a scenario tailored to the needs of the administration of a big German city. The preliminary results show that our system provides high-quality answers for the most questions.

**Keywords:** e-government, direct question answering, interactive IR

## 1 Introduction

Government services challenge both, citizens and agencies. Agencies continuously work on new ways for improving the efficiency and quality of services. IT-based solutions combining information retrieval (IR) and machine learning (ML) technologies are promising approaches for supporting the administration in improving the access to offered services. The administration of the German city Berlin already operates an online platform allowing users to inform themselves about all services the administrative agencies provide. However, the platform currently only offers a basic search but it does not provide direct answers to specific questions. Citizens must read through comprehensive service descriptions and find the piece of information they are interested in on their own. In addition, citizens often are not familiar with officialese making the formulation of search queries a challenging task for most users. In order to support the citizens getting informed, a system is needed that analyzes the users' intentions and applies advanced retrieval methods for providing detailed information tailored to the specific questions. In this work we present an IR-based e-government question answering system for German capable of handling three major tasks:

1. **Customized Ranking:** The system retrieves service descriptions and ranks them applying a customized scoring function that is based on service popularity.
2. **Passage Retrieval:** The system provides direct answers to user questions instead of showing comprehensive full-text documents requiring much effort for reading. Users do not need to scan the entire document anymore; users now quickly find a concrete answer to their question.
3. **Interactive QA:** The system is able to handle ambiguous and unclear questions. It addresses the problem by asking additional questions. If a user question is too general, the system checks back to refine the question.

The remaining paper is structured as follows. In Sec. 2 we describe the fundamentals of QA systems including our dataset, the evaluation metric, and existing QA systems. We present our approach in Sec. 3. The evaluation results are discussed in Sec. 4. Finally, a conclusion and an outlook to future work are given in Sec. 5.

## 2 Related Work

This Section describes common approaches to question answering and compares related question answering systems in the e-government domain. In particular, we review two major online services offering information for Berlin citizens.

### 2.1 Question Answering

Question answering systems find the correct answer to a (natural language) question based on a set of documents [4]. In general, there are two paradigms for QA: information retrieval-based and knowledge-based approaches.

Information retrieval-based systems answer a user's natural language question "by finding short text segments" [5, p. 2] in a collection of documents. These systems typically consist of three main components, often integrated in a pipeline: question classification, information retrieval, and answer extraction.

Knowledge-based approaches answer "a natural language question by mapping it to a query over a structured database" [5, p. 9]. Hence, they rely on already structured data, for example in a relational database. Among the knowledge-based approaches there are rule-based and supervised methods. Concerning rule-based methods the rules must be defined by hand, which is feasible for very frequent information needs. Supervised methods build a semantic representation of the user's query and then map it to the structured data.

Interactive QA is the combination of QA and dialogue systems in which users find answers in an interactive way. QA systems initiate a dialogue with the user in order to clarify missing or ambiguous information or to suggest further topics for discussion [6].

Our system uses an IR-based approach. It derives the answer types applying a set of rules defined by experts and retrieves passages as answers.

## 2.2 Related Systems

There are only a few publicly available online systems that enable Berlin's citizens to inform themselves of governmental services.

One of them is the nationwide platform "Behördenfinder". The platform comes as an extended redirection service. It passes the entered search terms unmodified to the respective search pages of the federal states. For example, the service redirects Berlin's citizens to the service portal of Berlin, which performs the search.

At the service portal of the city Berlin the citizens may search in the database of governmental services by entering keywords. The server returns a list of documents that contain the entered query terms, sorted by relevance. No further filter options are provided. The user has to open the links and manually search the appropriate section.

The service portals of other federal states often work in a similar way. But, some portals extend the search component by functions like "related" sections, categorizations (e. g. *buergerservice.niedersachsen.de*, *service-bw.de*), or similar search terms (e. g. *muenchen.de/dienstleistungsfinder*).

To the best of our knowledge, there is no question answering system available for Berlin's citizens able to answer government-related questions interactively and as accurately as possible.

## 3 Approach

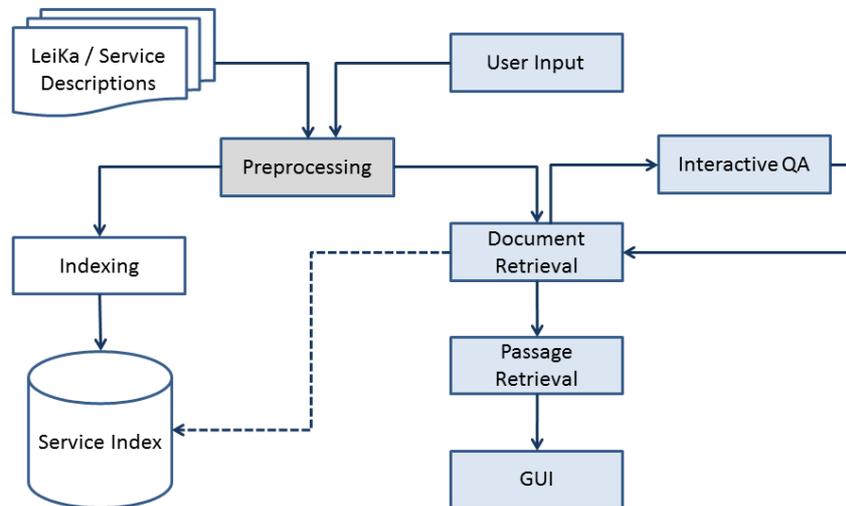
In this Section we describe the used data sources and the approach we index the data to make the content searchable. We explain our document retrieval strategy and present three ranking methods. Our approach to interactive question answering includes grouping search results by selected service features and offering additional filters to the users. In order to provide users with concrete answers we present a method to question categorization allowing the retrieval of appropriate document passages. A web GUI enables users to access the system and to ask questions about services offered by administrative agencies. Figure 1 shows the system components and their interaction.

### 3.1 Data Sources

The main data source of the implemented system is the *LeiKa* ("Leistungskatalog der öffentlichen Verwaltung" [1, p. 1]). The *LeiKa* is a catalog assembling and categorizing services in order to create a Germany-wide central information base with uniform descriptions of services offered by administration departments.

Each service is identified by a key and categorized using multiple levels:

1. "Leistungsobjekt", the *service object* the service description deals with, e. g. driver's license.
2. "Verrichtung", the *service action* to perform, e. g. whether the citizen applies for a driver's license or his license has to be replaced.
3. "Verrichtungsdetail", the *service action detail* to describe the *service action* more precisely, e. g. whether it is an EU or an international driver's license.



**Fig. 1.** The architecture of the presented interactive QA system consists of a service index, document and passage retrieval components, a component for enabling interactive searching, and a graphical user interface.

In addition, the LeiKa catalog provides for each service a set of standardized information such as a textual description, the costs for the service, the responsible authority and other necessary information [2, pp. 11-14].

The LeiKa catalog is already an important foundation of multiple projects, e. g. the uniform authority telephone service 115 (D115). In addition to the LeiKa service descriptions the D115 project provides popularity rankings for the top 100 services and commune-specific service descriptions [7, p. 5ff.].

Our system exploits a combination of the LeiKa and the D115 data of the Berlin government, including the ranking positions of the top 100 services.

### 3.2 Indexing

The data sources are provided in different formats. In order to make the data searchable we parse, aggregate, enrich and store it in an inverted index by using Elasticsearch [3]. This involves the annotation of the service documents with meta information, e. g. the popularity rankings (D115 top 100 services). We extend the services with additional keywords by applying NLP techniques. With the *Stanford* part-of-speech (POS) tagger we extract nouns and verbs from the title and the textual description of the services. Based on the extracted words, we determine additional keywords, i. e. synonyms and the stems of the words. We rely on the *Wortschatz* German dictionary web service [8] maintained by the University Leipzig.

### 3.3 Document Retrieval

The QA system is designed as an extension of a classical IR system. The retrieval of relevant documents builds the foundation of the question answering system. The retrieval part consist of two sub-tasks: (1) processing the user input and (2) formulating an appropriate query to search the document index. The user input is processed with the *Stanford* POS tagger and the *Wortschatz* web service in a similar manner as during the indexing process. We use a caching system to minimize response time and to reduce the number of required *Wortschatz*-queries. Based on the analyzed and enriched user input we formulate search queries. Our system implements the following three document scoring approaches to rank the retrieved documents:

*Keyword scoring* We assume that the description of relevant services contain at least one keyword of the extended user query. The more query keywords a service contains the more relevant it is. Therefore, we retrieve all services that contain at least one single keyword and sort them by the number of keyword occurrences within the service title or description.

*TF-IDF scoring* ElasticSearch supports full-text search by default, which is based on the term frequency-inverse document frequency (TF-IDF) scoring [9] from the Apache Lucene implementation<sup>3</sup>. For our retrieval task, we use the full-text search with the keywords from the user input on the complete service documents, whereby the weighting of document fields is normalized based on their content length.

*Custom scoring* Our custom scoring method (for ranking the documents) is an extension of the standard ElasticSearch scoring method based on the TF-IDF score. We modify the scoring function by adding a popularity factor to the formula, i. e. the rank of a service in the D115 top 100 list.

$$\text{score}(q, d) = \frac{1}{\text{D115-rank}(d)} \text{TF-IDF-score}(q, d) \quad (1)$$

Eq. 1 shows the custom scoring function of the query  $q$  and the service document  $d$ , where  $\text{D115-rank}(d)$  is the rank in the D115 top 100 list and  $\text{TF-IDF-score}(q, d)$  is the standard TF-IDF score.

### 3.4 Interactive Question Answering

The major challenge in interactive QA is the detection of ambiguities. Instead of detecting ambiguities in questions, we detect ambiguities in the retrieved services documents. We are capable of doing this, since the services descriptions are structured and not only plain texts.

The services are organized in objects, categories and actions. We extract these features from the retrieved services, group them and sort them by occurrence and popularity. We provide additional filters to ambiguous service results and let users interactively choose which object, category and action they are interested in.

<sup>3</sup> [http://lucene.apache.org/core/4\\_0\\_0/core/org/apache/lucene/search/similarities/TFIDFSimilarity.html](http://lucene.apache.org/core/4_0_0/core/org/apache/lucene/search/similarities/TFIDFSimilarity.html)

Was meinen Sie?  
Verrichtung:  
     
Objekte:

**Fig. 2.** The GUI for interactive QA: A user enters an ID-related query. Filters for ID card (“Personalausweis”), obligatory identification (“Ausweispflicht”) and others are shown.

### 3.5 Passage Retrieval

Our application is an e-government QA system. We assume that users only enter questions related to governmental services. Therefore, the set of questions, our system needs to be able to answer, is limited. Based on the information passages (provided in the service documents), we manually pick four distinct types of questions, which can be answered: the costs of a service, the required documents, the opening time, and the location of the agency responsible for offering the service. For each type of question we define a static set of keywords. If a query contains such keywords (case-insensitive), we determine the type of the question and give the respective answer, i. e. we provide the corresponding excerpt of the service document directly in the search results or we provide several excerpts if multiple types match.

### 3.6 GUI

We design a simple and clear graphical user interface (GUI) that reflects the essential features of the QA system. The layout of the search result page is presented as a screen shot in Fig. 3. Users can input their questions, refine their questions (if they are ambiguous or too general), and read the search results including the corresponding text passages.

## 4 Evaluation

To allow a comparison to other QA systems and to judge whether algorithmic changes improve the system or not, we evaluate the QA system in a quantitative and distinct manner. Hence, we investigate the performance of our document and passage retrieval approach with two task-specific gold standard data sets.

### 4.1 Gold Standard

In order to evaluate our approach we need a data set containing German e-government questions along with the correct answers. To the best of our knowledge there is no gold standard data set that satisfies our needs. Thus, we create two data sets: one data set for measuring the performance of the document retrieval and one data set for the passage retrieval part, respectively.

Auf den Serviceseiten finden Sie, was Sie suchen.

The screenshot shows a web interface for a question-answering system. At the top, a search bar labeled 'Ihre Frage:' contains the text 'Wo kann ich einen Personalausweis beantragen?'. Below this is a section titled 'Was meinten Sie?' with three categories: 'Verrichtung:' (Ausstellung, Zulassung, Zuteilung, Erteilung, Reservierung), 'Objekte:' (Personalausweis, Kraftfahrzeug, Kraftfahrzeugkennzeichen, Melderegisterauskunft, Kinderreisepass), and 'Gruppen:' (Personalausweis, Fahrzeugzulassung, Wohnsitz, Reisepass, Fahrzeugregister). The search results are displayed in a list. The first result is 'Personalausweis vorläufig beantragen' with a blue header and a yellow background for the passage: 'Bürgeramt 1 (Kreuzberg), Yorckstr. Yorckstraße 4-11, 10965 Berlin mehr'. The second result is 'Personalausweis beantragen' with a blue header and a yellow background for the passage: 'Bürgeramt 1 (Kreuzberg), Yorckstr. Yorckstraße 4-11, 10965 Berlin mehr'.

**Fig. 3.** The GUI of the QA system: The user input is highlighted in the red box, the interactive QA in the green box. The search results are displayed with service titles (blue) and respective passages (yellow).

For the evaluation of the document retrieval process we develop a data set consisting of 6,700 questions, partly generated through permutation of synonyms or similar terms. Each question is associated with the correct set of answer documents.

For the passage retrieval evaluation we annotate 70 questions with their corresponding answers. The answers consist of the LeiKa- and D115-ID, the answer type for the passage retrieval, and the relevant LeiKa category information (service object, action, action detail). The set of questions and answers focuses on our four implemented answer types (see Sec. 3.5).

## 4.2 Document Retrieval

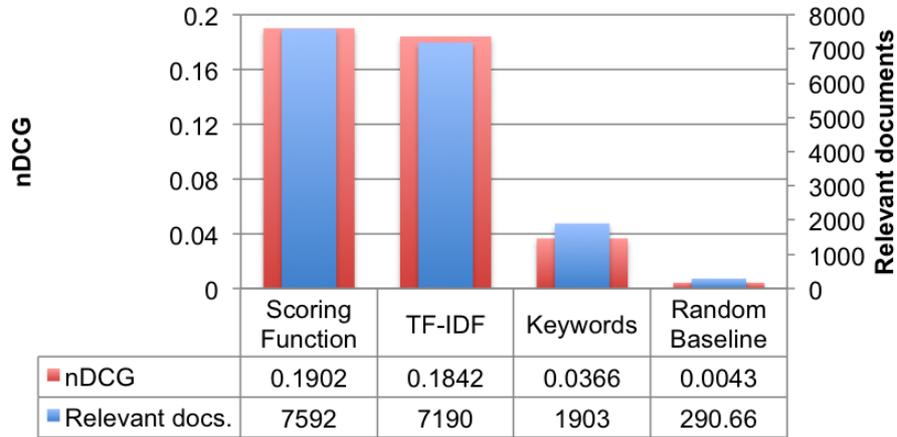
As our QA system follows an IR-based approach, our retrieval component provides a list of relevant services ordered by the relevance with respect to the user query. We assess the performance of our IR approach with the normalized Discounted Cumulative Gain (nDCG) measure. This measure considered both the relevance level and the order of the retrieved documents.

$$\text{DCG}_p = \sum_{i=1}^p \frac{2^{\text{rel}_i} - 1}{\log_2(i + 1)} \quad (2)$$

The standard Discounted Cumulative Gain (DCG) at a particular rank position  $p$  is computed as shown in Eq. 2. Our gold standard provides a binary relevance classification of service documents. Thus, the graded relevance of the result at position  $i$  is defined as  $rel_i \in \{0, 1\}$ .

$$\text{nDCG}_p = \frac{DCG_p}{IDCG_p} \quad (3)$$

In the e-government use-case of the QA system, we expect users to be only interested in the first few results and therefore we take only the top-k results with  $k=10$  into account. We compare three document retrieval methods (Section 3.3) with a random baseline, i. e. documents retrieved in a random order. The scores of the random baseline are calculated as average over three runs.



**Fig. 4.** Evaluation of query functions for document retrieval based on 6,700 questions and the top-10 documents of each approach.

The results in Fig. 4 show, that the custom scoring function leads to the best document retrieval performance in terms of nDCG and the total number of relevant documents. The inclusion of the popularity ranking affects the performance positively. The state-of-the-art TF-IDF method achieves lower but similar performance. On the contrary, the keyword-based approach yields rather poor results.

### 4.3 Passage Retrieval

We evaluate the passage retrieval component with a gold standard consisting of 70 questions and the corresponding passages.

Our QA system retrieves 54 passages (77%) correctly, while a random baseline achieves with 17.5 correct passages an accuracy of 25%.

## 5 Conclusion & Future Work

We developed a prototype of an e-government QA system applying a well-performing approach. The system is IR-based. It analyzes user questions, retrieves service documents from an inverted index and ranks them with a customized scoring function. We make use of the structured information encoded in the LeiKa catalog to provide a question categorization and passage retrieval feature and to interactively resolve ambiguous questions. A web-GUI enables users to interact with the system. We created two data consisting of 6,700 document retrieval and 70 passage retrieval questions. In a quantitative evaluation we showed that the use of the popularity ranking improves the retrieval quality.

The presented system is still work in progress. Hence, we propose two areas that require future work:

*Incrementally optimized rankings* Our nDCG evaluation already proves a good performance of the retrieved services. Anyhow, especially the ranking by relevance can be further improved by adjusting the ranking function to the domain of government services. We plan to evaluate additional user signals, e.g. clickstreams, to fine-tune this ranking function.

*Improved QA* A future goal is to provide QA features that do not rely on the structured information of the services or that answer only a limited set of question types. We aim for answering user questions based on knowledge and understanding and to optimize the handling of specific sub-tasks. In order to achieve this goal we plan to extend the data sets and to incorporate machine learning approaches.

## References

1. Geschäfts- und Koordinierungsstelle LeiKa / BFD. Information der Geschäfts- und Koordinierungsstelle LeiKa. [http://www.gk-leika.de/fileadmin/user\\_upload/gk-leika.de/dokumente/Startseite/News\\_der\\_GK\\_18082010.pdf](http://www.gk-leika.de/fileadmin/user_upload/gk-leika.de/dokumente/Startseite/News_der_GK_18082010.pdf), 2010. Accessed: 2016-03-30.
2. GK LeiKa. Handbuch: LeiKa-plus. [http://www.gk-leika.de/uploads/media/Handbuch\\_LeiKa-plus\\_Stand\\_27.05.2014.pdf](http://www.gk-leika.de/uploads/media/Handbuch_LeiKa-plus_Stand_27.05.2014.pdf), 2014.
3. C. Gormley and Z. Tong. *Elasticsearch: The Definitive Guide*. O'Reilly Media, Inc., 2015.
4. P. Gupta and V. Gupta. A survey of text question answering techniques. *Intl. Journal of Computer Applications*, vol. 53, 2012.
5. D. Jurafsky and J. H. Martin. Speech and language processing. Chapter 28: Question answering. <https://web.stanford.edu/~jurafsky/slp3/28.pdf>, 2015.
6. N. Konstantinova and C. Orasan. Interactive question answering. <http://pers-www.wlv.ac.uk/~in0988/documents/iqa-chapter-final-21-07-2011.pdf>, 2011.
7. Projekt D115. Leitfaden: D115-informationsbereitstellung. [http://www.115.de/SharedDocs/Publikationen/DE/Spezifikation\\_Leitfaeden/leitfaden\\_d115\\_informationsbereitstellung.pdf?\\_\\_blob=publicationFile&v=1](http://www.115.de/SharedDocs/Publikationen/DE/Spezifikation_Leitfaeden/leitfaden_d115_informationsbereitstellung.pdf?__blob=publicationFile&v=1), 2011.
8. U. Quasthoff, M. Richter, and C. Biemann. Corpus portal for search in monolingual corpora. In *Procs. of the 5th Intl. Conf. on Lang. Resources and Eval.*, volume 17991802, 2006.
9. K. Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.