A Flexible Question Answering System for Mobile Devices

Ortiz-Arroyo, Daniel

Published in:
ICDIM 2008, Third International Conference on Digital Information Management

DOI (link to publication from Publisher):
10.1109/ICDIM.2008.4746794

Publication date:
2008

Document Version
Early version, also known as pre-print

Flexible Question Answering System for Mobile Devices

Daniel Ortiz-Arroyo
Department of Electronic Systems
Aalborg University
Denmark
do@cs.aaue.dk

Abstract

This paper presents a Flexible Question Answering System (FQAS) for mobile wireless devices. FQAS comprises a fuzzy logic-based Information Retrieval (IR) System together with a question answering system that employs information extraction techniques and passage retrieval mechanisms. The IR system includes techniques such as fuzzy term mining and concept clustering. The information extraction module recognizes and extracts relevant information from tables embedded in web pages at specialized web sites. The passage retrieval system determines the passage in a document collection that most likely contains the answer to a question. Our current prototype is capable of answering queries and questions within the chosen football (soccer) test domain.

1. Introduction

Mobile devices provide unprecedented flexibility in information access. Users of mobile devices are capable of getting online information from diverse service providers virtually from any location, at any time. Information is made available in a variety of formats and sources, ranging from specialized data bases and internet news to standard web pages. A popular information media is Really Simple Syndication (RSS) feeds. These feeds supply quick and easy syndication of news and headlines. Users subscribed to these services get directly on their mobile devices, the latest updated news about a variety of subjects. A configurable RSS feed reader in the mobile device periodically retrieves syndicated information from a server. Podcasting uses a similar technique to provide subscribed users with audio or video files from radio programs or music videos automatically.

Personal Digital Assistants (PDAs) or mobile phones have limited input/output capabilities that make accessing web search engine services difficult. The 12-key keypad (or pen) together with the small screen normally provided with these devices make difficult for a user to manage the abundance of hits that are commonly produced by standard search engines. Moreover, many users are interested not only in searching for information but in knowing the exact answer to a query and they get easily discouraged by the many documents they usually have to browse to satisfy their information needs. For these reasons, recent research has investigated the application of techniques aimed at providing spoken query answering capabilities on mobile devices [5], [6].

In this paper we describe an information retrieval system with query answering capabilities. Our system processes queries and provides the user with either, the direct answer to a question or the passage in a document that most likely contains the answer. Our *Flexible Question Answering System* (FQAS) works together with a Distributed Speech Recognition (DSR) system where users can pose spoken queries. However, in this paper we will mainly describe the FQAS module. The DSR system is described in detail in [2].

The paper is organized as follows. Section II, discusses previous related work on information retrieval and question answering systems for mobile devices. The system's architecture is presented in Section III. Section IV describes the FQAS, providing detailed information about its main components. Finally, in Section V, we present some conclusions and future work.

2 Related Work

Chang et al. in [5] describe a spoken query system that retrieves textual information from a database over a PDA. The system employs some of the linguistic features of the Chinese language to improve retrieval precision. Speech retrieval of broadcast news via mobile devices is presented in [6]. The system provides speech recognition and the user information need is satisfied by processing natural language queries.

978-1-4244-2917-2/08/$25.00 ©2008 IEEE

A more recent project, called SmartWeb [1], addressed the use of mobile devices with multi-modal access to the semantic web. To cope with the very limited amount of information available in the current semantic web, this project is also investigating the use of language technology and information extraction (IE) techniques for the automatic semantic annotation of standard web pages.

Buyukkokten et al. in [3] describe techniques to interact with the WWW through mobile devices. A proxy server handles the interaction with the WWW and performs computing intensive tasks such as deeply indexing the web sites frequently visited by a user. This feature provides efficient accessing to these web sites each time they are visited. Additionally, the local indexing capabilities of the retrieval system allow automatic keyword completion of user queries. Finally, the Power Browser described in [4] allows navigating web sites without having to view the full pages on the small screen of a mobile device.

The system presented in this paper has some similarities with some of the approaches previously described. Our system consists of a knowledge-based IR proxy server that contains the FQAS, similarly as it is done in [5]. However, in addition to information retrieval (IR), we provide also question answering (QA) services through passage retrieval and Information Extraction (IE) techniques. The IR system applies a novel combination of fuzzy logic based techniques to retrieve a small number of the most relevant documents (news articles) to a query. To our knowledge the combination of all these techniques has not been explored before within the context of an information retrieval system.

The system's interface as experienced by the end-user consists of a special Graphical User Interface (GUI) specially tailored for mobile devices that shows either, the focused part of a document that most likely contains the answer to a query or the direct answer to a question.

The next sections describe in detail the main subsystems of our FQAS.

3 System Architecture

The flexible question answering system design employs a fully distributed architecture that includes a specialized server containing the FQAS. The overall architecture of the FQAS is depicted in Figure 1.

The logic module in the FQAS server receives the user's question in text form and determines whether the question should be sent to the IR system or if it can be answered directly by the QA system. The server logic also handles data and profile information of users registered in the system.

The FQAS server has been implemented in Java and C# using web services protocols. The client application running on the mobile device was implemented using C# and ad-hoc protocols that allow communication with the speech recognition server [2] and the FQAS server.

4 The FQAS Components

The crawler shown in figure 1 periodically retrieves information from specialized sports related web sites. Retrieved web pages are stored in the Database (DB) module where they can be accessed by the information extraction module. The RSS feed reader client module retrieves periodically, updated news from diverse sport news service providers. The documents retrieved by the RSS feed reader are also stored in the DB. The DB stores both full documents in the XML format and tables containing the information extracted from HTML documents. The DB is also used to store the user's preference profile that is utilized to improve the precision of the FQAS.

The document retrieval system and the question answering system are described in detail in the following subsections.

4.1 The Information Retrieval System

As it is done in many IR systems, the document collection is pre-processed to reduce the set of representative keywords by applying stemming and stop-word removal. A stop word list and a stemmer for the Danish language were used for this purpose. Queries are also pre-processed using the same techniques and then sent to the IR engine.

The IR engine employs a model consisting of the user's context representation, a fuzzy term net [11] and concept clusters created using fuzzy aggregation operators [10]. The user's context representation consists of the historical list of queries, the user's preferences, and the historical list of documents that the user has viewed in previous queries. This context is used by the IR system as supplementary knowledge to reduce the uncertainty in the semantic interpretation of the query. A fuzzy term net is created (before the user enters any query) from the document corpora obtained from the diverse RSS feed sport news providers. A fuzzy term net depicts a binary fuzzy relation about a domain. The term net essentially contains term associations mined from the corpora through the fuzzy data mining technique described in [9].

The indexing system consists of an inverted file data structure containing terms obtained by a technique based on concept clusters [8]. One advantage of using concept clustering is that it allows reducing index term dimension as only the terms belonging to the representative concept clusters are used during indexing. The steps proposed in [8] to create the concept clusters comprise: discovery of lexical chains in the document, identification of the representative concept clusters, and extraction of the indexing terms.

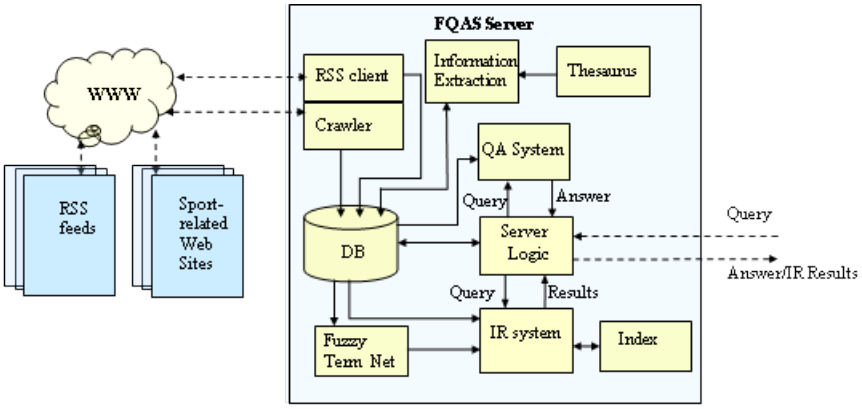

Figure 1. Main Components of the Flexible Question Answering System

The lexical chains in the document are modeled by a fuzzy term net. To create the term net, the internal structure of the documents (e.g. title, abstract, source) is parsed first. Terms coming from these documents sections are assigned different weights according to its importance. Representative concept clusters are identified by calculating the overall weight of each fuzzy term net. The whole process of concept cluster creation is described in a more formal way in the following paragraphs.

Let $N = \{n_1, n_2, .., n_i\}$ be the set of terms (most likely to be nouns) in a document, $S\{n_i, n_j\} = s_{ij}$ be the association relation between terms (coming from the fuzzy term net), where s_{ij} represents the strength of the association (calculated using the method described in [9]), and $C = \{c_1, c_2, .., c_i\}$ a set of concept clusters. A concept cluster is composed of all n_i and n_j term combinations where $s_{ij} > \epsilon$, (ϵ being a threshold value). The score $W_{Term}(t_i)$ assigned to a term t_i is:

$$W_{term}(t_i) = Occ_i + \sum_{j=1, j\neq i}^{k} s_{ij}$$

where Occ_i is the number of occurrences of term t_i in a document. Each score is normalized by calculating the maximum score over all clusters. Additionally, terms coming from important parts of the text (e.g title) are assigned a higher score by applying a factor ζ in $W_{Term}(t_i)^{\zeta}$. Using the normalized score of terms $W^{norm}_{Term}(t_i)$, we calculate the score of a concept cluster c_i in a document as:

$$W_{Concept}(c_i) = \sum_{i=1}^{k} (W^{norm}_{term}(t_i))^{\zeta}$$

being k the number of different terms in the concepts. A large value of $W_{Concept}(c_i)$ indicates that concept c_i is important indeed. The concepts whose value $W_{Concept}(c_i) > \frac{\alpha}{m} \sum_{j=1}^{m} W_{Concept}(c_i)$, are selected as the most representatives and used for indexing the document. α is a parameter used to control the number of concepts employed and m, the number of documents.

The user's query is enhanced with different sources of knowledge with the goal of improving the prediction on the user's information needs. Query enhancing is performed by using global and local analysis. In global analysis each term is expanded through a fuzzy term net with the goal of finding documents that contain the same concepts although may use different terms. The goal of this technique is enabling the model to find the documents that contain the same concepts introduced by the user in a query. Local analysis expands the query automatically by pseudo feedback using concepts mined from the user's profile. The method used in local and global analysis is the same as the one used for indexing of documents, namely the creation of concept clusters. However, in this case, concepts are created amongst the top-ranked documents and terms coming from representative concepts are used for query expansion (including term re-weighting). As a result of local and global analysis, concepts are identified within the query Q and the user's profile P. The terms coming from Q and P have a weight associated $w_Q \in [0, 1], w_P \in [0, 1]$, respectively. The satisfaction score of a document's content compared the user needs is calculated as the aggregation of the satisfaction scores of each of these concept clusters at a level of andness $\rho \geq 0.5$. To perform the aggregation we selected Andness-directed Importance Weighted Averaging (AIWA) [10] and Andness-directed Averaging (AA) operators. AA operators are essentially AIWA operators where importance weights have the value of 1. The satisfaction score of each concept is obtained using an AIWA operator, where the importance weight is made of w_Q or w_P weights. The andness degree of the AIWA operators is $\varepsilon \leq 0.5$. Therefore, the satisfaction score of the user's needs is calculated as:

$$Sat(Needs|doc) = AA_{\rho \geq 0.5}(a_{C_1}, a_{C_2}, .., a_{C_n})$$

where $a_{C_i} = AIWA_{\varepsilon \leq 0.5}^{(v_1,..,v_n)}(a_1, .., a_m)$ and $v_j = w_Q$ or w_P being $a_j = \oplus_{i=1}^{p}(d_i \otimes s_{ij})$ the satisfaction of term $t_j \in T$ and $\otimes, \oplus$ are the T-norm and T-conorm respectively. The satisfaction score of the user's needs can be modified by a weight w to account for user's preference as in $Sat(Needs|doc)^w$, with $w \in [0, 1]$.

Retrieved documents are ordered according to the relevance to the user's query. Moreover, to improve the system's response to the user, only the top ranked document's text is sent from the IR server to the client, jointly with the title of other relevant documents. If the user selects a new document on the GUI, the document's text is retrieved directly from the server.

To display the most likely answer to a user's query, we used a binary search algorithm to locate the passage in the document that is most relevant to the query. The algorithm divides initially the answering document into two sections. Then, the query's satisfaction is computed in each half of the document. The division process is repeated until a minimum acceptable window size is reached. The weights assigned to the terms in the document shown to the user are calculated using a normalized term frequency weighting scheme. Finally, to improve the system's response to the user, the text of the most relevant document is sent from the FQAS server to the client, jointly with the title of the other documents considered relevant. When a user selects a new document on the GUI, the document's text is retrieved from the server. The sequence of documents reviewed by the user is also sent to the FQAS server and stored in the historical user's profile.

Finally, to improve the system's response to the user, the text of the most relevant document is sent from the FQAS server to the client, jointly with the title of the other documents considered relevant. When a user selects a new document on the GUI, the document's text is retrieved from the server. The sequence of documents reviewed by the user is also sent to the FQAS server and stored in the historical user's profile.

We performed a preliminary evaluation of the IR engine comparing its performance with Lucene, a popular open source IR engine based on the vector space model. Lucene provides several searching types: lexical proximity search, word occurrence proximity search, wildcards etc. The IR efficiency was measured by an importance weighted and-like aggregation of the recall (r) and precision (p), namely $r^{w_r} \cdot p^{w_p}$ where w_r and w_p ($w_r, w_p \in [0, 1], max(w_r, w_p) = 1$) are the weights that represent the importance of getting high recall and high precision, respectively. In this case we gave higher importance to precision since getting a few documents containing the answer, and only such documents, is more important than getting all possible documents that may contain the answer. Our preliminary experiments show that the IR system provides a minimum of 5% of improvement over Lucene.

4.2 The Question Answering System

A special multithreaded crawler was designed to provide the question answering system with data retrieved from diverse web sites. The crawler's performs a breadth first search strategy when retrieving web pages within a site. Pages are downloaded and parsed to extract the hyperlinks contained in them. These hyperlinks are placed in a circular queue that each of the crawling threads can access to retrieve new web pages. Since relevant information about the football soccer domain is usually stored in HTML tables, the information extraction techniques employed are targeted to this type of data format. The crawler retrieves information from the web pages published at some specific sport-related web sites. The relevant information about the football soccer domain consists of players' names, matches, clubs, tournaments, etc. To extract that information, firstly valid HTML tables must be identified and filtered. To perform this task, we used a modified version of the algorithm proposed in [7]. The algorithm first filters out tables containing a single cell. Then, it checks that the content enclosed by the table wrappers does not contain too many hyperlinks, forms, or figures, as this indicates that the table probably does not contain data but it is used as a formatting element. Our modified algorithm, contrarily to [7], does not apply string similarity and named entity similarity measures , since these measures do not have applicability in the football soccer domain. Additionally, we also modified the original algorithm to use a normalized image capacity value, remove cells with irrelevant content during table comparison, and filter out tables contained within tables. The performance obtained by our crawler, after the filtering step, had a precision of 96% and recall of 100% with a test collection of 55 documents retrieved from diverse sport related web sites.

Once valid HTML tables have been identified, the process employed to extract information from these tables, consists of two steps. First it is determined whether the table contains information related to our domain of interest (football soccer) or not and then how the information on the table should be interpreted i.e. in a column or row wise manner. To identify tables containing information about football soccer we used a prioritized evaluation of the context where the tables are located within a web page. Priority was given to the following table context features: table caption, identification of the row closest to the top of the table, keywords found in the text surrounding the table, and keywords found in the context of the table preceding the current one. Four

different types of terms were employed in the prioritization process used to identify soccer related tables: tournament names, locations, seasons, and team names. Additionally, some tournament names and locations for the Danish league were introduced manually into the database. With this information, the crawler was able to automatically fill in the data base with the values corresponding to the other two terms considered in our system, namely seasons and team names.

To learn whether a valid table should be interpreted column wise or row wise we used the algorithm described in [7]. The algorithm essentially measures the similarity between columns and rows and selects an interpretation mode of a table based on how many rows or columns are similar. We measured the performance of our crawler during the identification phase of football soccer tables obtaining a recall of 100% and a precision of 87% on an experiment that included a collection of documents containing a total of 316 tables. Once valid soccer related tables have been identified, these tables are processed to extract the relevant information into a data base. The relevant information extracted from the tables for our selected domain is: tournament standings, match scores, goal scores, players information, club facts, etc. A thesaurus was employed to identify variations in names that occur frequently in the documents. The variations that were entered manually in the thesaurus are teams' names, tournaments and player names. Furthermore, to recognize abbreviations in names (e.g Esbjerg Football Club also known as EFC) that are commonly employed in tables, we performed a matching test using regular expressions of the names retrieved against those stored in the thesaurus. Finally, as spelling errors are commonly made by authors of web pages we accepted as valid, names and terms within a Levenshtein (or edit) distance of no more than one from the corresponding names stored in the thesaurus.

The question answering system employs an ontology constructed using an Entity-Relationship Model (ERM) to represent the knowledge extracted from HTML tables about the football soccer domain. The ERM was mapped into the DB tables managed by MySQL DBMS.

The QA system recognizes some typical question formats about football soccer using regular expressions. This simple approach is effective due to the limited number of formats considered and the closed domain targeted by our current prototype. The question formats were obtained from a poll performed among groups of users interested in soccer. Users were asked what type of information was more interesting to them and what kind of questions they would be interested in submitting to the FQAS. A question whose format is recognized by the question answering system, is converted into a sequence of SQL commands to obtain the direct answer from the database. Finally, it is worth mentioning that the question answering system also supplies relevant supplementing information to the direct answer; for instance, when a user asks "*who won the match between X and Y*", the direct answer is supplied by the match results, the names of the players who scored and the time they scored, plus the date, time and location where the match took place. Some examples of the types of questions that our rule grammar-based speech recognizer and QA system are able to recognize are:

Who is $\langle team \rangle$ next opponent?

Who won $\langle tournament \rangle$ in $\langle season \rangle$?

How many goals did $\langle player \rangle$ score in $\langle tournament \rangle | \langle time \rangle$?

Which $(team|player)$ has made the $(most|least)$ goals in $\langle league \rangle$?

(What was the result | who won) (the match | game) between $\langle team \rangle$ and $\langle team \rangle$ at [$\langle time \rangle | \langle date \rangle$]

Who (is | plays) $\langle position \rangle$ (in | for) $\langle team \rangle$?

What is (the name of) $\langle team \rangle$ home court?

How many (victories | defeats | draws) (did $\langle team \rangle$ have | have $\langle team \rangle$ had) (in $\langle season \rangle$ | the last $\langle number \rangle$ matches)

A total of 150 different questions were tested in the QA system. Some examples of the questions and answers (translated from Danish) that we obtained during our tests are:

Question 1: *Who is playing for Esbjerg?*

Answer: a list of players' names.

Question 2: *Who won the match between Odense and Viborg?*

Answer: The match was played on 24-04-2005 at 15:00:00. The match ended in a tie (0-0).

Question 3: *Who won the match between Esbjerg FB and Viborg?*

Answer: Match was played on March 27 2005 at 16:OO.

Esbjerg won the match 2-1.

F. Berglund scored at minute 21. J. Kristiansen scored at minute 42. Mota scored at minute 82.

5 Conclusions and Future Work

This paper presented a mobile information service system with spoken query answering capabilities. Our system adopts a fully distributed architecture with a thin mobile client and a proxy server system composed of speech recognition (described in [2]) and Flexible Question Answering System (FQAS) servers. Most of the demanding computing processing is performed on these dedicated servers. The FQAS, employs a novel combination of diverse techniques such as fuzzy term nets, concept clustering and information extraction. Our current prototype is capable of either answering directly questions or showing on the screen the document passage that most likely contains the answer to

a query. The question answering system employs information extracted from valid relevant HTML tables retrieved from sports related web sites to answer questions directly. The integrated information system has the capability of answering Danish spoken queries over a PDA in the domain of football (soccer). Although this system may be ported to other natural languages than Danish and other domains than football, its practical application is restricted to limited domains (like soccer), due to the manual construction of the domain ontology and the use of the common question formats applied in the domain. We plan to extend our current prototype to include support for the English language and to allow users to retrieve other media such as audio and video clips.

As is described in Section 4.1, our experiments show that the performance of the IR system is sensitive to the value of the multiple parameters used to regulate its operation. These parameters were given values that we thought were the most appropriated. To improve the preliminary performance results obtained by our current prototype we plan to apply optimization techniques such as genetic algorithms or multi-objective evolutionary computing aimed at finding the optimal values.

Finally, we are currently experimenting with the application of fuzzy logic in a mechanism designed to improve the precision of the information retrieval system when answering queries not recognized by the QA system.

6 Acknowledgments

The author want to thank Tom Brønsted, Lars Bo Larsen and Zheng-Hua Tan for providing the DSR system. We also acknowledge the important contribution of Henrik Legind Larsen, one of the project leaders and author of some of the key fuzzy logic based techniques applied in the prototype. Finally, the author wants to thank Marthe Buffire, Henrik Mathiassen, Niels Nygaard Nielsen, Allan Pedersen, and Frederic Pichon for developing most of the modules described in the paper. This research was supported by the Center of TeleInFrastruktur (CTIF), Aalborg University.

References

[1] Smartweb: Mobile broadband access to the semantic web. http://www.smartweb-project.org.

[2] T. Brøndsted, L. B. Larsen, B. Lindberg, M. Rasmussen, Z. Tan, and H. Xu. Distributed speech recognition for information retrieval on mobile devices. In *Workshop on Speech in Mobile and Pervasive Environments, Espoo 2006*, 2006.

[3] O. Buyukkokten, H. Garcia-Molina, and A. Paepcke. Focused web searching with pdas. In *Proc. 9th International WWW Conference (WWW9), Amsterdam, Netherlands - May 15-19, 2000*, 2000.

[4] O. Buyukkokten, H. Garcia-Molina, A. Paepcke, and T. Winograd. Power browser: Efficient web browsing for pdas. In *CHI 2000 Human-Computer Interaction Conference 2000*, 2000.

[5] E. Chang, F. Seide, and H. Meng. A system for spoken query information retrieval on mobile devices. *IEEE Trans. Speech and Audio Proc.*, 10(8):531–541, 2002.

[6] B. Chen, Y. Chen, and C. Chang. Speech retrieval of mandarin broadcast news via mobile devices. In *Interspeech 2005, Lisbon, Portugal, Sep. 2005*, 2005.

[7] H. Chen, S. Tsai, and J. Tsai. Mining tables from large scale html texts. In *Proc. 18th International Conference on Computational Linguistics, Saabrucken, Germany, July 2000*, 2000.

[8] B. Kang, Y. Kim, and S. Lee. Exploiting concept clusters for content based information retrieval. *Information sciences*, 170:443–462, 2005.

[9] H. L. Larsen. Fuzzy data mining of term associations for flexible query answering. In *Proc. International Conference in Fuzzy Logic and Technology Leicester, England, 5-7 September 2001 (EUSFLAT'2001)*, 2001.

[10] H. L. Larsen. Efficient andness-directed importance weighted averaging operators. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 11(Supplement-1):67–82, 2003.

[11] H. L. Larsen and R. R. Yager. The use of fuzzy relational thesauri for classificatory problem solving in information retrieval and expert systems. *IEEE J. on System, Man, and Cybernetics*, 23(1):31–41, 1993.