Journal of King Saud University – Computer and Information Sciences (2016) **28**, 345–361

King Saud University

Journal of King Saud University – Computer and Information Sciences

www.ksu.edu.sa
www.sciencedirect.com

A survey on question answering systems with classification

Amit Mishra *, **Sanjay Kumar Jain** [1]

Computer Engineering Department, NIT Kurukshetra, Haryana 136119, India

Received 25 January 2014; revised 9 May 2014; accepted 23 October 2014
Available online 2 November 2015

KEYWORDS

Question answering system;
Information retrieval;
Natural language processing;
Natural language understanding;
Search engine

Abstract Question answering systems (QASs) generate answers of questions asked in natural languages. Early QASs were developed for restricted domains and have limited capabilities. Current QASs focus on types of questions generally asked by users, characteristics of data sources consulted, and forms of correct answers generated. Research in the area of QASs began in 1960s and since then, a large number of QASs have been developed. To identify the future scope of research in this area, the need of a comprehensive survey on QASs arises naturally. This paper surveys QASs and classifies them based on different criteria. We identify the current status of the research in the each category of QASs, and suggest future scope of the research.

Contents

1. Introduction . . . 346
2. Related work . . . 346
3. Criteria for classifying question answering systems . . . 347
4. A classification of question answering systems . . . 347
   4.1. Classification based on application domain . . . 347
      4.1.1. General domain QASs . . . 348
      4.1.2. Restricted domain QASs . . . 348
   4.2. Classification based on types of questions . . . 351
      4.2.1. Factoid type questions [what, when, which, who, how] . . . 351

* Corresponding author. Tel.: +91 9355782052.
E-mail addresses: amitmishrag@gmail.com (A. Mishra), skj_nith@yahoo.com (S.K. Jain).
[1] Tel.: +91 9996127295.
Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jksuci.2014.10.007

4.2.2. List type questions . . . 351
4.2.3. Hypothetical type questions . . . 351
4.2.4. Causal questions [how or why] . . . 352
4.2.5. Confirmation questions . . . 352
4.3. Classification based on types of analysis done on questions . . . 352
4.3.1. Morphological analysis . . . 353
4.3.2. Syntactical analysis . . . 353
4.3.3. Semantic analysis . . . 353
4.3.4. Pragmatic and discourse analysis . . . 353
4.3.5. Expected answer type analysis . . . 354
4.3.6. Focus recognition of questions . . . 354
4.4. Classification based on types of data sources . . . 354
4.4.1. Structured data source . . . 354
4.4.2. Semi-structured data source . . . 355
4.4.3. Un-structured data source . . . 355
4.5. Classification based on types of matching functions used in different retrieval models . . . 355
4.5.1. Set theoretic models . . . 355
4.5.2. Algebraic models . . . 355
4.5.3. Probability models . . . 355
4.5.4. Feature based models . . . 356
4.5.5. Conceptual graph based models . . . 356
4.6. Classification on the basis of characteristic of data sources . . . 356
4.6.1. Source size . . . 356
4.6.2. Language . . . 356
4.6.3. Heterogeneity . . . 356
4.6.4. Genre . . . 357
4.6.5. Media . . . 357
4.7. Techniques used in QASs . . . 357
4.8. Classification based on forms of answer generated by QASs . . . 358
4.8.1. Extracted answer . . . 358
4.8.2. Generated answer . . . 358
5. Comparison with other classifications . . . 358
6. Conclusions . . . 359
References . . . 359

1. Introduction

Search engines present a ranked list of relevant documents in response to users' formulated keywords based on various aspects such as popularity measures, keyword matching, frequencies of accessing documents, etc. However, they do not truly accomplish the task of information retrieval as users have to examine each document one by one for getting the desired information (Ferret et al., 2001); it makes information retrieval a time consuming process. Ideally, a search engine should return few relevant and concise sentences as answers along with their corresponding web links. A large number of QASs have been developed since 1960's (Androutsopoulos et al., 1995; Kolomiyets, 2011). Current QASs attempt to answer questions asked by users in natural languages after retrieving and processing information from different data sources even like semantic web (Vanessa, 2011; Dwivedi, 2013; Suresh kumar and Zayaraz, 2014). The format of answers is also going to be changed from simple text to multimedia (Voorhees and Weishedel, 2000). QASs developed since 1960s address different domains, data sources, types of questions, formats of answers, etc.; the number of such QASs is too large. To assess the success of these QASs and their ability to satisfy current and future needs, a systematic survey of all these QASs becomes necessary.

In this paper, we classify QASs based on explicitly identified criteria like application domains, questions, data sources, matching functions, and answers. We make a survey of the literature on QASs classified on each criterion and identify future scope of research in this area.

The rest of of this paper is organized as follows: section 2 presents related work on QASs beginning from early days of research in the form Natural Language Interface to Databases (NLIDB) to open domain QASs over text. Section 3 presents criteria identified for classification of QASs. Section 4 deals with classification of QASs based on different criteria. Section 5 makes a comparison of the proposed classification with others. In section 6, we draw conclusions.

2. Related work

In this section, we present a background on development of QASs since 1960's to the present time. The plan of developing systems that can deal with natural language questions began in the fifth generation of computer programing language (Hill I 1982). NLIDB is a system that provides facility to users for asking questions in their natural languages for getting information from databases (Androutsopoulos et al., 1995). It eases human computer interaction as users need not to learn formal languages such as SQL, Prolog,