Methods 74 (2015) 36–46

Contents lists available at ScienceDirect

Methods

journal homepage: www.elsevier.com/locate/ymeth

Question answering for Biology

Mariana Neves [a,*], Ulf Leser [b]

[a] Hasso-Plattner-Institut, Potsdam Universität, Potsdam, Germany
[b] Humboldt-Universität zu Berlin, Knowledge Management in Bioinformatics, Berlin, Germany

ARTICLE INFO

Article history:
Received 28 April 2014
Received in revised form 3 September 2014
Accepted 21 October 2014
Available online 28 October 2014

Keywords:
Question answering
Biomedicine
Natural language processing
Data integration

ABSTRACT

Biologists often pose queries to search engines and biological databases to obtain answers related to ongoing experiments. This is known to be a time consuming, and sometimes frustrating, task in which more than one query is posed and many databases are consulted to come to possible answers for a single fact. Question answering comes as an alternative to this process by allowing queries to be posed as questions, by integrating various resources of different nature and by returning an exact answer to the user. We have surveyed the current solutions on question answering for Biology, present an overview on the methods which are usually employed and give insights on how to boost performance of systems in this domain.

1. Introduction

When planning or analyzing experiments, scientists look for related and previous findings in the literature to obtain external evidence on current observations. For instance, biologists often seek information regarding genes/proteins (biomarkers) expressed in a particular cell or tissue of a particular organism in the scope of a particular disease. Finding published answers to such questions requires dealing with a variety of synonyms for the genes and diseases and posing queries to different databases and search engines. Further, it often also involves screening hundreds of publications or data returned for the queries.

The task of searching for relevant information in a collection of documents, such as web pages using search engines or scientific publications using PubMed[1], is generally called information retrieval (IR) [1]. In IR, queries are usually expressed in terms of some keywords and answering a query does not usually take into account synonyms, i.e., when a certain concepts has more that one name, and homonyms, i.e., when the same name refers to more than one concept. IR systems typically return a list of documents potentially relevant to the query, including related metadata (e.g., journal name and year of publication) and snippets of text matching the query keywords.

In contrast to IR, question answering (QA) [2,3] aims to support finding information in a collection of unstructured and structured data, e.g., texts and databases, respectively. Furthermore, QA systems take questions expressed in natural language (e.g., English) and generate a precise answer by linguistically and semantically processing both the questions and data sources under consideration. In particular, a question answering system distinguishes from IR systems in three main aspects (cf. Table 1): (1) queries can be posed using natural language instead of keywords; (2) results do not consist of passages but are generated according to what has been specifically requested, be it a single answer or a short summary; (3) answers are based on the integration of data from textual documents as well as from a variety of knowledge sources.

The first aspect aims to facilitate usage for non-IR experts, i.e., users do not need to be concerned on how to best pose a query to receive a precise answer. For instance, when questioning about the participation of a certain gene in a pathway, e.g., the gene p53 in WNT-signaling cascade, users would usually write both terms in the search field of a search engine. In case of not finding the answer in any of the top list documents, the user could consider entering synonyms for both the gene (e.g., "TP53") and the pathway (e.g., "WNT signaling pathway"). In order to cope with this problem, some IR systems allow the use of ontological terms instead of keywords for a more precise retrieval of relevant documents. For instance, GoPubmed[2] automatically suggests candidates terms in the Medical Subject Headings (MeSH) and Gene Ontology during keywords typing. However, understanding of ontological concepts is not straightforward for scientists not familiar to them. The use of natural

* Corresponding author at: August-Bebel-Str. 88 Potsdam, Brandenburg 14482, Germany. Fax: +49 (0)331 5509 579.
E-mail address: marianalaraneves@gmail.com (M. Neves).

[1] http://www.ncbi.nlm.nih.gov/pubmed.

[2] http://gopubmed.org.

http://dx.doi.org/10.1016/j.ymeth.2014.10.023
1046-2023/

Table 1
Main differences between QA and IR systems.

Feature	Question answering	Information retrieval
Query	Question in natural language	Set of keywords and/or concepts
Results	One or more exact answers	List of candidate documents
Answer	Based on multiple documents and resources	Based on passages from multiple documents

language is a more intuitive way to inquire for information, by posing questions (how, what, when, where, which, who, etc.) or requests (show me, tell me, etc.). For instance, for the example above, users could simply write the question "Is p53 part of the WNT-signaling cascade?". Of course, allowance of free questions requires very advanced natural language processing (NLP) techniques.

The second characteristic of QA systems is to provide precise answers instead of only presenting potentially relevant documents. When using IR systems, figuring out the answer to a query requires reading the documents returned by the system. QA systems strive to simply return the answer "No" to the question above along with a list of references that gave evidence for this answer. This requires QA systems to perform a deep linguistic analysis of both the question and the potential relevant passages, also considering the meaning of terms. Not only synonyms, hypernyms and hyponyms must be considered during answer construction, but also disambiguation of entities should be performed whenever necessary, such as figuring out whether the word "WNT" refers to part of the pathway name or to a mention of a member of the WNT gene family.

Third, QA is not limited to textual resources and can include integration of data resources by converting natural language questions to an appropriate query language for searching for answers in databases, for instance in RDF triples [4]. Data extracted from different sources need to be assembled into a coherent single answer by means of exploring interlinks, dealing with contradictions and joining equal or equivalent answers. Currently, conversion of biomedical natural language questions to RDF triples are being evaluated in the BioASQ challenge (cf. Section 3.2.1) and question answering over linked data for three biomedical databases is being assessed in one of the Question Answering for Linked Data (QALD) shared tasks (cf. Section 3.2.4). Further, a prototype of the LODQA system [5] (cf. Section 3.1.2) converts questions to SPARQL queries for submission to BioPortal end point.

The technology behind information systems has evolved from simple Boolean keywords-based queries to complex linguistic processing of both the question and textual passages. Fig. 1 shows an overview on the evolution of these techniques and illustrates the complexity of question answering systems. Many of the current information systems available for querying PubMed implement some of these techniques (cf. survey in [6]).

Question answering has been successful applied for other domains, and examples of such systems are START[3] and Wolfram Alpha[4]. Recent interest in question answering has been also motivated by the IBM's Watson system [7], which beat human participants in a game show. Various researchers advocate that QA system can provide many benefits to the biological domain and they expect that these systems can boost scientific productivity [8]. Indeed, a study carried out with physicians showed that they do trust in the answers provided by QA systems [9]. However, Life Sciences also poses many challenges to QA systems, specially: (1) highly complex and large terminology, (2) exponential growth of data and hundreds of on-line databases, and (3) high degree of contradictions. Often, answering a question not only requires identifying relevant facts in a single document or database, but merging parts of the answers from distinct sources. Nevertheless, research on question answering in Biology is still scarce, in contrast to the medical domain (cf. Section 4).

The first community-based challenge which included a task related to biomedical question answering took place in 2006 and 2007 and consisted in the evaluation of passage retrieval and restricted to topics related to Genomics (cf. TREC Genomics in Section 3.2.3). Later on, in 2012 and 2013, the Question Answering for Machine Reading Evaluation (QA4MRE) Alzheimer Disease challenge assessed systems on the machine reading task, which consists of multiple choice questions related to a single document (cf. Section 3.2.2). Use of RDF in biomedical QA tasks is currently being evaluated in the QALD challenge (cf. Section 3.2.4) and the more comprehensive challenge related to biomedical QA so far, BioASQ (cf. Section 3.2.1), has been running since last year (2013).

In this work, we present an overview on question answering systems and techniques for the biological domain. In the next section, we give an overview on the most common components of a question answering system. Section 3 describes current systems and results obtained in shared tasks. Section 4 discusses the state-of-art of QA systems for the medical domain and give insights on which improvements could be achieved in biological field in the near future.

2. Fundamental techniques

Question answering systems are usually composed of three steps [10,11] (Fig. 2): question processing; candidate processing, and answer processing. The first step receives the input entered by the user, i.e., a natural language question, and includes preprocessing of the question, identification of the question type and the type of answer to be required (e.g., the entity type) and building an input to the next step. In the candidate processing step, relevant documents, passages or raw data are retrieved and ranked according to their relevance to the question. Finally, the answer processing step receives the retrieved text passages and data items and builds the final answer by extracting and merging information from different sources. In general, more techniques and reusable software components are available for first two steps based on years of research work on NLP and IR. On the other hand, the last one requires some techniques specific of QA, which is a rather recent field in comparison to NLP and IR, and might require dealing with data of different nature and sources.

Researchers usually classify questions in three types: yes/no, factoid/list and definitional/summarization question. The yes/no questions are the simplest ones, as the only two possible answers are known beforehand: "yes" or "no". Factoid and list questions expect a single or a list of short facts in return, such a named-entity (e.g., a gene, a disease), or an amount (e.g., number of mutations for a certain gene). The main difference between factoid and list questions is that the first expects a single answer while the second one allows a list of them. Finally, the summary and definition questions expect a summary or short passage in return. For instance, the expected answer for the questions "What is stem cell?" or "How does the mitosis of a cell work?" should be a short text summary, e.g., up to 10 sentences. Although this kind of answer might seem similar to the ones provided by traditional IR systems, in QA systems, summaries are automatically constructed specifically for the query in hand and may contain text extracts from different documents or data sources. Further, as opposed to text snippet returned by IR system, these summaries can be viewed as a single answer instead of passages derived from hundreds of documents with inconsistent definitions.

In this section we present a detailed description on the methods employed in these steps along with practical examples based on the

[3] http://start.csail.mit.edu/.
[4] http://www.wolframalpha.com/.