DOI:10.1145/1378727.1378743

Beyond Google, emerging question-answering systems respond to natural-language queries.

BY DMITRI ROUSSINOV, WEIGUO FAN, AND JOSÉ ROBLES-FLORES

Beyond Keywords: Automated Question Answering on the Web

SINCE THE TYPICAL COMPUTER USER spends half an hour a day searching the Web through Google and other search portals, it is not surprising that Google and other sellers of online advertising have surpassed the revenue of their non-online competitors, including radio and TV networks. The success of Google stock, as well as the stock of other search-portal companies, has prompted investors and IT practitioners alike to want to know what's next in the search world.

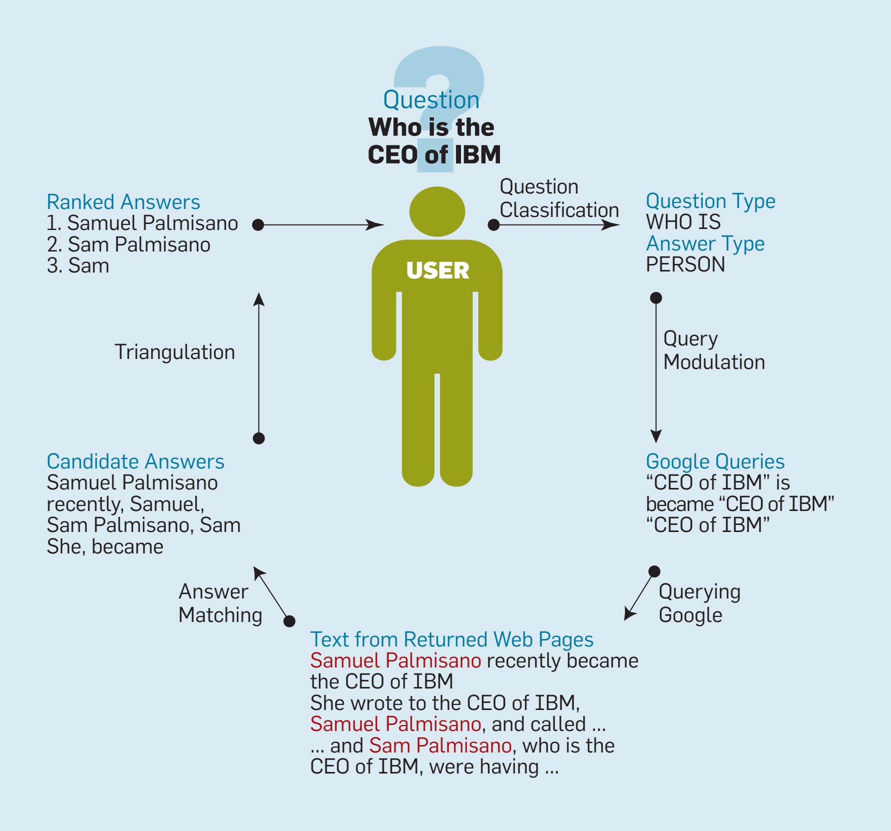
The July 2005 acquisition of AskJeeves (now known as Ask.com) by InterActiveCorp for a surprisingly high price of $2.3 billion may point to some possible answers. Ask.com not only wanted a share of the online-search market, it also wanted the market's most prized possession: completely automated open-domain question answering (QA) on the Web, the holy grail of information access. The QA goal is to locate, extract, and provide specific answers to user questions expressed in natural language. A QA system takes input (such as "How many Kurds live in Turkey?") and provides output (such as "About 15 million Kurds live in Turkey," or simply "15 million").

Search engines have significantly improved their ability to find the most popular and lexically related pages to a given query by performing link analysis and counting the number of query words. However, search engines are not designed to deal with natural-language questions, treating most of them as "bags," or unordered sets, of words. When a user types a question (such as "Who is the largest producer of software?"), Google treats it as if the user typed "software producer largest," leading to unexpected and often not-useful results. It displays pages about the largest producers of dairy products, trucks, and "catholic software," but not the answer the user might expect or need (such as "Microsoft"). Even if the correct answer is among the search results, it still takes time to sift through all the returned results and locate the most promising answer among them.

It is more natural for people to type a question (such as "Who wrote King Lear?") than to formulate queries using Boolean logic (such as "wrote OR written OR author AND King Lear"). Precise, timely, and factual answers are especially important when dealing with a limited communication channel. A growing number of Internet users have mobile devices with small screens (such as Internet-enabled cell phones). Military, first-responder, and security systems frequently put their users under such time constraints that each additional second spent browsing search results could put human lives at risk. Finally, visually impaired computer us-

ILLUSTRATION BY JON HAN

Architecture of a typical Web QA system.



ers simply cannot enjoy the quantity of information available on the Web, since they are unable to glance through the pages of snippets that are returned by search engines. The best available reader software and refreshable Braille screens do not provide enough bandwidth for real-time interaction.

Although Google and Microsoft have announced that they've added QA features to their engines, these capabilities are limited, as we found in the simple experiment we report here and reconfirmed at the time of publication. Since many practitioners are familiar with the concept of online QA, we review only the recent advances in automated open-domain (Web) QA and the challenges faced by QA. We contrast the most noticeable (in terms of academic research interest and media attention) systems available on the Web and compare their performance, as a "team," against two leading search portals: Google.com and MSN.com.

Technology Foundation

For the past decade, the driving force behind many QA advances has been the annual competition-like Text Retrieval Conference (TREC).[8] The participating systems must identify precise answers to factual questions (such as "who," "when," and "where"), list questions (such as "What countries produce avocados?"), and definitions (such as "What is bulimia?").

The following distinctions separate QA from a fixed corpus (also called "closed domain," as in TREC competitions) and QA from the entire Web (typically referred to as "open corpus" or open-domain QA):

Existence of simpler variants. The Web typically involves many possible ways for answers to begin, allowing QA fact-seeking systems to look for the lowest-hanging fruit, or most simple statements of facts, making the task easier at times;

Expectation of context. Users of Web-based fact-seeking engines do not necessary need answers extracted precisely. In fact, we've personally observed from our interaction with practitioners (recruited from among our MBA students) that they prefer answers in context to help verify that they are not spurious; and

Speed. Web-based fact-seeking engines must be quick, and TREC competition does not impose real-time constraints. This emphasizes simple, computationally efficient algorithms and implementations (such as simple pattern matching vs. "deep" linguistic analysis).

A typical Web QA system architecture is illustrated by the NSIR system (see the Figure here),[5] one of the earliest Web QA systems (1999–2005) developed at the University of Michigan, and the more recent Arizona State University Question Answering system (ASU QA).[6] When given a natural-language question (such as "Who is the largest producer of software?"), the system recognizes a certain grammatical category (such as "what is," "who is," and "where was"), as well as the semantic category of the expected answer ("organization" in this example). NSIR uses machine-learning techniques and a trainable classifier to look for specific words in the questions (such as "when" and "where"), as well as parts of speech (POS) of the other words supplied by the well-known Brill's POS tagger.[2] For example, in the question "What ocean did Titanic sink in?," the tagger identifies "ocean" as a noun and "sink" as a verb. The trained classifier classifies the expected answer type as "location."

ASU QA matches the question to one of the trained regular expressions. For example, the question "What ocean did Titanic sink in?" matches "What <C> did <T> <V>," where <C> is any word that becomes the expected semantic category ("ocean"), <T> is the word or phrase that becomes the question target ("Titanic"), and <V> is the verb phrase ("sink in"). While NSIR and ASU QA use only a few grammatical and semantic categories, some other (non-Web) systems involve more fine-tuned taxonomies. For example, Falcon,[7] one of the most successful TREC systems, is based on a pre-built hierarchy of dozens of semantic types of expected answers, subdividing the category "person" further into "musician," "politician," "writer," "athlete," and more.

Web QA systems generally do not crawl or index the Web themselves. They typically use the "meta engine" approach: send one or more queries to commercial engines providing application programming interfaces (APIs) specifically designed for this purpose. The query-modulation step in the Figure creates requests for the search engine based on the words in the questions that are sometimes expanded with synonyms. For example, ASU QA takes dozens of patterns for each question type from the questions and answers seen previously during the training process. It transforms the question, say, "Who is the CEO of IBM?" into the Google query "became the CEO of IBM" because it has previously seen the answer "Washington" to the question "What is the capital of the U.S.?" in the sentence "Washington became the capital of the U.S. on June 11, 1800."

Since many of the factual answers are named entities (such as people, organizations, countries, and cities), QA systems typically employ third-party named-entity-identification techniques to extract candidate answers (such as Minipar).[1] All named entities in the proximity of the question words and that match the desired semantic category are identified as candidate answers. Meanwhile, ASU QA, employs a pattern-matching mechanism to perform answer extraction. A sentence like "Samuel Palmisano recently became the CEO of IBM" matches the pattern "<A> became <Q>," where <A> = "Samuel Palmisano recently" is the candidate answer, and <Q> = "the CEO of IBM" is the question's focus. ASU QA also treats all subphrases from each candidate answer as candidates themselves. In the example, the subphrases are "Samuel Palmisano recently," "Samuel Palmisano," "Palmisano recently," "Samuel," "Palmisano," and "recently."

In order to identify the most probable (supported) answer, ASU QA has gone several steps beyond frequency counts explored earlier by Dumais et al.[1] and other groups involved in TREC competitions that involved a probabilistic triangulation mechanism. Triangulation is a term widely used in the intelligence and journalism fields for confirming or disconfirming facts by checking multiple sources. Roussinov's and Robles's algorithm is demonstrated through the following intuitive example: Imagine that we have two candidate answers for the question "What was the purpose of the Manhattan Project?": (1) "To develop a nuclear bomb" or (2) "To create an atomic weapon." They support (triangulate) with each other since they are semantically similar. In the example involving the CEO of IBM, "Samuel Palmisano" and "Sam Palmisano" win because they reinforce each other.

Although QA technology is maturing quickly and seems promising for a number of practical applications (such as commonsense reasoning and database federation), few QA systems go beyond information seeking. Although the Ford Motor Company and Nike, Inc. began using Ask.com as their site search engine in 2005, they've never reported if QA features are indeed practical and useful. In 2005, Roussinov and Robles demonstrated empirically that ASU QA helps locate potentially malevolent online content, potentially helping law-enforcement and public oversight groups combat the proliferation of materials that threaten cybersecurity or promote terrorism.

Feature Comparison

When comparing features and performing our informal evaluation, we chose only the QA systems (see Table 1) mentioned in popular IT magazines or academic publications and that were (and still are) available online during the first run of our study in spring 2005. We did not include Google or MSN since their QA capabilities were (and still are) quite limited. Google occasionally produces precise answers with respect to geography-related questions (such as "What is the population of Cambodia?") but does not attempt to target more general or dynamic topics (such as "Who is the CEO of Motorola?") or more grammatically or semantically challenging questions ("How long can a British Prime Minister serve in office?"). MSN uses only Encyclopedia Encarta as a source of precise answers and is similarly limited in terms of complexity and coverage.

Although AskJeeves enjoyed immense popularity and investor interest at the time it was acquired, its QA capabilities are limited in practice. Its answers to natural-language questions could be provided only from manually created databases, and the topics of inquiry were limited to simple "encyclopedic" requests (such as "What is the population of Uganda?"). When the question does not match any of the anticipated questions, Ask.com would reroute the question as a simple keyword query to its underlying keyword search engine—Teoma, which was acquired by Ask.com in 2001 when it was a failing dot-com based on technology originally created by IBM and further developed at Rutgers University. In 2005, Ask.com introduced certain answer-matching capabilities over the entire Web but is still short of specifying the precise answer while displaying a set of ordered snippets (up to 200) with the words from the highlighted question, similar to Google's approach.

Table 1: Features of selected Web (open-domain) QA systems.

System	Purpose	Output Format	Multilingual	Technology/ Algorithms	Crawling
AskJeeves	Commercial	Up to 200 rank-ordered snippets	Yes	Undisclosed	Entire Web
BrainBoost	Commercial	Up to 10 snippets or sentences	No	Undisclosed	Meta search
Language Computer Demo	Commercial/ research prototype	Up to 10 snippets	No	Deep parsing, theorem proving, large taxonomy of answer types	Meta search
AnswerBus	Commercial/ research prototype	Up to 10 sentences	Yes	Shallow parsing, entity extraction, small taxonomy of answer types	Meta search
NSIR	Research prototype	Exact answers or snippets	No	Shallow parsing, entity extraction, small taxonomy of answer types	Meta search
ASU QA	Research prototype	Up to 20 snippets	No	Pattern matching, small taxonomy of answer types	Meta search

Since BrainBoost (www.brainboost.com) is a commercial system, little is known outside the company about the algorithms it employs. Nevertheless, it quickly gained popularity among bloggers and other online information seekers, since it delivers decent accuracy and quick response. Answers.com bought BrainBoost for $4 million in cash and shares of restricted BrainBoost stock in 2005.

Another prototype developed by Language Computer Corporation (www.languagecomputer.com) returns up to 10 answer snippets, with the words from the question (not the precise answer itself) highlighted. AnswerBus (misshoover.si.umich.edu/~zzheng/qa-new/) and NSIR (tangra.si.umich.edu/clair/NSIR/html/nsir.cgi) were the two earliest open-domain Web QA systems developed in academic institutions, and their algorithms are detailed in a number of publications.[5] Based on matching the question to a trained set of answer patterns, ASU QA uses probabilistic triangulation to capitalize on the redundancy of publicly available information on the Web. Along with BrainBoost, ASU QA was used for several years in a $2 million project supported by NASA (www.aee.odu.edu) aimed at developing collaborative distributed engineering knowledge/information management systems and intelligent synthesis environments for future aerospace and other engineering systems.

Table 2: Comparing search-engine performance: Google and MSN (as a team) vs. selected online QA systems (as a team).

Question	Google MRR	MSN MRR	Average MRR for the Search Portals Team	Average MRR for the QA Team
Aspartame is also called what?	0.00	0.33	0.17	0.29
At what speed does the Earth revolve around the sun?	0.00	0.50	0.25	0.25
At what time of year is air travel at a peak?	0.00	0.00	0.00	0.00
Boxing Day is celebrated on what date?	0.50	1.00	0.75	0.63
CNN is owned by whom?	0.20	0.50	0.35	0.65
How big is our galaxy in diameter?	0.14	0.00	0.07	0.65
How did Al Capone die?	0.00	0.00	0.00	0.21
How did Bob Marley die?	0.17	0.00	0.08	0.51
How far is it from Denver to Aspen?	0.11	0.00	0.06	0.29
How far is Pluto from the sun?	1.00	0.11	0.56	0.75
How long can a British Prime Minister serve in office?	0.00	0.00	0.00	0.13
How many copies of an album must be sold for it to be a gold album?	1.00	0.00	0.50	0.35
How many Stradivarius violins were ever made?	0.00	0.00	0.00	0.38
How many teachers are there in the U.S.?	0.00	0.00	0.00	0.08
How much folic acid should an expectant mother get daily?	0.20	0.11	0.16	0.25
In what country is a stuck-out tongue a friendly greeting?	0.25	0.00	0.13	0.00
What color is a giraffe's tongue?	0.50	1.00	0.75	0.63
What continent is Argentina on?	0.33	0.00	0.17	0.63
What continent is Italy on?	0.25	0.00	0.13	0.41
What do you call a professional map drawer?	0.00	0.00	0.00	0.00
What famous model was married to Billy Joel?	1.00	0.13	0.56	0.07
What is the collective noun for geese?	0.17	0.33	0.25	0.75
What is the collective term for geese?	0.33	0.20	0.27	0.81
What is the Islamic counterpart to the Red Cross?	1.00	1.00	1.00	0.68
What is the largest city in Wisconsin?	0.33	1.00	0.67	1.00
What is the largest snake in the world?	1.00	0.50	0.75	0.81
What is the largest variety of cactus?	0.00	0.00	0.00	0.33
What is the most heavily caffeinated soft drink?	0.00	0.00	0.00	0.05
What ocean did the Titanic sink in?	0.20	0.14	0.17	0.68
Average score across all questions:	**0.30**	**0.24**	**0.27**	**0.42**

Beyond Keywords

Comparing and evaluating different Web QA systems is not straightforward and, to our knowledge, has never been done before the study we describe here. In the annual TREC competition, the rules are set in advance, and participating researchers approximately predict the distribution and types of questions that would be expected from their experience in prior years. Meanwhile, the objectives of each Web QA system are different. The commercial systems (such as Ask.com and BrainBoost) are primarily interested in increasing traffic volume and visibility online to generate maximum potential advertising revenue or investment capital. The research prototypes (such as ASU QA and NSIR) are primarily interested in demonstrating innovative ideas in certain unexplored fields of research involving information seeking, not in competing with commercial systems. As a result, the systems we consider here support different sets of features and interfaces, as in Table 1.

The goal of our study in spring 2005 and repeated in 2007/2008 was not to compare QA systems against each other but to determine whether any of them might offer additional power relative to keyword search engines, exemplified by Google and MSN. In particular, we wanted to know whether automated QA technology provides answers to certain questions that keywords may find difficult or impossible to answer. For this reason, we performed an informal comparison of the QA systems in Table

1 as a "team" vs. the keyword searching technique (Google and MSN as another "team"). We also wanted to know whether QA might decrease the cognitive load during the answer-seeking process. We therefore claim only that the idea of "going beyond keywords" is possible while searching for answers to questions, not that a particular system is better than another particular system.

No researcher has yet claimed to have produced a representative set of questions for evaluating QA systems. Indeed, such a set might have to include thousands of questions to adequately represent each possible type of question. We built a data set based on our experience with IT practitioners. We merged all the TREC questions with a set of 2,477,283 questions extracted earlier by Radev et al.[5] from the Excite search engine log of real search sessions.[5] We then distributed nonoverlapping sets of 100 randomly drawn questions to each of the 16 students in a technology-related MBA class at Arizona State University. The survey was followed by interviews and resulted in the selection of 28 test questions guided by participant choices and comments. In order to avoid researcher bias, it was crucial that we not enter any of the questions into an online system—search engine or QA—before deciding whether to select that particular question for the test.

We used the mean reciprocal rank (MRR) of the first correct answer, a metric also used during the 2001 and 2002 TREC competitions and in several follow-up studies. It assigns a score of 1 to the question if the first answer is correct. If only the second answer is correct, the score is ½. The third correct answer results in a score of ⅓. The intuition that went into devising this metric is that a reader of online question-answering results typically scans answers sequentially, and "eyeballing" time is approximately proportional to the number of wrong answers before the correct one pops up. However, this computation is known to "misbehave" statistically, being overly sensitive to the cut-off position, the lowest-ranked answer considered,[5] thus its reciprocals are typically reported and used for averaging and statistical testing. Results are outlined in Table 2.

By rerunning our analysis with each of the members excluded from the QA team, we verified that no weak players would pull down the QA team's performance. Because our intention was not to compare individual QA systems, we did not include the data for each individual QA system. The average results support the following observations:

- The QA team performed much better than the keyword-search-engine team, an MRR of 0.42 vs. 0.27; a remarkable 50% improvement was statistically significant, with the p value of the t-test at 0.002;
- The average performance of the QA team is better than the performance of each search engine individually; moreover, each QA system performed better than each keyword search engine;
- For each question to which an answer was found by a keyword search engine, at least one QA system also found an answer; the reverse was not always the case; and
- If a QA system found the correct answer, it was typically second or third in the ranked list; only the fourth or fifth snippet from Google or MSN typically provided the correct answer.

To verify the stability of these observations, we re-ran our tests in spring 2006. Although most of the measurements of the specific systems with respect to the specific questions had changed, their overall performance did not change significantly, and our observations were further reinforced.

Conclusion

Based on our interaction with business IT practitioners and an informal evaluation, we conclude that open-domain QA has emerged as a technology that complements or even rivals keyword-based search engines. It allows information seekers to go beyond keywords to quickly answer their questions. Users with limited communication bandwidth (as a result of small-screen devices or having some visual handicap) will benefit most. And users under some time constraint (such as first responders at a natural disaster) will likely find it more suitable compared to the keywords-to-snippets approach offered by popular search portals like Google and MSN.

However, to compete with established keyword-based search engines, QA systems still must address several technical challenges:

Scalability. Web QA system response time lags the one-to-two-second performance provided by today's search engines; more research needs to be done as to how to make Web QA systems more scalable in order to process the comparable loads simultaneously;

Credibility. Information on the Web, though rich, is less factually reliable than counterpart material published on paper; how can QA system developers, as well as search users, factor source credibility into answer ranking?; and

Usability. Designers of online QA interfaces must address whether QA systems should display precise answers, sentences, or snippets.

We look forward to the next five to 10 years for advances in all of them.

References

1. Berwick, R.C. Principles of principle-based parsing. In *Principle-Based Parsing Computation and Psycholinguistics*, R.C. Berwick, S.P. Abney, and C. Tinny, Eds. Kluwer Academic Publishers, Norwell, MA, 1991, 1–38.
2. Dumais, S., Banka, M., Brill, E., Lin, J., and Ng, A. Web question answering: Is more always better? In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Tampere, Finland, Aug. 11–15). ACM Press, New York 2002, 291-298.
3. Kwok, C., Etienne, O., and Weld, D.S. Scaling question answering to the Web. *ACM Transactions on Information Systems 19*, 3 (2001), 242–262.
4. Lempert, R.J., Popper, S.W., and Bankes, S.C. *Shaping the Next One Hundred Years: New Methods for Quantitative, Long-Term Policy Analysis.* RAND Corp., Santa Monica, CA, 2003; direct.bl.uk/bld/PlaceOrder.do?UIN=138854587&ETOC=RN&from=searchengine.
5. Radev, D., Fan, W., Qi, H., Wu, H., and Grewal, A. Probabilistic question answering on the Web. *Journal of the American Society for Information Science and Technology 56*, 6 (Apr. 2005), 571–583.
6. Roussinov, D. and Robles, J. Applying question answering technology to locating malevolent online content. *Decision Support Systems 43*, 4 (Aug. 2005), 1404–1418.
7. Surdeanu, M., Moldovan, D.I, and Harabagiu, S.M. Performance analysis of a distributed question/answering system. *IEEE Transactions on Parallel and Distributed Systems 13*, 6 (2002), 579–596.
8. Voorhees, E. and Buckland, L.P., Eds. *Proceedings of the 13th Text Retrieval Conference TREC 2004* (Gaithersburg, MD, Nov. 16–19). National Institute of Standards and Technology, Gaithersburg, MD, 2004; trec.nist.gov/pubs/trec13/t13_proceedings.html.

Dmitri Roussinov (Dmitri.Roussinov@cis.strath.ac.uk) is a senior lecturer in the Department of Computer and Information Sciences at the University of Strathclyde, Glasgow, Scotland. The study described here was performed when he was an assistant professor in the Department of Information Systems in the W.P. Carey School of Business at Arizona State University, Tempe, AZ.

Weiguo Fan (wfan@vt.edu) is an associate professor of information systems and of computer science at Virginia Polytechnic Institute and State University, Blacksburg, VA.

José Robles-Flores (jrobles@esan.edu.pe) is an assistant professor in the School of Business at ESAN University, Lima, Perú. The study described here was performed while he was working on his doctoral dissertation in the Department of Information Systems in the W.P. Carey School of Business at Arizona State University, Tempe, AZ.