Natural Language Engineering **15** (1): 1–8. © 2008 Cambridge University Press
doi:10.1017/S1351324908004877 Printed in the United Kingdom

Special issue on interactive question answering: Introduction

N. WEBB[1] and B. WEBBER[2]

[1]*Institute of Informatics, Logics and Security Studies, University at Albany, SUNY, USA*
e-mail: nwebb@albany.edu
[2]*School of Informatics, University of Edinburgh, UK*
e-mail: bonnie@inf.ed.ac.uk

(*Received 1 March 2007; revised 1 October 2007; accepted 9 September 2008; first published online 22 October 2008*)

Abstract

In this introduction, we present our overview of interactive question answering (IQA). We contextualize IQA in the wider field of question answering, and establish connections to research in Information Retrieval and Dialogue Systems. We highlight the development of QA as a field, and identify challenges in the present research paradigm for which IQA is a potential solution. Finally, we present an overview of papers in this special issue, drawing connections between these and the challenges they address.

1 What is interactive question answering?

Question answering (QA) differs from keyword search in two respects: in the increased information that a question can convey over a simple list of keywords and in the targetting of answers, as opposed to returning a list of links to potentially relevant documents or web pages, possibly augmented by a snippet of text from the document or web page, that suggests its relation to the query.

On the first point, while retrieving documents in which the keyword 'Mozart' occurs in close proximity to the keyword 'born' is likely to yield ones whose snippets contain an answer to the question 'When was Mozart born?', doing the same for the keywords 'countries' (or 'country'), 'Pope', 'visit', '1960' and '1970' is not going to provide any answer, much less a comprehensive one, to the question 'Which countries did the Pope visit between 1960 and 1970?' One needs, for example, to distinguish between finding a match for the keyword 'country' and finding things which *are* countries. Similarly, one needs to distinguish between finding a match for both the terms '1960' and '1970' in a document and finding dates within that ten-year time span. Additional differences abound.

On the second point, a user in search of particular information may find their information contained in the snippet itself, obviating the need for any further access. QA as a discipline moves this from an accidental property of search to its focus, but doing this reliably requires queries that are more than a bag of keywords or phrases. Information in questions can help not only to identify documents that might contain

an answer, but also to extract that answer and return it to the user, more reliably than a snippet does.

In keyword search, this process is seen as a single-shot affair: The user asks a query, the system responds and the user goes away happy. However, there are significant issues in query formulation which mean that often a user's initial query is not enough. While a user may attempt to formulate a query with a good chance of retrieving useful material, s/he will not know whether the query has been successful until something has been returned for examination. If s/he is unhappy with the results or discovers some new information that leads to broadening or narrowing the scope of their inquiry, s/he can reformulate the query and try again. In this view, the query cycle can be seen as having more than a single iteration, and ideally the system serves as a cooperative partner in the information search process. This realization has, in part, led to the development of *interactive QA* (IQA) systems.

Here we present our own perspective of the development of QA, so as to set the stage for the challenges that IQA addresses. We will also point to recent work in Natural Language dialogue systems in order to define IQA as a paradigm that sites QA within the framework of continuous, cooperative interaction used in dialogue systems.

1.1 Historical development

Early QA systems were developed to enable users to ask increasingly complex queries about well-structured data sets, such as baseball statistics (Green *et al.*, 1961) and analyses of rock and soil samples returned from the Apollo lunar missions (Woods, Kaplan and Nash-Webber, 1972). Given the complexity of the queries and the fact that users were often unaware of the underlying database, early QA development focussed on such issues as handling questions that could not be parsed or that could not be interpreted as a valid database query; correctly resolving syntactic and referential ambiguity in user questions; handling differences in how user and system conceptualized the domain (e.g., user queries about the age of lunar rocks versus data on potassium/rubidium ratios, as well as differences between what user and system believed to be true in the domain (Kaplan, 1982; Mays, Joshi and Webber, 1982; Pollack, 1986; Webber, 1986) and identifying, in the case of distributed databases, what information needed to be imported from where, in order to answer the user's question (Hendrix *et al.*, 1978).

While devising interactions to handle mismatches between user and system beliefs about the domain was a step toward *IQA* systems (Section 1.3), the enterprise of database QA was essentially abandoned in the late 80's for lack of audience take-up: Ordinary people lacked access to large data sets, and managers whose companies maintained them lacked sufficient interest. Companies such as Symantic ended up abandoning their software for then state-of-the-art database QA (Hendrix, 1986).

With the advent of the web, came ever increasing amounts of information that people wanted to access. Search methods pioneered in document retrieval (a long-stagnant field newly revitalized by the web) became commonplace. And QA too

found new life, with the idea that users would find not just relevant web pages or documents, but the particular information they were seeking.

This new vision of QA is very different from QA over databases, where questions map onto a *computation* carried out over a database – for example, computing whether some property held of some set (e.g., 'Are all employees enrolled in a health plan?'), computing property values (e.g., 'What is the average concentration of iron in ilmenite?'), computing which entities had some possibly complex set of properties (e.g., 'Samples that have greater than 13 percent olivine'), etc. In contrast, within the current QA paradigm, the answer to a question is meant to be *found* rather than *computed*, and QA became viewed as natural extension to information retrieval (IR), accelerated by its adoption as a track within the Text Retrieval Conference (TREC).

Initially, TREC QA focused on factoid questions – those questions of a specific type, such as *who*, *what* and *where*, that can be answered by a short word or phrase. From 1999 to 2007, TREC QA advanced to address increasingly large document collections, increasingly complex questions and ever more complex evaluation strategies. While the basic approach to factoid QA is now well understood, challenges remain, including the fact that performance in answering list questions (identifying all entities that have some property), is notoriously poor compared to single-entity factoid questions. A second challenge is the fact that the same answer may be conveyed in so many different ways, even within a single sentence, that even using lexical resources to extend search based on pattern matching may not help find answers (Kaisser and Webber, 2007). Shallow inference based on a long-tail of infrequent patterns may help in recognizing answers, but it is not yet clear how this could be exploited in *finding* answer candidates. A third challenge is the fact that current techniques cannot identify answers whose evidence is distributed over multiple sentences (either over more than one adjacent sentence or over an arbitrary set of sentences from across the corpus, comparable to *joins* in database QA. Bouma *et al*., 2005 have called these 'which' questions, as in 'Which ferry sank southeast of the island Utö?', which requires combining evidence that some entity sank southeast of the island Utö with evidence that the entity is a ferry. Even factoid questions still present a challenge, such as ones involving superlatives (e.g., *the most*, *the worst*, *the largest*, etc.) or averages that can only be answered through a calculation that has not yet been performed or whose value depends on when it is performed, such as relative dates. As noted before, current methods are limited to cases where the answer has already been computed and indexed by the terms used in the question.

Moreover, development of QA inside an IR program has meant that preconceptions from IR have carried over into the present QA system development, including the idea of the *perfect query* (with respect to the user's desired information) – the query that the user *would* have asked if only they were familiar with the underlying data. This idea underlies recent research in *interactive IR* (Dumais and Belkin, 2005), which focuses on the relevance of feedback techniques, to allow the user to modify the query, over some iterations. While this is a form of interaction, it does not reflect the potential for users to change what they want to know as they learn more.

One important class of users for whom this is true and whose needs are being addressed through work on *complex QA* and the ARDA AQUAINT program, a US research program that is driving the development of *analytical QA*, are professional information analysts employed by government and industry. Their questions are not factoid in nature and cannot be satisfied by phrases or one word answers. Complex questions (such as *what are the long term implications of China's one child policy?*) require sophisticated approaches for the identification of relevant information, which may involve hypotheses and their consequences, analogies and comparisons, none of which plays any part in factoid or even definition QA. These needs of these users can rarely be satisfied through one or more independent questions.

1.2 Dialogue systems

Whilst the idea of continuous Natural Language interaction with users is fairly new to QA, the idea has had a significant history of its own, going back almost as long, under the rubric *dialogue systems*. In one of the earliest systems, SHRDLU (Winograd, 1973), users engage in dialogue with an animated or robotic agent that is capable of various actions and 'aware' of its own behavior, including its interaction with the user. As such, dialogues involve the user asking or telling the system to perform some action or achieve some goal; the system accepting the request/command or rejecting it (along with an explanation); the system asking for clarification of some aspect of the request/command; the user responding with clarifying description; the user asking the system questions about the state of its world or its previous actions or its plans and the system responding with an appropriate description.

Interactions with SHRDLU and with Vere and Bickmore's Basic Agent (Vere and Bickmore, 1990) were through typed text, where later systems supported limited spoken interaction (Allen *et al*., 1996; Lemon *et al*., 2001; Eliasson, 2007). Because these systems do not have any goals of their own, apart from those adopted in response to user requests/commands, and because no way was provided for users to collaborate with them, dialogues capabilities lack comparable questions, requests, or any initiative from the system.

More recent dialogue systems have played the tutor role in *Intelligent Tutoring Systems*. Here, it is the system that has goals – for example, assessing the student's knowledge, correcting the student's errors and imparting information that the student is missing. Dialogues can thus involve the system introducing and describing a topic for tutoring; asking the student about some aspect of the problem; explaining why the student's response is correct or incorrect and/or reminding the student of something already said previously during the interaction, with (in all cases), the student replying with an answer or saying s/he does not know. Again, the earliest tutoring systems, like SOPHIE (Brown and Burton, 1975), interacted through typed text, while later systems such as ITSPOKE (Litman and Forbes-Riley, 2006) allow for spoken interaction.

The most industrially relevant role played by dialogue systems is in information provision and user assistance, such as in helping users to book flights (Hemphill,

Godfrey and Doddington, 1990; Walker, Passonneau and Boland, 2001; Seneff, 2004; Demberg and Moore, 2006) or find a place to have dinner (Walker, Whittaker and Stent, 2004) routing user telephone calls to the appropriate staff person (Gorin, Riccardi and Wright, 1997; Chu-Carroll and Carpenter, 1999) or handling directory enquiries (De Roeck *et al.*, 2000; Lehtinen *et al.*, 2000). All such tasks can be viewed in terms of getting the user to fully or partially instantiate some frame, which the system then evaluates, presenting the results to the user as a basis for further interaction. In such cases, the user's goal may be anywhere from completely formed to vague. It may or may not be possible to achieve exactly as specified, and as such, may need to be reformulated on the basis of additional knowlege and relaxed constraints. Dialogues can thus involve the system asking the user questions related to values of frame elements; the user specifying such values (either precisely or vaguely); the system listing and/or describing the results (when too numerous to list); the user choosing some item from among the results or modifying or replacing some already specified values and the system requesting confirmation of its understanding.

Recent research on dialogue systems has involved widening their dialogue capabilities (Demberg and Moore, 2006), using data-driven models of dialogue (Hardy *et al.*, 2004), learning optimal dialogue strategies for a given task (Henderson, Lemon and Georgila, 2008) and exploiting formal, semantic approaches to information update (Traum and Larsson, 2003). A key-emerging element of dialogue approaches is their inherent generality – the potential for subdialogue structures independent of task or application (such as for error correction or clarification) that are vital for allowing users to explore open collections of data.

1.3 IQA

The emerging paradigm of *IQA* can be placed at the intersection of these two research directions in QA and dialogue systems. We define IQA as a process where the user is a continual part of the information loop – as originator of the query, as arbitrator over information relevance and as consumer of the final product. This mode of operation is useful for both factoid and analytical or complex QA, but perhaps provides greatest benefit in those cases where information need is as yet vague, or is comprised of multifaceted complex concepts, or contains user misconceptions or subjective opinions – all cases where the expected information content returned is complex and contains a greater degree of variability or ambiguity. IQA systems borrow from dialogue systems, their interaction, emphasis on completion of user task, their handling of incomplete or underspecified (as well as overspecified) user input and the constraint and relaxation phases of the query process, whilst remaining focussed on large or open domains, such as the Internet.

In June 2006 we organised a workshop on IQA, in conjunction with the Human Language Technology (HLT) 2006 conference, in Brooklyn, NY, USA. The six papers in this special issue are expanded, reviewed and significantly revised versions of papers from among those presented there.

2 Overview of special issue

The first two papers in this special issue describe work from a QA background that incorporates interaction or dialogue. The next three come from a dialogue background and consider the implications of QA. And the final paper centres on evaluation frameworks for IQA systems.

In the first paper, Varges and his colleagues address issues of generic methods for selecting and presenting information succinctly in Natural Language in the context of spoken database QA. Constraints provide an elegant way of dealing with a range of cases in which there are no results returned from the user's original query, or a small number of results or too many results for succinct presentation.

In the second paper, Small and Strzalkowski describe the HITIQA system (High Quality Intelligence through IQA), developed as a part of the ARDA AQUAINT program. HITIQA allows users to ask complex questions, uses frames as a representation of both question and underlying data and helps users through a simple dialogue and sophisticated visualization system over those frames to expand their query space and explore the resulting data. HITIQA uses a scalable, data-driven approach to suggest items of possible relevance to the analyst, which s/he can either accept or reject, which then shapes and defines information s/he sees as the process proceeds over time.

The third paper, by Rieser and Lemon, discusses the problem of learning a policy for IQA where the policy involves dialogue moves for acquiring reliable query contraints and for presenting some number of database results. Optimality involves trade-offs between the length of the answer list, the length of the full interaction and the amount of noise in the communication channel. Of particular interest is the authors' use of simulated database retrieval to investigate the effect that the nature of the database, together with noise, has on policy learning.

In the fourth paper, Quarteroni and Manandhar describe an IQA system implemented as a ChatBot whose web-based answers to description and definition questions have been designed to reflect the age, reading level and interests of the user posing the question. The system allows for both new and follow-up questions (factoid, description and definition) and can engage in clarification interactions to resolve cases of referential ambiguity in follow-up questions.

The fifth paper, by Van Schooten *et al.* describes a pair of dialogue systems, IMIX and Ritel, which have been used to study follow-up questions. In this study, they use corpora collected from these systems as well as other sources, and compare and contrast the handling of follow-up questions between their two systems, drawing important generalizations about these questions across domains and applications.

In the final paper, Kelly and her colleagues address the issue of instability of traditional evaluation metrics in multiuser environments and describe the use of questionnaires to evaluate a range of IQA systems, as a method of garnering effective user feedback about the systems themselves, and involve users in a subjective evaluation process. Key is the ability to discriminate between the resulting systems on the basis of several hypotheses, such as effort and efficiency.

Acknowledgments

The editors wish to thank all the participants in the June 2006 workshop, the authors who submitted papers to this special issue, and particularly the dedicated reviewers, without whom which we could not have possibly put this issue together.

References

Allen, J., Miller, B., Ringger, E. and Sikorski, T. 1996. A robust system for natural spoken dialogue. In *Proceedings of the 34th Annual Meeting, Association for Computational Linguistics*, University of California, Santa Cruz, pp. 62–70.

Bouma, G., Fahmi, I., Mur, J., van Noord, G., van der Plas, L. and Tiedemann, J. 2005. Linguistic knowledge and qa. *Traitement Automatique des Langues (TAL)* **46**(3):15–39.

Brown, J. S. and Burton, R. 1975. Multiple representations of knowledge for tutorial reasoning. In D. G. Bobrow and A. Collins (eds.), *Representation and Understanding*, pp. 311–49. New York: Academic Press.

Chu-Carroll, J. and Carpenter, B. 1999. Vector-based natural language call routing. *Computational Linguistics* **25**: 361–88.

De Roeck, A., Kruschwitz, U., Scott, P., Steel, S., Turner, R. and Webb, N. 2000. The YPA - an assistant for classified directory enquiry. In *Intelligent Systems and Soft Computing: Prospects, Tools and Applications. Lecture Notes in Artificial Intelligence (LNAI)*, Berlin, Germany, vol. 1804. Springer Verlag.

Demberg, V. and Moore, J. 2006. Information presentation in spoken dialogue systems. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL), Trento, Italy*.

Dumais, S. T. and Belkin, N. J. 2005. The TREC interactive tracks: putting the user into search. In E. M. Voorhees, and D. K. Harman, (eds.), *TREC: Experiment and Evaluation in Information Retrieval*, Cambridge, MA, USA, pp. 123–53. MIT Press.

Eliasson, K. 2007. Case-based techniques used for dialogue understanding and planning in a human-robot dialogue system. In *Proceedings of the International Joint Conference on Artificial Intelligence*, Hyderabad, India, pp. 1600–05.

Gorin, A., Riccardi, G. and Wright, J. 1997. How May I Help You? *Speech Communication* **23**: 113–27.

Green, B. F., Wolf, A. K., Chomsky, C. and Laughery, K. 1961. Baseball: an automatic question answerer. In *Proceedings of the Western Joint Computer Conference*, NY, USA, pp. 219–224.

Hardy, H., Biermann, A., Inouye, R. B., Mckenzie, A., Strzalkowski, T., Ursu, C., Webb, N. and Wu, M. 2004. Data driven strategies for an automated dialogue system. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, Barcelona.

Hemphill, C., Godfrey, J. and Doddington, G. 1990. The ATIS spoken language systems pilot corpus. In *Proceedings of the DARPA Speech and Natural Language Workshop*, Hidden Valley, PA, USA, pp. 96–101.

Henderson, J., Lemon, O. and Georgila, K. 2008. Hybrid reinforcement/supervised learning of dialogue policies from fixed datasets. *Computational Linguistics*, **34**, to appear.

Hendrix, G. 1986. Bringing natural language processing to the microcomputer market. In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics*, NY, USA.

Hendrix, G., Sacerdoti, E., Sagalowicz, D. and Slocum, J. 1978. Developing a natural language interface to complex data. *ACM Transactions on Database Systems* **3**: 105–47.

Kaisser, M. and Webber, B. June 2007. Question answering based on semantic roles. In *ACL 2007 Workshop on Deep Linguistic Processing*, pp. 41–48, Prague, Czech Republic. Association for Computational Linguistics.

Kaplan, J. 1982. Cooperative responses from a portable natural language database query system. In M. Brady and R. Berwick (eds.), *Computational Models of Discourse*, pp. 167–208. Cambridge MA: MIT Press.

Lehtinen, G., Safra, S., Gauger, M., Kaspar, B., Pardo, J. M. and Louloudis, D. 2000. Idas: interactive directory assistance services. In *Proceedings of the COST249 ISCA Workshop on Voice Operated Telecom Services* Ghent, Belgium, pp. 51–54.

Lemon, O., Bracy, A., Gruenstein, A. and Peters, S. 2001. The witas multi-modal dialogue system i. In *Proceedings of 7th European Conference on Speech Communication and Technology* (eurospeech), Aalborg, Denmark.

Litman, D. and Forbes-Riley, K. 2006. Correlations betweeen dialogue acts and learning in spoken tutoring dialogues. *Natural Language Engineering* **12**: 161–76. Further information at http://www.cs.pitt.edu/litman/itspoke.html.

Mays, E., Joshi, A. and Webber, B. 1982. Taking the initiative in natural language data base interactions: monitoring as response. In *Proceedings of the European Conference on Artificial Intelligence* Orsay, France, pp. 255–56.

Pollack, M. 1986. *Inferring Domain Plans in Question-Answering*. Ph.D. Thesis, Department of Computer & Information Science, University of Pennsylvania.

Seneff, S. 2002. Response planning and generation in the MERCURY flight reservation system. *Computer Speech and Language* **16**: 283–312.

Traum, D. and Larsson, S. 2003. The information state approach to dialogue management. In J. van Kuppevelt, and R. Smith (eds.), *Current and New Directions in Discourse and Dialogue*, Berlin, Germany, pp. 325–53. Kluwer.

Vere, S. and Bickmore, T. 1990. A basic agent. *Computational Intelligence* **6**(1): 41–60.

Walker, M., Passonneau, R. and Boland, J. E. 2001. Quantitative and qualitative evaluation of darpa communicator spoken dialogue systems. In *Proceedings of the Meeting of the Association of Computational Linguistics*.

Walker, M., Whittaker, S. and Stent, A. 2004. Generation and evaluation of user tailored responses in dialogue. *Cognitive Science* **28**: 811–40.

Webber, B. 1986. Questions, answers and responses. In M. Brodie and J. Mylopoulos (eds.), *On Knowledge Base Systems*, pp. 365–401. New York: Springer-Verlag.

Winograd, T. 1973. A procedural model of language understanding. In R. Schank and K. Colby (eds.), *Computer Models of Thought and Language*, pp. 152–86. New York: W. H. Freeman and Company. Reprinted in B. J. Grosz, K. Spark-Jones and B. L. Webber (eds.). 1986. *Readings in Natural Language Processing*, pp. 249–66. Los Altos CA: Morgan Kaufmann.

Woods, W., Kaplan, R. and Nash-Webber, B. 1972. The Lunar Sciences Natural Language Information System: Final Report. In *BBN Report 2378*.