

Chapter #

Habitability in Question-Answering Systems

Ogden, William¹, James McDonald¹, Philip Bernick², Roger Chadwick¹

¹*The Computing Research Laboratory, New Mexico State University, Las Cruces, NM*

²*Arizona State University, Tempe AZ*

Key words:

1. ABSTRACT

This chapter discusses our work in evaluating question-answering systems. We begin by reviewing evaluation concepts and methods that have been used in the past. We then consider more recent efforts to evaluate search engines that have contributed to the techniques we currently use. Finally we discuss our techniques and discuss results from user studies we have conducted. Implications are discussed for the future design of habitable question-answering systems.

2. INTRODUCTION

2.1 Background

For several decades researchers have worked hard to design and build tools for information retrieval. Some attempts involved building natural language interfaces to these systems. By the 90's, many of the problems associated with natural language interfaces had been identified, but the expense and difficulty of resolving them was so great that many developers refocused their energy elsewhere. Some of the redirected effort was spent developing tools for publishing, and locating and navigating information on the web.

In the mid-1990's several tools were available: web crawlers that collected links and information published on the web into a database, and search engines for navigating or querying those databases. Current versions of these tools enable users to form queries using natural language, and significant energy is being expended to build and refine them. Many refinements are in the area of question-answering systems. While search engines typically provide users with links to potentially useful source information, question-answering systems perform more sophisticated processing of user queries and source data, and attempt to answer queries directly. As these systems mature it is appropriate to look at methods for evaluating and identifying ways to improve them. These methods are the focus of this chapter.

2.2 Usability and Habitability

Habitability, in the context of information retrieval systems, is a term that has been used by Watt [19] and others to describe how well a system supports the language people use to interact with the system. For example, if there are several ways to ask a question, a habitable system language supports all of them by providing a useful response to each. While people may not get the information they are looking for, habitable systems make the reasons for failure clear. Looking for habitability is a particularly useful way to evaluate natural language systems because these systems attempt to support all of the ways that people naturally use language to interact with the system. By looking at how systems fail to be habitable we can identify ways to improve them. Present day search engines and question-answering systems are similar to earlier natural language interfaces (even when no natural language processing component is present) because queries often

take the form of natural language. For this reason habitability is at the center of our work.

We use the term *habitability* to mean more than usability. Habitability describes how easily, naturally, and effectively people can use language to express themselves within the constraints of a system language. Habitable systems enable users to express everything that is needed for a task using language they think the system will understand.

Ogden [17] describes habitability in the context of four domains: *conceptual*, *functional*, *syntactic* and *lexical*, that impact system habitability. To successfully use a system a user “must learn to stay within the limits of all four domains, and habitable systems must provide information about their capabilities in all four” (pp. 138-39). Evaluating system habitability requires observing how people use a system and how the system responds to and supports their search strategies..

Information retrieval systems generally, and question-answering systems particularly must:

1. Respond to people’s queries in useful ways, and
2. Support search strategies that people are already familiar with or can learn easily.

The goal of the user studies described below is to discover habitability problems and their solutions

2.3 Important Evaluation Considerations

The goal of our evaluation effort is to find out how people actually use the system being evaluated, and/or to find out how well a system meets users’ needs. As a result, choosing users and tasks for system evaluation is particularly important.

2.3.1 Choosing Users for System Evaluation

System habitability depends on how well a system language matches a user’s knowledge in the domain of discourse. Studies of systems that have been designed for unique user populations select participants carefully or train them in the subject domain. But question-answering systems may have no such defined user population, and there may be no constraints on the kinds of questions users can ask. For obvious reasons, evaluating systems designed for general user populations is more difficult than evaluating systems designed for restricted populations.

2.3.2 Choosing User Tasks for System Evaluation

Evaluation tasks, like users, must resemble the tasks real systems might be used to solve. Tasks for question-answering systems are sometimes selected from searches people have already performed, sometimes generated by experimenters in order to exercise greater control over elements of the system that get used, or to compare two or more systems. Participants may also be allowed to generate their own queries. Participant-generated tasks can provide a higher level of validity because they may more closely resemble tasks an actual system might encounter. On the other hand, participants may have a difficult time coming up with appropriate questions, or questions that exercise a sufficient number of system components.

2.3.3 User Training

If actual users are not available, or if a system is designed for users who need to be trained to use it, study participants might need to be trained both on knowledge of the domain and on how to use the system. Training affects how participants perform in system evaluations, so the training must reflect the kinds of experience users would get with an actual system.

2.3.4 Data Collection and Analysis

A variety of methods for data collection are available. These include user observations in the workplace, interviews, and laboratory experiments that record everything that happens during a session using audio, video, and think-aloud techniques. Some researchers, like Lewis and Rieman [20], distinguish between two sorts of data: process data, and bottom-line data. Process data consist of observations of what participants are thinking about and doing during a study. Bottom-line data include the end results, tell us what happened, and are represented in terms of numbers, i.e., how long it took to process a query, how long it took to complete a task, user satisfaction ratings, numbers of results returned, accuracy of results, etc.

Measuring the accuracy or success of a system is an important consideration. It is difficult to determine if a correct answer has been returned by a system without doing an exhaustive search of the data, which, for most systems, is tedious and difficult. Methods for estimating accuracy have been developed for the Text Retrieval Conferences (TREC), but they require a fixed set of questions and independent human assessment. For habitability evaluations, it is important to allow for question variability. Measuring system accuracy is much more important for comparison studies where one is looking for comparison measures. For habitability evaluations,

where the goal is to find ways of improving habitability of a system, user-ratings of the success of an interaction may be more diagnostic. The user may be the best judge of the information that satisfies a given question.

Think-aloud methods can also be used to complement data collected through observation. In the context of our studies, thinking aloud involves having participants articulate what it is they are doing and why, and can help clarify what is observed during the course of the study.

3. RELATED WORK

There is a growing body of work that looks at how people search for information and tools that help them. Several studies look at search engine interfaces to electronic information systems, and identify general characteristics that useful interfaces ought to have.¹ Researchers in these studies looked primarily at bibliographic databases and the web interfaces to them. They did not, however, address search engines generally, or how people use them to locate information on the web. Hölscher and Strube [6] looked at user behaviors to determine useful elements to include in search engine interfaces but did not look at the implemented characteristics of specific search engines.

Other studies have explored user behaviors during browsing or navigating the web, but none of them have focused on search engines.² Choo, Detlor and Turnbull [4] looked at the information seeking behavior of workers over a two week period using surveys, interviews and client-side logs. They characterized several information seeking behaviors of web users, and summarized them using a model of behavioral modes and moves. Navarro-Prieto, Scaife and Rogers [13] identified cognitive strategies related to web searching by comparing web searchers with high and low experience. They concluded that expert searchers plan ahead in their searching behavior based on their knowledge about the web, while novices hardly plan at all and are driven by external representations, for example, by what they see on the screen. Of interest here, but still missing, are examples of how specific interface characteristics supported or interfered with this behavior.

¹ Examples of these include studies by Marchionini, Dwiggins, Katz and Lin 35-69; Shneiderman, Byrd, and Croft <http://www.dlib.org/dlib/january97/retrieval/01shneiderman.html>.

² These studies include Cockburn and Jones 105-129; Catledge and Pitkow 1065-73; Tauscher and Greenberg 97-137.

Other researchers have collected large datasets derived from the logs of Internet search engines like Excite and AltaVista.³ While these studies offer a picture of the actual queries that search engine users produce, and how the engine responds to those queries, the relationships between a query and the user's actual information need are difficult to determine because the data source consisted of a query log. There were no users who could be asked about the tasks or task goals that motivated their queries, or how well the search results met their goals. None of these studies looked at system habitability.

Other studies have looked at user behaviors and queries, but these have often been in the context of understanding behaviors so that interfaces could be designed to support them, rather than identifying system characteristics that impact those behaviors. For example, O'Day and Jeffries [14] looked at the search behaviors of people (those with an information need and those conducting searches to satisfy that need) to identify behaviors that an electronic system would need to support. Spink, et al., looked more closely at the queries produced by search engine users.

Missing from these studies is research that tells us about current systems from the perspective of usability, and more particularly, from the perspective of habitability. Furthermore, few of the methodologies used in those studies can help answer questions of system habitability. As part of his dissertation research, Bernick [2] performed a comparison study of two generally available search-engine interfaces. This study, which attempted to identify system characteristics that impact habitability, is described in the following section.

4. A COMPARISON STUDY

For many years the Text Retrieval Conferences (TREC), have provided a venue for researchers working on problems associated with information retrieval to build, demonstrate, and test techniques and technologies. In the past, the TREC Interactive Track has used experimenter-generated tasks across systems to evaluate how well (how quickly and effectively) they retrieve information. While much of this work has focused on problems associated with text retrieval and summarization, and in building tools that can meet users' needs, less time has been spent looking at how the technologies are used once they are made more widely available. It is important to identify user needs prior to implementing a technology, but it is equally important to look at how the implementation of features designed to

³ See for example Jansen, Spink, and Saracevic 289-90; Jansen, Spink, Bateman, and Saracevic 5-17; Spink, Wolfram, Jansen, and Saracevic 226-34.

meet those needs are used once they are in the hands of users. As a result, Bernick [2] used a variation of the Guidelines from the TREC 2002 Interactive Track as the basis for this comparison study⁴.

The methodology for this comparison study varied from TREC in two ways. First, the TREC interactive track has a goal of identifying systems that produce efficient search results and accurate information. Less attention has been given to engine interfaces or user interaction. The comparison study looked primarily at user interaction; participant's determined whether results were useful or not, and task performance was not constrained by task time. Second, the TREC interactive track uses a database consisting of information derived solely from the .gov domain, whereas the comparison study used two popular web search engines (Google and Ask Jeeves) whose data are derived from all available web domains.

Bernick attempted to identify characteristics that impact habitability by asking the following questions:

- **Strategies:** Are there interface characteristics that support or prevent search strategies, or help users work with different search strategies?

Looking at strategies involved looking at how users approached search tasks, and evaluating whether those strategies changed over time or with different engines.

- **Query Formulation:** Are there interface characteristics that impact query formulation?

Evaluating query formulation involved looking at how participants used vocabulary, the number of query terms that were used and the number of unique terms, whether tasks were paraphrased, and how queries were modified.

- **Query Results:** Are there interface characteristics that facilitate or impede successful searches?

Query results were useful for identifying cues that participants attended to. The comparison study looked at how study participants used cues, i.e., how participants evaluated query results, located information and selected the sites that were visited.

- **Unsuccessful Search Tasks:** Are there interface characteristics that contribute to or mitigate failure?

In the comparison study, success or failure on a particular task was determined by the participant performing the task. Participants would usually indicate this by stating that they had either found what they needed to satisfy task criteria, or that they had not found what they wanted but were done looking. Data collection for the comparison study involved recording the sessions with software that provided a record both of the participant's

⁴ <http://www-nlpir.nist.gov/projects/t11i/guidelines.html>.

interaction with the system and verbal comments made while performing tasks.

Bernick found that most participants were successful in most searches. The most successful search strategies involved using enough key terms from the task, or paraphrasing the search task with enough specificity to produce results, identifying the links to information sources that did have information related to the information need, and continuing to search that information source for information that satisfied the participant's interpretation of the task's information need. This suggested two significant problems with the study.

1. Experimenter generated tasks were cast in a way that kept researchers from understanding how people go about satisfying their own information need because the task influences the way participants formulate the query.
2. Task participants using experimenter-generated tasks don't own the information need. Rather, they are trying to satisfy an artificial information need, and may be attempting to satisfy the researcher rather than themselves.

Most participants in the comparison study used strategies that resembled their initial strategy for all tasks; none seemed to change their strategy significantly over the course of the study. This would suggest that a person's past experience with similar tools influences the ways in which they use a new tool. Participants formulated queries by using key words from the task or by paraphrasing key words from the task, by forming natural language queries based on the task, or by using terms that were designed to locate a site, or at least a source, of useful information. The choice of vocabulary used was determined by a participant's immediate information need, but didn't always reflect the information needed to satisfy task criteria.

Most importantly, the vocabulary used by participants in the comparison study tended to reflect the information need posed by the task, but not always, and queries often used vocabulary from the experimenter-generated tasks. While the vocabulary used to describe a task was often understood and used in the same sense by the task author, experiment participant, and web authors, Bernick tried to explain task failure in part by appealing to the use of different word senses by web authors and task participants. However, it may simply be that the information need posed by the experimenter-generated task in addition to the language used in the task description produced results that were difficult for a particular participant to understand. In other words, since the participant didn't truly own the information need, they may not have really understood the task in a useful way, and hence, couldn't understand the results. The study design makes it difficult to

pinpoint when the participant was actually working outside their domain of expertise.

The comparison study was useful for gaining insight into the characteristics of search engines that help people locate information, some characteristics that might not be so useful, and for suggesting other areas and methods of inquiry. Most importantly, it demonstrated that other methods were needed in order to evaluate question-answering systems from the perspective of habitability.

From the comparison study it was clear that methods would need to be developed that:

1. Enable evaluation rather than comparison.
2. Provide insight to the relationship between how a user thinks a question-answering system works and the queries they produce for it.
3. Help us understand:
 - a. The sorts of knowledge that people need to have to formulate successful queries.
 - b. How much benefit people get from knowledge of a system's structure and content.
 - c. Whether people who get training do better, and if so, for how long.
 - d. The training that helps most.

3. IMPROVING METHODS FOR QUESTION-ANSWERING SYSTEM EVALUATION

The Bernick study looked at question-answering as an activity that could be accomplished with web search tools. There has been recent progress in developing systems that are designed to do a better job of natural-language question analysis and to return results that more directly answer users' questions. These systems attempt to let users ask questions the way they normally would and, unlike most web search engines, return answers, not web documents. We took the lessons learned from the Bernick study and applied them to the development of a methodology for studying two natural-language question-answering systems currently available as web demonstrations.

By looking at how systems fail to be habitable we can identify ways to improve them. Our method is focused on creating situations in which we can observe users' natural question-asking behavior. The goal of the user studies described below is to discover habitability problems and their solutions. The

results of the Bernick study led us to modify our method to allow participants to generate their own information needs. While this makes system comparisons more difficult, it allows us to better focus our evaluation on one system at a time.

A key interest of this approach will be: Does the design of the interface influence the types of questions people ask? Habitability is about discovering the ways in which people ask questions so that systems can answer them. If the design of the system influences the questions people ask, then it is important to determine those elements within the system that are responsible. In particular, if the system is designed to best answer full natural-language questions, then the system's interface displays and feedback should be designed to encourage this type of input.

3.1 LCC Study

The Language Computer Corporation (LCC) has developed state-of-the-art technology for question answering [12]. The purpose of this study was to examine the degree to which naïve users found the LCC system habitable, to identify interface issues, and to propose changes to the interface that would enhance usability. Our main concern was the development of a methodology that would lead to improvements in the system and to identify the useful characteristics of the system.

In this study we used participants' self-generated information needs as the sources of questions for the system. One of the problems in evaluating natural language systems is giving them questions to ask. The phrasings used in the questions provided often influence participant queries. Some methods [17] attempt to circumvent this by presenting "big picture" problems to be solved via decomposed questions. However, knowledge of system domain can greatly influence problem decomposition and problem-solving ability, and this is a potentially confounding variable. Having the participants generate their own questions from a short list of potential topics precludes problem-statement phrasing influences, provides a rich source of unique queries for analysis, and allows a better determination of answer satisfaction. The participant is the best judge of the information that satisfies a given question. Such information cannot always be gleaned directly from an initial question phrasing. For example, the question "how do I care for roses" might indicate a need for factual information on gardening, or it might be a request for a list of available products. The participant is in a unique position to determine if the information need is met by the system's answer. One drawback of this methodology, however, is the large number of

unanswerable questions generated. This can also be considered an advantage, however, since unanswerable questions reveal a great deal about the system interface, which is a major concern of habitability.

Method

Seven graduate psychology students at New Mexico State University participated in this study. All were experienced computer users and also experienced with search engine tasks. Participants were paid for their participation. Each student participated in a single session lasting between one and a half and two and half hours.

Participants were informed that their voice and computer screen data would be recorded and gave their consent. A brief survey was completed by each participant containing demographic questions and ratings of computer and search engine experience. Instructions on think-aloud verbal protocol were then given. After being shown a list of general topic categories, participants were requested to generate twelve questions of interest and rank them in order of interest. Of these questions, the eight rated highest in participant interest were selected for the search task. Participants also rated the questions for “breadth” or scope. Participants then queried the LCC system’s internet demo page⁵, for answers to each of their eight questions. Thus, a given information need would consist of one question and as many queries to the LCC system as required to complete the task. Audio and computer screen data were recorded using background software. Participants were instructed to verbalize their thoughts as they performed the task (think-aloud protocols). Instructions were given to terminate a specific search when the participants received what they considered to be satisfactory answers, when they came to believe that the system could not answer a question, or if they became too frustrated to continue with the given question. At the conclusion of each question, participants rated their satisfaction with the results, usefulness of the system, and change in scope (breadth) of the queries. At the conclusion of all eight questions, participants answered questions regarding the nature of their experience with the LCC system during the session.

Results

⁵ http://www.languagecomputer.com/demos/question_answering/internet_demo/index.html

The video/audio files captured during the sessions were edited into short clips covering specific questions or queries. An index to the HTML log files, which contained links to each of the video clips, is available online.⁶

Query Types

Since this study was primarily concerned with the behavior of the participants in how they approached question-answering systems, an analysis was performed on query types. Queries were categorized as directives (e.g. 'show me'), keyword, or natural language type questions. There were 146 queries, of which 52% were keyword type and 46.6% were natural language question type. The mean number of queries per information need was 2.61 (SD = 1.93). The tendency for the subjects to migrate either to or from keyword usage required further analysis. Despite the emphasis on using natural language with LCC, subjects tended to use a large percentage of keyword queries. Comments from users indicate some confusion about the necessity of using natural language questions. Participants had a tendency to use keywords due to their experiences with other Internet search engines such as Google and Yahoo. The LCC system is designed to work better with natural language questions, so it is unfortunate, but informative, that participants in this study did not always use questions. Developers of question-answering systems need to identify ways to encourage the use of natural language instead of keyword queries.

Answer Success

Since the LCC system uses the Internet database, we did not attempt to independently assess the extent to which the database contained an answer to each question. Instead, successful answers were measured by user ratings of their satisfaction with the results obtained from the system. Ratings were given using a 7-point scale. If success is defined as a rating above 4, 67.9% of the 56 generated information needs were considered successful and 32.1% unsuccessful.

User Comments

User comments were solicited from the seven participants at the conclusion of the study. Some of these comments are paraphrased below. Note that some of the comments have 'done' appended to show cases where changes were incorporated into the LCC system.

⁶ <http://troia.nmsu.edu:8001/transcripts/>.

"Which features of the retrieval system made it more useful?"

Question based queries
Short descriptions [of results]
The natural language method was nice but it didn't work well.
You could type phrases
Suggestions to misspellings
The system seemed capable of handling fairly narrow searches via the use of questions instead of just phrases.

"Were there any features of the retrieval system that could be improved?"

Organize the results chronologically or alphabetically
Avoid chat groups and personal emails [subjects did not like results that were not credible]
Bold the keywords
Include the website for each result [done]
Have a general menu with topics like "education"
Add [instructions ?] that you can use keywords not just questions.
No system feedback when you click on a link.
Didn't like that a new window opens
Web address not given to you in results [done]
Broader searches
No repeat websites
Have it look for keywords, not the question.
'The whole darn thing'
It seemed a little slow, reported status in a small blue font easy to miss so I thought nothing was happening.

Because the experimenter prompted the participants for comments, they may be exaggerations and should be carefully considered. However, it is clear that many users were uncomfortable or confused about the capabilities of the system.

General Observations and Conclusions

The users in this study generated a large number of time-critical questions, but the LCC system had no time processing capability. This created a significant number of problems. Apparently participants believed the system would be able to understand phrases like, "yesterday", "today",

and “currently”. For example, one participant entered the following set of questions before giving up:

Did the Bruins win today?
Did the Boston Bruins win their last game?
Did the Bruins win on 02/26/03?
Where can I find the Boston Bruins record for 2003?

This example is important for two reasons. First, the function necessary to answer these questions should be included in the capabilities of a natural-language question-answering system because users expect them. Second, and more importantly, the system should provide a feedback mechanism to alert users when they ask questions that are outside of the system’s capabilities. The system, like most traditional information retrieval systems, always returns some answer. The natural-language user can become more confused than the traditional user because the system’s response suggests that it understood the question, when in fact it did not. Natural language systems have many such “hidden” capabilities. If these capabilities were made visible, other missing capabilities might be easier to detect. In short, better feedback and/or better visibility of systems capabilities should be provided.

Other commonly occurring problems have simpler solutions::

1. Spell checking: The spell checker often came up with suggestions for spelling changes that seemed unnecessary. For example, the system suggested 'France' in response to 'france', although the results were not sensitive to case. There were other examples where spell checking seemed to need fine-tuning

2. Repeated web Sites in results: Several results were often presented which referred to the same web page, thus being redundant.

3. The 'Power Answer' button could be misleading. Some participants thought that performing a 'power answer' was somehow more than just asking for an answer. A simpler button that indicated 'search' or 'answer' would be simpler and avoid confusion and wasted time.

4. Phrasing of question: There appeared to be considerable difficulty determining the best way to ask for information. Participants tried natural language questions and keyword phrases. For some questions it seemed difficult to phrase the information need as a question.

8. While it seems to be unimportant from the system's perspective, participants often worried over whether or not to use a question mark. They often repeated queries with or without the punctuation if the system did not give them successful results. Again, users were confused about the hidden or missing capabilities of the system.

3.2 START study

START, a web-based question answering system, has been online since December, 1993. START was developed by Boris Katz and his associates in the InfoLab Group at the MIT Artificial Intelligence Laboratory [9]. The purpose of this study was to examine the degree to which naïve users found START habitable, to identify interface issues, and propose changes to the interface that would enhance habitability. The method was similar to that used in the LCC study reported above. We had participant's self-generated information needs as the source of questions for the system.

Method

Three female graduate students between the ages of 18 and 37 were paid to participate in this study. Each participant's session lasted between one and a half and two and a half hours. All participants were experienced computer users and experienced with search engine tasks.

After being shown the categories and example questions from the START web site, participants were asked to generate twenty questions of interest, distributed among the five categories. Participants ranked their questions in order of interest to themselves and rated each question for breadth. No training was given to participants beyond a review of the START home page and the categories and example questions,. Participants queried the START system for answers to each of their twenty questions in the order in which the questions were generated. Thus, a given information need consisted of one question and as many queries to START as required. Audio and computer screen data were recorded. Participants were instructed to verbalize their thoughts as they performed the tasks. Instructions were given to terminate a specific search if the participant received a satisfactory answer, came to believe that the system could not answer the question, or became too frustrated to continue with the search. At the conclusion of each question, the participant rated her satisfaction with the results, usefulness of the system, and change in scope (breadth) of the query. At the conclusion of

all twenty questions, participants answered questions regarding the nature of their experience with the START system.

Results

Overall, participants had a difficult time understanding the conceptual domain and functional restrictions of the system in the twenty-question session. A common comment was that they could not figure out how the system wanted them to phrase questions. Participants switched back and forth between using natural language questions and short 'keyword' questions. A summary of questions and ratings is provided online⁷. Participants were asked to rate how satisfied they were with the results they obtained on a 7 point scale. The mean satisfaction rating was 3.07, and mean usefulness was 3.65. Out of 60 questions, 35% were judged to be answered correctly by START, 10% were answerable by START but the right question was not asked, 47% were unanswerable by START, and 8% were judged to be partially answerable. These judgments were made after reviewing each question and extensively probing the START system for possible answers. There were 3.35 queries per question on average (standard deviation of 2.24), with fewer (mean 1.76, standard deviation 1.04) queries on the answered questions, as would be expected. Of the queries, 60% were natural language questions, 34% were short keyword phrases, and 6% were directives (e.g. 'show me').

Discussion

The results are perhaps best discussed in terms of specific examples, which illustrate the interface issues. Several key issues were examined and each is discussed below.

Failure Analysis

Some of the most useful results of this study came from a detailed failure analysis. The use of self-generated tasks provides a rich source of unique examples of queries. There are several types of failures to be considered. For example, some questions are answerable, but a satisfactory answer is not obtained. There are also answered question which required an unnecessary number of restructured queries. The researchers examined each question submitted by participants and submitted queries to START as needed in order to understand what went wrong. Causes of query failure

⁷ URL http://troia.nmsu.edu:8001/start-transcripts/start_subjets.html

were not readily apparent during the sessions, leading to much confusion about how the system functioned.

As an example of the failure to answer, one participant asked, "How many total episodes of 'Excel Saga' are there?" We subsequently found that the simple query 'excel saga' produced information that includes a direct answer to the question, although the answer is embedded in summary text and not directly accessed as a database item. The participant failed despite entering the query ["Excel Saga"]. We found that this query can produce a good result, with the answer embedded within the summary text. However, the START's automatic processing of ellipsis prevented the query from retrieving the results during the test session. This example, and others, led us to conclude that the ellipsis processing of START was doing much more harm than good, at least in our study, and should be re-evaluated. The same type of ellipsis interference problem also occurred in response to the question "Population of croatia". The initial query submitted without the prior queries resulted in a good answer. However, the participant rephrased the question in order to produce an answer because, due to the sequence of queries and perhaps timing, some not readily apparent ellipsis function resulted in an erroneous 'I don't know'. By examining the queries one by one in this manner we were able to glean useful information about interface characteristics that could lead to improvements. It is our belief that the self generated nature of the questions employed contributed greatly to the value of the analysis method.

Ellipsis

We concluded that the use of ellipsis in the system as implemented did more harm than good. The fact that the ellipsis function is hidden from the user and is not easy to discover was a source of confusion for participants. Queries that would have returned successful results, as illustrated in the previous paragraph, resulted in failure due to specific sequences, and perhaps the timing of queries submitted. In some cases participants were confused by feedback that, in it's natural-language generative form, indicated a response to a previous question. Beginning with an examination of these inappropriate and odd results, we were able to identify the use of ellipsis as the source of the confusion. Ellipsis processing could be signaled to the user by showing the expanded interpretation of the users question. Also we suggest that ellipsis processing only occur when the system could find and answer to the previous query.

Feedback

Verbal protocols provide a rich source of behavioral understanding, although much is still left to inference. It became apparent both in real time and when reviewing the recorded videos that users were confused by the feedback provided by the system. The feedback from START is often “generated” natural language, with variability in responses for the same query. This variability is intended to add “naturalness” to the interface, but may actually contribute to user confusion. Users commented that they could not determine what START wanted for input. The rephrasing that was prompted by the feedback may have been misguided. Consider the question "Who created the clarinet".

User: who created the clarinet

System: I'm afraid I can't help you with that.

User: who made the first clarinet

System: Sorry - I don't know who made the first clarinet.

User: where did the first clarinet come from

System: I don't know.

User: where did the first musical instrument come from

System: Sorry, I don't know the answer.

==> where was the first clarinet made

Sorry - I don't know where the first clarinet was made.

Upon further scrutiny, we concluded that in addition to the issue of variability in response, the feedback was insufficient in discriminating between possible failure types. For the question "Who is credited with the idea of chaos theory", the system understands “chaos theory” but doesn't have the ability to look up 'who is credited with'.

We are currently conducting a follow-up study with START to address this issue of variable feedback.

Fallback

Using the methodology of failure analysis, we probed the system extensively for answers. In many cases a direct answer was not possible using START, even though useful information was available in the database using simple queries. As an example, the question "What is SKA music"

does not produce a result, however "What is SKA" does. We suggest that there are many cases (e.g. "When did the Korean war happen") where falling back to the information available on the main query terms would produce satisfactory results,.

Conclusion

The use of self-generated questions, good data capturing, and detailed failure analyses provide a methodological structure that can yield practical results with natural language systems. Several key interface issues were uncovered using these methods. The developers of the START system have begun to address many of these.

4. EXPLAINING HABITABILITY: A TWO-STEP MODEL OF THE COGNITIVE PROCESS

Analyzing the problems users have when struggling with question-answering systems provides insights into their thought processes. Here we propose a model of these processes to focus our discussion of question-answering system design.

Our model of a successful interactive information retrieval process from an online source consists of two steps. First, the user imagines how the target information is represented in the computer database. The user first has to know what an answer looks like and must believe that the system contains it. Then the user needs to know how the retrieval system will access and return the target. (The user needs to understand enough about the mechanism of the systems to control which information is retrieved.)

For web-based retrieval, users first have to imagine that there exists a web document or set of documents with the information that they need. They then must think of a set of keywords or phrases (the set of instructions the system uses) that uniquely identify and return the documents. Good retrieval systems are designed (or have evolved) to map the keywords and phrases that users typically think of on to retrieval engines that return good (or useful) documents.

It is unlikely that most users understand this two-step process, at least not explicitly. However, we suggest that successful users employ this process,

and that systems that make it easier for them to do so will have greater success. People fail either because they do not have a clear idea of how an answer is represented by the system, or do not understand how to control the search engine well enough to retrieve the target information. Users find it difficult to determine whether the information is missing or if they simply do not know how to retrieve it.

4.1 Implications for system design

The ways that users formulate queries when interacting with information retrieval systems is determined both by their expectations about what questions systems may be able to answer (what they think the system knows) and how they think the system will use their input to retrieve the answers.

There are two ways a user can fail and a system can help in both cases. The necessary information could be in the database, but the user does not have a clear idea how it is represented. Alternatively, the user might know how the information is represented, but does not know how to make the system work. The system can help in both cases by transforming the user's query into terms or procedures that are more likely to match the user's intentions. A natural language question-answering system can be helpful in transforming the user's question into the language contained in the data. The problem comes when a transformation alters the user's intention and the user does not have access to the interpreted or transformed version of the query.

This is why we feel that feedback and visibility are critical design features for natural language systems. Unfortunately these features are often overlooked. Habitability requires that users have accurate expectations about what they can or cannot ask. Making visible the hidden capabilities of natural-language interfaces remains one of the biggest challenges for the design of future question-answering systems.

5. REFERENCES

- [1] Bates, Marcia J. "An Exploratory Paradigm for Online Information Retrieval." *Intelligent Information Systems for the Information Society*. Ed. B.C. Brookes. Amsterdam: North-Holland, 1986. 91-99.
- [2] Bernick, Philip. "Habitability in Search Engine Interfaces: Characteristics Identified Through Formative Evaluation Techniques." Doctoral dissertation, New Mexico State University, Las Cruces, NM, 2003.
- [3] Catledge, Lara D., and James E. Pitkow. "Characterizing Browsing Strategies in the World-Wide Web." *Computer Networks and ISDN Systems*. 27.6 (1995): 1065-73.
- [4] Choo, C. W., B. Detlor, and D. Turnbull. "Information Seeking on the Web - An integrated model of browsing and searching." *Proceedings of the Annual Meeting of the American Society for Information Science (ASIS)*, 23 June 2003 <<http://choo.fis.utoronto.ca/fis/respub/aisis99/>>
- [5] Chu, H., and Rosenthal, M. 1996. Search Engines for the World Wide Web: A Comparative Study and Evaluation Methodology. In *ASIS 1996 Annual Conference Proceedings*, 23 June 2003 <<http://www.asis.org/annual-96/ElectronicProceedings/chu.html>>
- [6] Hölscher, Christoph and Gerhard Strube. "Web Search Behavior of Internet Experts and Newbies." *Proceedings of the 9th International World Wide Web Conference (WWW9)*. 23 June 2003 <<http://www.www9.org/w9cdrom/81/81.html>>
- [7] Jansen, B.J., A. Spink, and T. Saracevic. T. "Failure Analysis in Query Constructions: Data and Analysis from a Large Sample of Web Queries." *Digital Libraries 98*. New York: ACM, 1998. 289-90.
- [8] Jansen, B. J., A. Spink, A., J. Bateman, and T. Saracevic. "Real Life Information Retrieval: A Study of User Queries on the Web." *SIGIR Forum*, 32.1 (1998): 5-17.
- [9] Boris Katz, Sue Felshin, Deniz Yuret, Ali Ibrahim, Jimmy Lin, Gregory Marton, Alton Jerome McFarland and Baris Temelkuran. "Omnibase: Uniform Access to Heterogeneous Data for Question Answering." *Proceedings of the 7th International Workshop on Applications of Natural Language to Information Systems (NLDB 2002)*, June, 2002.
- [11] Marchionini, G., S. Dwiggins, A. Katz, and X. Lin. "Information Seeking in Full-Text End-User-Oriented Search Systems: The Roles of Domain and Search Expertise." *LISR*, 15 (1993): 35-69.
- [12] Dan Moldovan, Marius Pasca, Sanda Harabagiu, and Mihai Surdeanu. Performance Issues and Error Analysis in an Open-Domain Question Answering System. 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002), July 2002, Philadelphia, Pennsylvania.
- [13] Navarro-Prieto, R., M. Scaife, and Y. Rogers. "Cognitive Strategies in Web Searching." *Proceedings of the 5th Conference on Human Factors and the Web*, 23 June 2003 <<http://zing.ncsl.nist.gov/hfweb/proceedings/navarro-prieto/index.html>>
- [14] O'day, Vicki, and Robin Jeffries. "Orienteering in an Information Landscape: How Information Seekers Get From Here to There." *HPL-92-127* Hewlett-Packard:n.p, 1992.
- [15] Ogden, William C., and Philp Bernick. "Oleada: User-Centered Tipster Technology for Language Instruction." *Proceedings of the Tipster Phase II 24 Month Workshop*. Tysons Corner, VA, 1996.

- [16] Ogden, William C., and Philp Bernick. "Tabula Rasa Meta-Tool: Text Extraction Toolbuilder Toolkit." Technical Report MCCS-97-305, Computing Research Laboratory, New Mexico State University, Las Cruces, NM, 1997.
- [17] Ogden, William C., and Philip Bernick. "Using Natural Language Interfaces." Handbook of Human Computer Interaction. Eds. Helender, M., T. Landauer, and P. Prabhu, 2nd Edition, Amsterdam: North Holland, 1997, 137-61.
- [18] Spink, A., D. Wolfram, B.J Jansen, and T. Saracevic. "Searching the Web: The Public and Their Queries." *Journal of the American Society for Information Science and Technology*, 52 (2001): 226-34.
- [19] Watt, W.C. "Habitability." *American Documentation*, July (1968) 338-51.
- [20] Clayton Lewis and John Rieman. Task-Centered User Interface Design: A Practical Introduction. available via anonymous ftp at: [ftp.cs.colorado.edu](ftp://ftp.cs.colorado.edu/pub/cs/distribs/clewis/HCI-Design-Book) in: [/pub/cs/distribs/clewis/HCI-Design-Book](ftp://ftp.cs.colorado.edu/pub/cs/distribs/clewis/HCI-Design-Book).