An Evaluation of the Chat and Knowledge Delivery Components of a Low-Level Dialog System: The AZ-ALICE Experiment

Robert P. Schumaker, Mark Ginsburg, Hsinchun Chen and Ying Liu
Artificial Intelligence Lab, Department of Management Information Systems
The University of Arizona, Tucson, Arizona 85721, USA
{rschumak, mginsbur, hchen, yingliu@eller.arizona.edu}

Word Count: 6326

Abstract

An effective networked knowledge delivery platform is one of the Holy Grails of Web computing. Knowledge delivery approaches range from the heavy and narrow to the light and broad. This paper explores a lightweight and flexible dialog framework based on the ALICE system, and evaluates its performance in chat and knowledge delivery using both a conversational setting and a specific telecommunications knowledge domain. Metrics for evaluation are presented, and the evaluations of three experimental systems (a pure dialog system, a domain knowledge system, and a hybrid system combining dialog and domain knowledge) are presented and discussed. Our study of 257 subjects shows approximately a 20% user correction rate on system responses. Certain error classes (such as nonsense replies) were particular to the dialog system, while others (such as mistaking opinion questions for definition questions) were particular to the domain system. A third type of error, wordy and awkward responses, is a basic system property and spans all three experimental systems. We also show that the highest response satisfaction results are obtained when coupling domain-specific knowledge together with conversational dialog.

Index Terms

Dialog Platform, Knowledge Delivery Evaluation, Domain-Specific Knowledge, chatterbot, ALICE, XML, AIML

Introduction

The World Wide Web is a vast distributed network of information, both credible and incredible. Myriads of users constantly access and try to make sense of the Web's content using a variety of tools, such as search engines and digital libraries. They are seeking to convert the information to knowledge – a subjective phenomenon in which their belief system is updated [5]. Since the times of Alexander the Great and the Great Library he established at Alexandria, a laudable goal has been a central repository of information to facilitate, to as broadly scoped an audience as possible, the transfer of knowledge. The explosion of Web content (both in sheer volume of pages and in supported format types, such as streaming media), coupled with the increasing ease of access to high speed bandwidth, means that researchers have a renewed focus on the design and implementation of large-scale knowledge transfer platforms. In its simplest form, this might be a digital library where access tools facilitate one-way flow of documents from the corpus to the end user. Another more dynamic approach is to allow the end users to be secondary contributors of information. This has been seen in electronic marketplaces of expertise such as Answer Garden [1] [2] and the Annotate! system, which allowed organizational workgroup-level document annotation to augment search engine results [11]. In situations where all participants are potential information donors, coordination mechanisms are critical between the primary content providers (authors, in the case of a digital library), secondary content providers (readers) and system administrators who are responsible to manage the system as it scales upward.

Given a specific domain of interest and its audience pool, there are two important aspects of a networked knowledge transfer platform. We have (a) knowledge delivery, where the system is able to answer a broad range of questions within the domain to the satisfaction of a broad range of the audience pool, and (b) knowledge acquisition, where the audience can contribute ideas to the system's knowledge base for the subsequent benefit of all. The second point contains an important social sub-problem: the intermediate validation, which can be reframed as an establishment of credibility, of the contributed ideas before they are accepted by the system.

One stream of work on these issues in the 1990s has focused on building a controlled vocabulary system, which is domain rich and can help users with specific goals [7, 8]. These approaches can lead to maintenance difficulty and semantic drift as vocabularies change and evolve [17].

Another approach is to utilize an Internet audience to build a large-scale information resource of interest, without specifying a priori the nature of the task an individual might have in mind when he or she accesses the resource. WordNet, OpenCYC, and Wikipedia, while adopting differing implementation philosophies [15] [20] all leverage large numbers of users to build the resource; a dictionary in the case of WordNet and a freely available encyclopedia in the case of OpenCYC and Wikipedia. In this paper we will not consider large-scale knowledge acquisition with its related social problem of contributor credibility and editorial effort, and focus instead on knowledge delivery.

One platform for knowledge delivery is a lightweight dialog system that will hold the user's attention with human-like responses. The ALICE system (Artificial Linguistic Internet Chat Entity) [21] was created by Richard Wallace uses an XML dialect called AIML (Artificial Intelligence Markup Language) to store patterns and responses upon encountering the pattern in a user dialog. Its standard distribution comes with approximately 24,000 patterns covering assorted geography, nature, and human interest facts. In addition, ALICE has mechanisms to engage the participant in conversational small talk and it supports access to third party networked resources via Rhino, a Java-like Javascript dialect. A test implementation linking the Java ALICE ProgramD to a set of Web services in a prototype portal was demonstrated in The Catacomb Project [10]. The distribution[1] also provides automated dialog logging, archiving, and visualization using XML and XSL.

This paper describes our use and evaluation of ALICE as a knowledge delivery and acquisition platform in experiments of conversational small talk and a specific Telecommunications domain. The evaluation depends on the development of new metrics, which we describe, to analyze ALICE's performance in the user sessions.

[1] There are various ALICE distributions freely available via http://www.alicebot.org (Python, C++, SETL, Java, and more).

In the remainder of the paper we describe related work, including quite a few implemented systems in the Literature Review. We then describe our research questions as well as our AZ-ALICE system. Following that we present our experimental design and an analysis of the experimental results. Finally, we conclude our study and provide details of further development for this platform.

Literature Review of Dialog Systems

Dialog system development and validation has been an active field of research for several decades. In this section we give an overview of prior work and highlight some key properties of the more relevant systems.

Dialog Systems can be divided into two main groups; the Theoretical and the Performance-driven [18]. The Theoretical or High-level systems involve symbolic reasoning and a deep understanding of user input. The Performance or Low-level systems forgo syntactic analysis and understanding for a much simpler pattern-matching algorithm. In both of these systems there are two elements that differentiate them, the level of analytical complexity and the level of complexity to understand context. High-level systems try to maximum these variables, while Low-level systems minimize them for performance gain. Hybrid, or mid-level systems, lie between and strike a balance between reasoning and performance.

A. High-level Dialog Systems

High-level dialog systems are sometimes referred to as Integrated Artificial Intelligence [7]. These are systems that possess planners, learning algorithms, speech recognition, and temporal reasoning. These systems also maintain state across user sessions, remembering which tasks have been accomplished and which ones remain. Because of their complexity, they typically focus on narrow bands of knowledge. Examples of these systems are TRIPS and TRAINS.

TRIPS, The Rochester Interactive Planner System, is an interactive spoken dialog collaborative transportation logistics planning assistant for crisis management [7]. This system behaves as a crisis

assistant on a mythical island, handling all the transportation-related logistics. Although this system does possess some contextual understanding and reasoning abilities, the system is not generalizable and takes a considerable amount of time to react to new stimuli.

TRAINS is the precursor to TRIPS and involves logistical routing of locomotives using a pre-defined map. As compared to TRIPS, TRAINS uses simpler route planning, and a simpler problem solving engine. However, TRAINS is also limited in scope and is incapable of handling unexpected environment changes.

B. Mid-level Dialog Systems

Mid-level dialog systems are those that can respond to a variety of requests about a task or domain. They will also typically have somewhat advanced reasoning abilities [9]. These types of systems can also take advantage of external resources such as Cyc or WordNet in their information gathering processes. The breadth of knowledge encompassed by these systems typically exceeds that of High-level systems but is less than Low-level systems. These systems do not typically maintain state between user sessions. Some examples of these systems are MALIN, Lucy, and Koko.

MALIN, the multi-modal application of the LINLIN programming language, is a bus time-table system that keeps track of bus routes and stations in Östergötland [9]. This system is less sophisticated than High-level systems which might plan, route, or remember previous itineraries. However, it does use natural language processing to answer simple user queries such as “Which bus passes the North gate”, and “Are there any bus stops near the Garden square” [9].

Lucy and Koko are both semantic interpreters of the English language [3]. Lucy attempts to process unfamiliar inputs into the CycL language while Koko performs the inverse function, taking CycL mappings and attempting to construct English sentences. Both of these systems rely upon the large Cyc knowledge base and thus operate in a much broader environment than MALIN.

C. Low-level Dialog Systems

Low-level dialog systems are those that seek to mimic conversation rather than understand it [13]. These systems employ simple algorithms to return dialog with minimal or no maintenance of state and thus no consideration of context. Because they are preprogrammed and have a large store of small talk canned responses, these systems can be entertaining in a large variety of conversational topic settings. Examples of these systems include MegaHAL and ALICE, which won the Loebner Prize in 2000, 2001, and 2004 for the most human-like computer.

MegaHAL, winner of the 1998 Loebner Prize for the most realistic human imitator, uses a method of Markov modeling to generate responses [13]. In this system a keyword in the user input is isolated and many Markov chains are assembled before and after the keyword. Through a process of relevance ranking, the response with the highest information content most relevant to the input is then returned to the user. This system learns from its interactions although it returns mostly nonsense replies.

ALICE uses simple pattern-matching of user input to predefined inputs, and then returns a response to the user [21]. This system is simple to administer and can be quickly adapted to new knowledge domains.

There are some interesting studies conducted using the ALICE chatterbot. One such study used an ALICE system to help Chinese university students practice their conversational English skills [14]. The study was qualitative in nature and used pre-existing conversational knowledge bases. The study itself was not very systematic, but did yield some interesting results. Users made a high preponderance of 'bad comments' about the system, and on average chatted for no more than 5 lines.

There are two studies that focused on using ALICE to augment or enhance an existing subject. One study focused on using ALICE systems to tutor students in Euclidean Geometry [12]. In this paper the author argues for the value of combining domain knowledge with conversational knowledge such that the system exhibits some form of personality and can respond to questions outside of the knowledge

domain. It is further posited that the conversational knowledge assists users in their decision making skills.

Another tutoring study focused on using ALICE as a course enhancement tool with Social and Political Theory knowledge [16]. This study found that most subjects used the system as a search engine rather than as a conversational partner. It was further concluded that their system was unable to function as a stand-alone tutor.

D. Challenges of Low-level Dialog System Analysis and Validation

There are several types of challenges in constructing and evaluating Low-level dialog systems. One of these is behavioral. Users might elect to insult the system and develop negative or sometimes abusive attitudes towards the system [6]. De Angeli conducted behavioral studies using ALICE and discovered that the friction arose from power differences between users and the system, where users were trying to exert their dominion of control over the system. From De Angeli's work it was found that some users will promote an abusive environment to establish their dominance. In addition, as stated by Pejtersen, users will sometimes use the system in unintended ways [16]. In that case, further exploration is needed to see if a) the system is lacking in topical knowledge, b) user training is inadequate, or c) it is an example of user abuse.

Another challenge is systemic. A simple pattern matching approach relies on pre-built input and output responses. Unfortunately this means that there will be some queries that the system cannot properly respond to. Using Zipf's law of distribution of English words, it has been found that 2,000 words cover 95% of the first words typed into the ALICE system [22]. Naturally, gaps remain and a challenge is to cover these gaps to improve user satisfaction.

A third challenge comes from the lack of systematic study in Low-level dialog systems, such as the ALICE system. Most prior ALICE studies have been qualitative and exploratory in nature, lacking objective, quantitative, field-based validation.

Research Questions

Dialog systems have two distinct modes of operation. First, they perform conversational interaction to engage the user in discussion. Second, they can be tailored to perform knowledge delivery by responding to specific queries and supply domain knowledge. Neither case has been well understood in the literature to date.

To address the research gap on the design and evaluation of Low-level dialog systems as conversationalists and as on-demand knowledge suppliers, we build an ALICE platform to deliver domain knowledge in addition to differing gradients of conversational ability. From the work of Han [12], we would expect that full conversation will perform best. We address these issues in the following research questions:

1. How accurate is a dialog system such as ALICE, and what types of error/deficiencies will occur?

2. How will a Dialog System perform with:

… strict conversational components?

… strict domain knowledge and sparse conversation?

… a hybrid approach involving full conversational and domain knowledge components?

Analysis of these research questions will point out the pros and cons of each approach and pave the way for knowledge delivery and knowledge acquisition extensions.

AZ-ALICE System Design

The AZ-ALICE dialog system is built on the Java-based ProgramD implementation that is freely available at www.ALICEbot.org. AZ-ALICE can be broken into five component parts: Chat User Interface, Chat Engine, AIML (Artificial Intelligence Markup Language) files, Logging, and a component for Evaluation.

The Chat User Interface uses an XML web page that allows the users to authenticate and converse with the system. The system also passes the authenticated username to the Chat Engine where it is stored as a state variable to allow for a more personalized conversation (e.g., Hello Bob389, how are you today?).

The Chat Engine contains the AIML knowledge base and conducts simple pattern-matching from the user's input to pre-defined AIML categories and then returns the corresponding response.

In our AZ-ALICE project we created a separate Telecommunications AIML knowledge set that contained 298 new categories to manage the Telecommunications knowledge domain. The Logging component enables the system to keep a user-specific record of discourse for future use. Finally, the Evaluation module allows users to evaluate and correct each system response. The evaluations are then later used for assessing the performance of each system.

Our AZ-ALICE implementation differed from ALICE ProgramD in several ways. Table 1 illustrates the differences between the two systems.

	Chat UI	Chat Engine	AIML	Logging	Evaluation
ALICE ProgramD	Uses XML to chat with users	Uses off the shelf ALICE ProgramD	Uses the freely available Standard AIML set	Logs everything to a monolithic XML Log file	None
AZ-ALICE	Uses a customized Perl skin for chat and evaluation purposes	Same as ALICE ProgramD	Depends on the bot as to whether it is Standard AIML customized Telecommunications knowledge	Keeps XML logs on a per user basis	Customized perl script that allows users to evaluate and suggest new patterns

Table 1. Differences between ALICE ProgramD and AZ-ALICE

Although the interfaces between the ALICE ProgramD and AZ-ALICE systems differ, the underlying information transfer between the interface and the chat engine remains the same.

To illustrate user/system interaction, consider the following real interaction complete with misspellings and poor grammar.

```
User: What is it like? I am a very emotional person so I can not
          imagin having no emotions.

AZ-ALICE: Do you think I should know that?  You are a emotional
          person so I can not imagin having no emotions?  Is
          that your job?
```

Here are the steps that led to the AZ-ALICE response.

- The User inputs the sentence into the Chat UI which then passes it along to the Chat Engine.
- The Chat Engine traverses a graphical representation of AIML nodes in memory.
- A match is found: the first sentence to the AIML category of "What is *" where '*' is treated as a wildcard.
- The chat engine passes the associated response of "Do you think I should know that?" back to the Chat UI and in addition echoes part of the user's original chat as a technique to keep the conversation going.

Experimental Design

The main thrust of this paper is to analyze the gradients of conversational knowledge in a domain-specific chatterbot. To do this, we created three chatterbot systems that differed only in their knowledge bases; BaseBot, TeleComm (Full), and TeleComm (Limited). BaseBot, the conversational and control chatterbot, used the Standard AIML files which consisted of 23,735 knowledge base entries. TeleComm (Full) contained 298 telecommunications-related definitions plus the same conversational knowledge as BaseBot. Finally, TeleComm (Limited) contained 298 telecommunications definitions and only a very rudimentary conversational capability of 3,892 knowledge base entries. We decided because of the limited amount of definitional knowledge in TeleComm (Limited), rather than run the potential risk of not returning any responses to participants and thus create alienation in our study, we instead chose to place TeleComm (Limited) in the role of providing full TeleComm knowledge and a limited array of conversational ability. From this position, we can test the effects of two different degrees of

conversational knowledge coupled with full domain knowledge. For our limited conversational knowledge, we selected a subset of conversational AIML that from the work of Richard Wallace, should be able to adequately answer 65% of the queries given to it. These files accounted for the 3,892 conversational knowledge base entries. Table 2 shows the breakdown of categories between each of the systems.

System Name	Std AIML	Telco AIML	Total # of Categories
BaseBot	23,735	0	23,735
TeleComm (Full)	23,735	298	24,032
TeleComm (Limited)	3,892	298	4,190

Table 2. The category breakdown between systems

From Table 2, the total number of categories for TeleComm (Full) does not equal the true summation of Standard and Telecommunications AIML entries. This was because one knowledge base entry overlapped between both knowledge sources. When this happens, the ALICE chatterbot will automatically omit the second instance thus decreasing the total number of categories available.

A. Performance Metrics

To gauge the various system performances, we measured the following evaluation variables; Correction Rate, Response Satisfaction, and Classification of User Inquiries.

Correction Rate is defined as a percent of system responses that were corrected by the user, divided by the total number of interactions typed into the system. Note that the act of correction requires time and effort on the users part (an opportunity cost) and thus the user may elect to bypass possible corrections he or she judges to be less pressing. This follows the conclusions drawn on prescriptive Restrictiveness Theory whereby limiting the amount of decision control allowed to the users, may serve to discourage system use [19].

Response Satisfaction is a measure of the appropriateness of system response given the context of the user query. This metric is evaluated by the users using a seven point Likert scale (1-strongly

disagree to 7-strongly agree). The aggregate Response Satisfaction number is then a summation of all the Response Satisfactions divided by the number of interactions typed into the system.

Classification of User Inquiries is the only measure that is not under the direct manipulation of users. In classification, user inquiries are scanned for any Telecommunications keywords that appear in the Telecommunications AIML file, and are then labeled as either conversational dialog or Telecommunications-related inquiries.

B. Participants

We assigned each experimental system to a different section of an Introductory Management of Information Systems course, such that participants would interact with only one of the chatterbots. Students, mostly freshman and sophomores, were instructed to interact with the system for approximately ½ hour and then evaluate all of the system responses for their particular session. They were also instructed to provide a Response Satisfaction score for each response, and were given the opportunity to provide an alternate system response to their particular query. Students were also instructed to focus their topic of conversation on Telecommunications knowledge; however, they were not forced to do so. All students were given an incentive by the award of bonus points for chatting a ½ hour and completing the evaluation component. Bonus points were awarded based upon full participation with the system, which included the final evaluation of chatterbot responses. Participation in the study was voluntary and students were permitted to chat from any computer terminal they wished which allowed them to span multiple chatting sessions. Student subjects were selected based upon their availability and represented a computer-literate demographic that is likely to use chatterbot entities. Table 3 shows a breakdown of the number of participants for each of the three systems.

System Name	Number of Study Participants	Number of Interactions
BaseBot	74	9,751
TeleComm (Full)	91	10,179
TeleComm (Limited)	92	10,005

Table 3. Study Participants by System

Because of some concern that some participants may not be fully interacting with the system, we analyzed the number of interactions that each user contributed. From there we determined that the majority of participants were using the system according to our expectations. Table 4 shows a breakdown of Interaction groupings between the three chatterbots. The average number of Interactions per chatterbot are provided as an aid.

Number of Interactions	BaseBot	Tele-Lmtd	Tele-Full
Average number of Interactions	131.8	108.8	111.9
Less than 50	9	13	10
Between 50 and 99	23	30	33
Between 100 and 149	24	33	25
Between 150 and 199	6	7	18
Between 200 and 249	5	7	1
Between 250 and 299	3	1	3
Greater than 300	4	1	1

Table 4. Interaction Breakdown

Experimental Results and Discussion

The results of the study are presented in Table 5. We discuss the results of Table 5 in light of our original research questions.

	Conversational dialog				Telecommunications			
		Correction Rate	Response Satisfaction			Correction Rate	Response Satisfaction	
	Percent of Total Usage		Mean	Std Dev	Percent of Total Usage		Mean	Std Dev
BaseBot	99.2%	17.4%	4.3040	1.5033	0.8%	28.4%	4.4321	1.6578
TeleComm (Full)	97.9%	16.1%	4.1565	1.4984	2.1%	15.1%	4.6376	1.7017
TeleComm (Limited)	95.8%	22.3%	3.8225	1.6104	4.2%	21.1%	4.4431	1.9250

Table 5. Analysis of AZ ALICE System Results

† - All pairwise comparisons were found to be significant with p-values < 0.001

A. Research Question 1

How accurate is a dialog system such as ALICE, and what types of errors/deficiencies occurred?

To answer the first part of this question we need to take a look at each of the system's total Correction Rate values irrespective of conversational dialog and Telecommunications knowledge, and assume that system accuracy can be modeled as one minus the Correction Rate. We acknowledge that measuring Accuracy as (1 – Correction Rate) can be problematic in an absolute sense, however since our measure is consistently applied between chatterbots, the relative measure is interpretable and has meaning. Table 6 provides a summary of system accuracy.

System Name	Correction Rate	Accuracy
BaseBot	17.5%	82.5%
TeleComm (Full)	16.0%	84.0%
TeleComm (Limited)	22.2%	77.8%

Table 6. System Accuracy as a measure of Correction Rates

AZ-ALICE's total accuracy figure hovers near 80% which might be satisfactory in some settings (e.g., conversation) and unsatisfactory in others (e.g., information retrieval). An example conversational interaction is shown below.

```
User: He did a good job teaching you.

AZ-ALICE: Yes he did didn't he.
```

For the second part of our research question, *what types of errors/deficiencies occurred*, we take a qualitative approach by categorizing the most prominent types of response errors that were observed in the system logs. Table 7 outlines the observed chatterbot response errors in both BaseBot and TeleComm (Limited).

Overall	BaseBot	TeleComm
Nonsense responses	35.7%	34.5%
Spurious user corrections	28.6%	6.9%
Wordy and awkward responses	28.6%	13.8%
Application or Opinion type questions	7.1%	20.7%
Incomplete lexicon of terms	0.0%	24.1%

Table 7. Observed chatterbot response errors

B. Conversational Dialog Categories

For BaseBot, the most common error categories are; nonsense responses, spurious user corrections, and wordy and awkward responses.

B-1. Nonsense responses

For nonsense responses, this type of error generally occurred when the system reflected the user input back to the user and either selects improper pronouns or uses a sentence fragment in an improper context. The following interaction is a good example of this type of deficiency.

```
User: Did you buy your car yourself?

AZ-ALICE: No I didn't buy your car yourself.  What's buy your car
          yourself like?
```

In this interaction the system incorrectly used the pronoun of 'yourself' and ended up with a nonsense reply.

B-2. Spurious User Corrections

The second conversational dialog error type, spurious user corrections, occurred when users offered unnecessary and spurious corrections to the system responses – corrections which do not benefit users of the system going forward. The following snippet illustrates this category.

```
User: do you like metallica?

AZ-ALICE: Not since they put pressure on Napster.

User Corrected Response: YES they are the BEST band in the
          history of humankind.
```

This interaction shows that the original response makes sense in the context of user input, however, the user decided to insert a more entertaining response for future system interactions. It was interesting to note that this category would usually have a higher than average Response Satisfaction score.

B-3. Wordy and awkward responses

For wordy and awkward responses, the AZ-ALICE system glued together non-sequitur sentences and sentence fragments when confronted with complex sentences containing multiple subject/verb pairs. Here is an example.

```
User: Yeah everyone else is sorry when they do hear me sing.

AZ-ALICE: Yeah I know what you mean about that.  That was not a
          yes or no question.  That's a rather sweeping
          generalization.
```

The above interaction shows the system picking three separate matches keying in on different words from the user's input.

C. Telecommunications categories

The telecommunications knowledge domain had its own class of errors and deficiencies; users would ask application or opinion oriented questions rather than definitional ones, the system held an incomplete lexicon of terms, or the system would provide nonsense responses.

C-1. Application or opinion type questions

The error category of application or opinion type questions houses all of the ways students would use telecommunications keywords outside of a definition type context.

```
User: Is internet telephony useful?

AZ-ALICE: The transmission of voice/data over long distances.
```

In this interaction, the system recognized the term 'telephony' and returned the predetermined AIML definition. Even though subjects were instructed to ask definition type questions, this category of error was nevertheless widespread.

C-2. Incomplete lexicon of terms

In the telecommunications error category of incomplete lexicon of terms, there was some discourse in which users would ask obvious telecommunications definitions for which the system possessed a response, however, the system would instead key in on a different part of the user input and ignore the telecommunication term.

```
User: How does an IP address work?
AZ-ALICE: How should I know it does?
```

In this example, the system matched 'How does *' rather than '* IP *'. These types of mismatch problems, while correctable by AIML maintenance, are common to a conversational system.

C-3. Nonsense responses

The error category of nonsense responses is the only category that spans both the conversational dialog and Telecommunications knowledge domains. Nonsense responses are a problem with the ALICE ProgramD chat engine, particularly when the elicitation of particular domain-related answers are desired. Again, this is the trade-off between conversational entertainment and terse knowledge delivery. The other interesting item to note was that both of the other domain-related system problems stemmed from an incomplete set of domain answers.

D. Research Question 2

How will a dialog system perform with varying degrees of Conversational dialog and Domain Knowledge?

System Name	Conversational dialog	Telecommunications domain knowledge	Percentage Gain
BaseBot	4.3040	4.4321	3.0%
TeleComm (Full)	4.1565	4.6376	11.6%
TeleComm (Limited)	3.8225	4.4431	16.2%

Table 8. Expanded view of Response Satisfaction scores from Table 5

† All pairwise comparisons were found to be with p-values < 0.001

The second research question is best answered by inspecting the user Response Satisfaction and Correction Rate numbers between the three systems. We were not as interested in the absolute values or motivations behind the Response Satisfaction scores, but more interested in the relative significance between them. Table 8 is an expanded view of the Response Satisfaction scores from Table 5. All three systems were rated higher in Telecommunications knowledge (4.4321, 4.6376, and 4.4431) as compared to conversational dialog (4.3040, 4.1565, and 3.8225 respectively). The most notable differences came from the two TeleComm systems whose Telecommunications domain knowledge was a double digit percent gain in Response Satisfaction as compared to conversational dialog. TeleComm (Full) showed an 11.6% increase while TeleComm (Limited) did even better with a 16.2% gain. Because of the large number of interactions, these values were found to be statistically significant. It would appear that users preferred the limited set of domain terms to conversational dialog. However, in defense of eliminating conversational dialog completely, the two systems that implemented full conversational dialog, BaseBot and TeleComm (Full) both performed better in conversational dialog, 4.3040 and 4.1565 respectively, than did TeleComm (Limited) with its relatively low Response Satisfaction score of 3.8225. This finding is of practical significance because domain-specific systems appear to perform better with a full complement of conversational dialog patterns to augment their domain-specific knowledge which is consistent with the observations of Han [12].

System Name	Conversational dialog	Telecommunications domain knowledge
TeleComm (Full)	16.1%	15.1%
TeleComm (Limited)	22.3%	21.1%

Table 9. Abbreviated view of the Corrected Responses of Table 6

It is interesting to note that the conversational dialog in the two TeleComm systems also had a higher Correction Rate than their domain knowledge counterparts, as shown in Table 9. TeleComm (Full) conversational dialog had a Correction Rate of 16.1% as compared to its Telecommunication knowledge Correction Rate of 15.1%. Likewise, TeleComm (Limited) had a conversational dialog Correction Rate of 22.3% as compared to its domain knowledge Correction Rate of 21.1%.

Discussion

Intuitively, it would seem that as Correction Rate drops the Response Satisfaction level should increase. However, our BaseBot proved to be an exception to this rule as it had a much higher Correction Rate and Response Satisfaction score in its Telecommunications domain (28.4% and 4.4321 respectively), than it did in conversational dialog (17.4% and 4.3040 respectively). As can be seen from Tables 8 and 9, Correction Rate and Response Satisfaction moved in the same direction rather than opposite ones. One explanation was that users liked the Telecommunications-related system responses but instead chose to amend them to reflect an elaboration or entertainment value. However, in a further examination of the results this did not appear to be the case. It was found that users were instead creating answers in the vacuum of BaseBot's Telecommunications knowledge and correcting compound and wordy responses. Another explanation is that the size of interactions with BaseBot's Telecommunications are simply too small (23 corrected Telecommunications-related responses) to make any kind of solid presumption. Although the limited rule set did make a fairly sizable and statistically significant impression on the two TeleComm systems, the level of Telecommunications interaction in proportion to conversational dialog was low. Subjects were instructed to limit themselves to Telecommunication definitions, but most students instead found the conversational dialog functionality after the first few telecomm interactions. It was interesting to find that even in the conversationally constricted environment of TeleComm (Limited), where conversational knowledge outnumbered telecommunications by nearly 13-1, subjects insisted on talking about non-telecomm topics, where conversational dialog accounted for 95.8% of TeleComm (Limited)'s interactions.

The experimental results included some examples of repetitive user input (just to satisfy the extra-credit incentive) and other examples of vulgar language. However, this may not be a limitation as much as expected student interaction with a system for a certain demographic subset.

We further investigated whether participants were performing corrections as needed by setting up a three judge panel. This panel was asked to independently determine whether corrections were required

on a random selection of user interactions. However, we ended up with poor inter-coder reliability because of the subjective nature of accepting responses.

Conclusions and Future Directions

In our experiment, using the results of Response Satisfaction, we found that domain knowledge is more effective in a chatterbot environment to obtain domain-specific knowledge than conversational knowledge alone. This comes from the breadth of knowledge that conversational knowledge would have to cover as opposed to the limited scope that domain knowledge encompasses. Further, we found that domain-specific knowledge coupled with conversational knowledge yields that highest response satisfaction scores. We feel that conversational dialog, while not strong on its own, is an important element in a domain-centric chatterbot.

We found that the AZ-ALICE system is better suited to answering specific domain-related queries than performing as a general conversationalist. In our analysis of the domain dependent systems, it was found that both TeleComm systems had higher Response Satisfaction (4.6376 and 4.4431 respectively) and lower Correction Rates within their knowledge domain (15.1% and 21.1% respectively) than in the conversational dialog arena (4.1565 and 3.8225), and (16.1%, and 22.3%) respectively. We further found that conversational dialog is an integral piece of a system's repertoire. This element handles those user queries that either fall outside of the bounds of the domain knowledge or are querying domain-specific knowledge that has not yet been entered into the system. Thus, a knowledge delivery system performs strongest when its domain knowledge is coupled with a storehouse of conversational dialog.

We believe our research made several contributions. In our study of the AZ-ALICE system, we addressed the problem of evaluating a low-level dialog system's ability to bestow domain knowledge in a very systematic way, which lead to quantifiable results, previously missing in the literature. We also created several new components that incorporated evaluation/feedback components in the AZ-ALICE system that allowed subjects to test and evaluate the AZ-ALICE project on a large scale (29,935

input/responses across all three systems). Our study also focused on the contrast of using conversational dialog versus a specific domain knowledge set. Although there is some literature that broaches the subject of using domain-specific knowledge in systems [12, 16], our research opens more avenues of research in this area. Future ALICE research should focus on specific categories of knowledge that participants are most likely to correct as well as which knowledge categories have higher Response Satisfaction scores. Finally, measuring the quality of user-suggested knowledge would be worthwhile.

The ALICE dialog system is promising as extensions readily come to mind to target both knowledge delivery and acquisition. The Java front-end component provides avenues for multimedia support and flexible connector code to third party network resources, i.e., Web Services. In addition, the project can be extended on the front-end with advanced natural language parsing techniques [4]. One avenue we are actively pursuing is the adaptation of AZ-ALICE into a Terrorism Support platform. We will build this platform, connect it to a wide variety of network resources, and evaluate it. This should prove an important resource to aid the victims of terror while increasing our understanding of the pros and cons of low-level dialog systems as broadly scoped knowledge platforms.

Acknowledgments

This work was supported in part by the NSF, ITR: "COPLINK Center for Intelligence and Security Informatics Research - A Crime Data Mining Approach to Developing Border Safe Research," Sept. 1, 2003 - Aug. 31, 2005.

We would like to thank Edna Reid and Larry Valida for their insightful comments. We would also like to thank the MIS 111 Instructors, Surya Pandruvada and Dennis Viehland, for assistance in setting up the experiment. Finally, we would like to thank the following pilot testers: William Paul Burger, Jacquelyn Calvo, Alejandrina Gonzalez, Eric Impraim, Jennifer M. Lee, Kim Sierra, and Kristen Smyser.

References

1. Ackerman, M. Answer Garden: A Tool for Growing Organizational Memory, Massachusetts Institute of Technology, 1994.

2. Ackerman, M.S. Augmenting Organizational Memory: A Field Study of Answer Garden. *ACM Transactions on Information Systems*, *16* (3). 203-224.

3. Barnett, J., Knight, K., Mani, I. and Rich, E. Knowledge and natural language processing. *Communications of the ACM*, *33* (8). 50-71.

4. Choi, D.-Y. Enhancing the power of Web search engines by means of fuzzy query. *Decision Support Systems*, *35* (1). 31-44.

5. Davenport, T.H. and Prusak, L. *Working Knowledge: How Organizations Manage What They Know*. Harvard Business School Press, Boston, MA, 1998.

6. De Angeli, A., Johnson, G.I. and Coventry, L., The unfriendly user: exploring social reactions to chatterbots. in *Proceedings of The International Conference on Affective Human Factors Design*, (London, 2001), Asean Academic Press.

7. Ferguson, G. and Allen, J.F. TRIPS: An Integrated Intelligent Problem-Solving Assistant. in *American Association for Artificial Intelligence*, 1998, 567-572.

8. Ferguson, G., Allen, J.F., Miller, B.W. and Ringger, E.K., The Design and Implementation of the TRAINS-96 System: A Prototype Mixed-Initiative Planning Assistant. in *Proceedings of the Third Conference on Artificial Intelligence Planning Systems*, (1996), 70-77.

9. Flycht-Eriksson, A. and Jönsson, A., Dialogue and Domain Knowledge Management in Dialogue Systems. in *Proceedings of the First SIGdial Workshop on Discourse and Dialogue*, (2000), 121-130.

10. Ginsburg, M., The Catacomb Project: Building a User-Centered Portal the Conversational Way. in *WIDM 2002 (Fourth International Workshop on Web Information and Data Management)*, (McLean, VA, 2002), ACM.

11. Ginsburg, M., Ajit Kambil, Annotate: A Knowledge Management Support System. in *HICSS-32*, (Hawaii, 1999), IEEE.

12. Han, S. and Kim, Y., Intelligent Dialogue System for Plane Euclidean Geometry Learning. in *International Conference on Computers in Education*, (Seoul, Korea, 2001).

13. Hutchens, J.L. and Alder, M.D., Introducing MegaHAL. in *Proceedings of the Human-Computer Communication Workshop*, (1998), 271-274.

14. Jia, J. The Study of the Application of a Keywords-based Chatbot System on the Teaching of Foreign Languages, University of Augsburg, Augsburg, Germany, 2002.

15. Lenat, D., Miller, G. and Yokoi, T. CYC, WordNet, and EDR: critiques and responses. *Communications of the ACM*, *38* (11). 45-48.

16. Moore, R. and Gibbs, G. Emile: Using a chatbot conversation to enhance the learning of social theory, Univ. of Huddersfield, Huddersfield, England, 2002.

17. Pejtersen, A.M. Semantic Information Retrieval. *Communications of the ACM*, *41*. 90-92.

18. Russell, R.S. *Language Use, Personality and True Conversational Interfaces*. Univ of Edinburgh, Edinburgh, 2002.

19. Silver, M.S. Decision Support Systems: Directed and Nondirected Change. *Information Systems Research*, *1* (1). 47-70.

20. Wagner, C. WIKI: A Technology for Conversational Knowledge Management and Group Collaboration. *Communications of the AIS*, *13*. 265-289.

21. Wallace, R.S. The Anatomy of A.L.I.C.E. in *A.L.I.C.E. Artificial Intelligence Foundation, Inc.*, 2004.

22. Wallace, R.S. The Elements of AIML Style, 2003.

Robert P. Schumaker is a third-year PhD student in the Department of MIS at The University of Arizona. He received his undergraduate degree in Civil Engineering from the University of Cincinnati and an MBA degree in Management and International Business from the University of Akron. His interests include Stock Price Prediction, Natural Language systems and Textual Analysis techniques.

Dr. Hsinchun Chen is McClelland Professor of Management Information Systems at the University of Arizona and Andersen Consulting Professor of the Year (1999). He received the B.S. degree from the National Chiao-Tung University in Taiwan, the MBA from the SUNY Buffalo, and the Ph.D. degree in Information Systems from New York University. He is author of seven books and more than 120 SCI journal articles covering intelligence analysis, data/text/web mining, digital library, knowledge management, medical informatics, and Web computing.

Dr. Mark Ginsburg is a former Professor of Management Information Systems at the University of Arizona. He received his B.A. in Biology from Princeton, M.A. in Pharmacology from Columbia, MBA from NYU, and PhD in Information Systems from NYU. He was also the recipient of the W. Edwards Deming Award in 1991. Dr. Ginsburg is also an International Chess Master, ranked 49th in the USA on the August 2000 rating list. Down from a best of 27th in 1993.

Ying Liu is a third-year PhD student in the Department of MIS at the University of Arizona. He received his B.E in EE from Xian Jiao Tong University and M.S in computer science from USC. His present interests include designing and implementing new IT applications to address real life requirements in knowledge management, data mining and business intelligence.