Towards Natural Clarification Questions in Dialogue Systems

Svetlana Stoyanchev[1] and **Alex Liu**[2] and **Julia Hirschberg**[3]

Abstract. Clarifications are often necessary for maintaining human-human as well as human-machine dialogue. However, clarification questions asked by Spoken Dialogue Systems (SDS) are very different from clarification questions asked in natural human interaction. While in human-human dialogues, speakers ask *targeted* questions using contextual information, SDS ask generic clarifications such as *please repeat* or *please rephrase*. We propose and evaluate a new strategy for creating more natural clarification questions. We model natural language clarification question generation rules based on human-generated behavior. We describe results of a user study to evaluate our automatically generated questions and show that subjective scores of the automatically generated questions are comparable to scores for human-generated questions — with some of the automatic rules even outperforming human-generated questions.

1 INTRODUCTION

Clarification questions are essential for successful dialogue communication. Without clarification, dialogue participants risk missing information and failing to achieve mutual understanding. The ability to clarify is especially important when there is a communication interference such as noisy environment, poor telephone connection, or impaired language proficiency of communication partners (such as a child or a new language learner). These issues also arise for Spoken Dialogue Systems (SDS), which rely upon often errorful Automatic Speech Recognition (ASR) to understand user input.

According to the four-level model of communication developed independently by Clark and Allwood, clarifications in human-human dialogue aim to resolve a miscommunication on one of the four levels: 1. securing attention, 2. hearing an utterance, 3. interpreting the meaning of an utterance, and 4. deciding which action is appropriate [7, 3]. In this work we model clarification questions for an automatic SDS. SDS provide a natural language interface for applications such as information systems, tutoring systems, technical support systems, or situated virtual assistant systems [15, 1, 5]. The four levels of communication where misunderstanding may occur are relevant to four components of an SDS. Level 1, securing attention, is implicit in most SDS as systems usually assume that a user is focusing attention on a system they have chosen to interact with. Level 2, hearing an utterance, is performed by the ASR component that converts an utterance from speech to text. Level 3, interpreting the meaning, corresponds to the Natural Language Understanding component (NLU) that derives semantics from text of an utterance. And level 4, deciding on an appropriate action, corresponds to the action the system decides to take as a result of the interpreted user input, such as accessing a database, moving of a robot arm, or generating a spoken response to the user. In SDSs, many errors are caused by failure at level 2, the ASR component; this challenge is similar to the one faced by human speakers communicating through a noisy channel. In this work, we address the problem of clarifying misunderstandings caused by ASR errors.

The recognition accuracy of ASR components in SDSs varies widely. For example, the word error rate (WER) in CMU's Let's Go system, a system which provides bus information over the phone in Pittsburgh is around 50%, while the WER in the English component of a speech-to-speech translation system developed in the DARPA Transtac program (SRI's IraqComm) is only 9% [20, 2]. Regardless of the amount of error, however, all SDS require strategies to detect errors in recognition and to allow them to recover from such errors.

Most dialogue systems today employ *generic* clarification strategies asking a speaker to repeat or rephrase an entire utterance. Human speakers, on the other hand, employ different and diverse clarification strategies in human-human dialogue. In this work we distinguish *generic* and *targeted* clarification questions. Consider the following exchange:

A: When did the problems with [*power*] start?
B: The problem with what?
A: Power.

Speaker B asks a *targeted* question where part of the utterance recognized correctly is repeated as context for the portion believed to have been misrecognized or simply unheard. Examining human clarification strategies, Purver [19] (following Ginsburgh and Cooper [9]) examines *reprise* questions, a type of a targeted clarification, which echo interlocutor's utterance as Speaker B's query above. In human-human dialogues, reprise questions are much more common than non-reprise questions; Purver found that 88% of human clarifications were reprise questions. In our domain, we also found that the participants frequently ask targeted reprise questions.

Generic questions are simply requests for a repetition or rephrasing of a previous utterance, such as *What did you say?* or *Please repeat*. Such questions crucially do **not** include contextual information from the previous utterance. *Targeted* question, on the other hand, explicitly distinguish the portion of the utterance which the system believes has been recognized from the portion it believes requires clarification. Besides requesting information, a clarification question also helps ground communication between two speakers by providing feedback which indicates the parts of an utterance that have been understood. In the above example, Speaker B has failed to hear the word *power* and so constructs a clarification question using a portion of the correctly understood utterance to query the portion of the utterance they have failed to understand. Speaker B's *targeted clarification question* signals the location of the recognition error to Speaker A. It achieves grounding by indicating that the hearer understands the speaker's request for information about *'the problem'* but has missed

[1] AT&T Labs Research, email: sveta@research.att.com (this work was done while at Columbia University)
[2] Columbia University, email: al3037@columbia.edu
[3] Columbia University, email: julia@cs.columbia.edu

the problem description. In this case, Speaker A is then able to respond with a minimal answer to the question — filling in only the missing information .

The use of generic *please repeat/rephrase* strategies by SDS to clarify non-understandings is neither natural nor optimal for communication with human users. SDS use this types of questions because they are much easier to construct than targeted questions and can be used for any understanding failure and do not require the location of the likely ASR error. While targeted clarification questions are easier to construct in form-filling systems, where the type of information the user is attempting to convey is easier to infer, it is much more challenging to create a targeted question for systems that handle unrestricted speech. A form-filling system requires a user to specify particular types of single words and phrases, e.g. *Where are you departing from?*. If a user's response is not recognized, the system may construct a targeted clarification *Departing from where?* simply from the knowledge that user was presumably trying to fill in the *departure* field. To construct a targeted clarification question for a system accepting free speech, a system must first determine which part of an utterance it believes contains an error. It must then construct an appropriate question based upon information in the correctly recognized part of an utterance. Use of a reprise clarification question also poses the challenge of determining how to merge the user's original sentence with their answer to the clarification question to produce the corrected system input. In this work, **we explore the use of targeted, and in particular, reprise, clarification questions** in SDS that accept unrestricted speech. Such systems are becoming more common today in interfaces to virtual agents, robots control spoken interfaces, speech-to-speech translation systems, and automatic troubleshooting systems. We evaluate users' subjective perception of automatically generated questions when an error segment has been successfully detected. We examine a corpus of clarification questions constructed by people in response to observing a sentence with missing information which we have collected for our research. We analyze how human speakers construct clarification questions about missing information and we create a set of rules for the automatic generation of targeted clarification questions. Our goal is the automatic construction of more natural clarification questions and improving the efficiency of automatic error recovery by allowing a user to correct a targeted poriton of their utterance. We hypothesize that this method also improves dialogue coherence by implicit grounding of recognized information. We evaluate the quality of our automatically generated questions with human subjects.

The paper is organized as follows. In Section 2, we describe previous research on error handling in dialogue. In Section 3 we summarize our corpus of human clarification questions. We present an automatic algorithm for constructing targeted clarification questions in Section 4 and its evaluation with human speakers in Section 5. We conclude in Section 6 and discuss future directions.

2 RELATED WORK

Past research on SDS has addressed the question of error handling and recovery in human-human, Wizard-of-Oz, and in automated systems. Skantze [24] collected and analyzed user responses to ASR errors in a direction-giving domain in Swedish, using a speech recognizer to corrupt human-human speech communication in one direction. Williams and Young [28] performed a Wizard-of-Oz study in a tourist information dialogue system in which recognition errors were systematically controlled. Koulouri and Lauria [12] performed another Wizard-of-Oz study in a human-robot instruction-giving domain with the "wizard" playing a role of a robot with restricted communication capabilities. In all of these studies, results indicate that, when subjects encounter speech recognition problems, they tend to ask task-related questions, providing feedback to the system and confirming their understanding of the situation. These studies also find that speakers rarely give a direct indication of their misunderstanding to the system, irrespective of the system's WER. Williams and Young's findings suggest that, at moderate WER levels, asking task-related questions appears to be a more successful strategy for error recovery than direct signaling of the error itself.

Some previous research has also focused on miscommunication detection [13, 14]. SDSs distinguish between different *confidence levels* in ASR hypotheses in their dialogue management strategies: Hypotheses in which confidence is high are accepted. Those with lower confidence scores may be confirmed, either explicitly (*I heard you say X. Is that correct? Please say yes or no.*) or implicitly in the context provided in a subsequent query (*When would you like to travel to Boston?*). Implicit confirmation questions are based upon system knowledge of the dialogue state and dialogue context [23, 6]. Lower confidence hypothesis may also evoke a generic clarification question (*I'm sorry. I didn't understand you. Could you please repeat* or *Could you please rephrase that?*). The latter case is termed *rejection* of a user input. Researchers have found that the formulation of system prompts has a significant effect on the success of SDS interaction. Goldberg et al. [10] find that formulation of a clarification question affects user frustration and consequent success of clarification subdialogue.

In our own previous research we have studied the detection of the location of ASR errors in utterances to support the creation of targeted clarification questions [26, 17]. We have also studied how users construct clarification questions, investigating when they choose to ask questions and which types of questions they are likely to propose when they choose to ask a questions [22]. We have also applied machine learning techniques to predict both user decisions to stop and ask a question or to continue the dialogue without asking for clarification and user decisions to ask a targeted clarification question [25]. Our goal is to design targeted clarification strategies for handling errors in automatic spoken dialogue systems when appropriate.

3 HUMAN CLARIFICATION QUESTIONS

In our previous research we conducted an experiment to study how humans ask clarification questions [22]. We collected questions using American English utterances from an open-domain Speech-to-Speech (S2S) translation system [2], the IraqComm corpus. The data were collected by NIST during seven months of evaluation of S2S translation systems exercises for the DARPA TRANSTAC program held between 2005 and 2008 [27]. The corpus contains acted dialogues between English and Iraqi Arabic speakers. The speakers use natural grammar and the vocabulary and domain are unrestricted. This corpus included manual transcriptions as well as recognition hypotheses. Our goal was to create a more natural clarification strategy for the Dialogue Manager of the SRI ThunderBolt S2S system.

Our data collection experiment was text-based and used the Amazon Mechanical Turk [4] (AMT) crowdsourcing tool. Each subject was asked to indicate how they would respond to a sentence containing some unknown word or words indicated by XXX, e.g. *how many XXX doors does this garage have?* . The sentences were created from actual ASR hypotheses in the IraqComm corpus, in which XXX was substituted for misrecognized words. These errors were identified by aligning the IraqComm reference transcriptions with the corresponding ASR hypotheses and finding mismatches between them. Although different ASR systems (or even the same system with a different acoustic or language model) can make different recognition errors, using actual errors is a more realistic approach than simulating ASR errors.

We asked the AMT workers to answer a set of questions about their perception of the misrecognized utterance as well as how they would try to recover the missing information. Table 1 shows a sample sentence and questions presented to the participants. In our previous research, we analyzed how the subjects came up with a decision on the question type that they would ask [25]. Each input sentence was presented to three AMT workers.

	Sentence presented to a participant: **how many XXX doors does this garage have**
	Questions to a participant
1.	Is the meaning of the sentence clear to you despite the missing word?
2.	What do you think the missing word could be? If you're not sure, you may leave this space blank.
3.	What type of information do you think was missing?
4.	If you heard this sentence in a conversation, would you continue with the conversation or would you stop the other person to ask what the missing word is?
5.	If you answered "stop to ask what the missing word is", what question would you ask?

Table 1. *Questions given to annotators.*

We collected 794 targeted clarification questions, 72% of all clarification questions asked. Figure 3 shows the distribution of question types in our data. Since we were interested in collecting a corpus of targeted questions, we instructed the participants to ask 'the most specific' question whenever possible. As a result, the proportions of targeted vs. generic reflect subjects' *ability* to ask a targeted question rather than their preference. The proportion of *targeted* questions asked varies with subjects' hypothesis of the POS of the missing target. Targeted questions were more frequent than *confirm* and *generic* questions combined for all POS tags except prepositions and question words. This indicates that annotators were using hypothesized POS to determine whether a targeted question was possible or not.

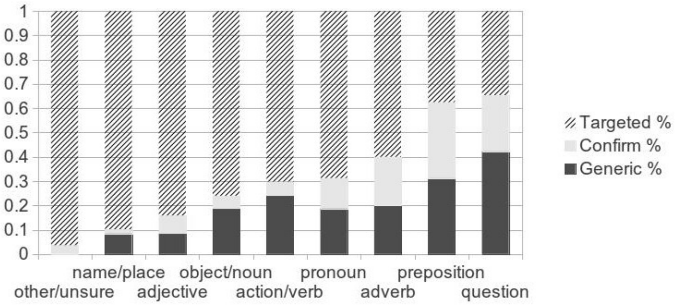

Figure 1. *Distribution of decisions for targeted, confirmation, and generic question types.*

In the work presented in this paper, we analyze the targeted clarification questions written by AMT workers in order to create rules to use in the automatic generation of targeted clarification questions.

4 GENERATING TARGETED CLARIFICATION QUESTIONS

Using the human-generated targeted questions discussed in Section 3 as a development set, we constructed rules for the automatic generation of targeted clarification questions. We create questions from sentences in an approach similar to that used in the First Question Generation Shared Task (QGSTEC) [21]. Systems in QGSTEC aimed to generate factual questions of a given type from an input sentence. In our task, we are given a sentence with a missing segment. Our goal is to construct a clarification question that prompts a user to respond by repeating the missing part of an utterance. Unlike the systems participating in the QGSTEC task, we do not know whether the missing segment corresponds to a 'who', 'what', 'where' or another type of question. Similarly to factual question generation systems [11], we construct syntactic and surface realization rules for transforming an input sentence into a question.

We model our question generation rules on the human-generated questions collected from our AMT experiments. The transcripts of these questions were annotated with part-of-speech tags, named entity tags, dependency tags, and dependency relations using an automatic dependency parser tool [16]. In an actual SDS, we would have access to automatically created tags like these. However, such tags on errorful sentences might be more errorful. We constructed five types of question generation rules based on the information in these annotations: 1) *generic* (R_WH); 2) *syntactic* (R_VB); 3) *noun-modifier* (R_NMOD); 4) *error word position* when an error occurs in the beginning of a sentence (R_START); 5) *named entity*(R_NE). The rules are summarised in Table 2.

The generic question generation rule R_WH is based on our findings in [26] that people create many targeted questions by simply replacing a missing word with a *wh-word*. R_WH is constructed from the recognized portion of the utterance before the identified error word (Utt[1..error-1]) concatenated with the *what* word. This simple rule works surprisingly well for many situations, as our evaluation shows. However, in some cases, it produces an incoherent clarification question. For example, for an input sentence like 1 below, R_WH would generate a marginally coherent question 1-A:

1 *When was the XXX contacted?*
*1-A *When was the what?*
1-B *When was what contacted?*

A more colloquial questions might include the following context *contacted* and possibly eliminate the article before *what* as in 1-B.

From such examples in our data, we find that the post-error context is most important when it contains a verb. In example 1, the word following an error is the verb *contacted*. Syntactic rule R_VB aims at producing a more coherent question than R_WH depending upon the location of the utterance's verb. R_VB1 and R_VB2 generate clarification questions by replacing the error word with *what* and including both pre- and post-error contexts. R_VB1 applies when a verb occurs after an error and when both the verb and the error term share a syntactic parent. R_VB1 will appropriately match the sentence the example above where the error word XXX and the verb *contacted* share the same parent, VP. It will correctly fail to match the sentence in example 2, as the error word and the following verb do not share a syntactic parent.

2 *As long as everyone stays XXX we will win.*
*2-A *As long as everyone stays what we will win?*
2-B *As long as everyone stays what?*

#	Application Rule	Question Generation Rule	Original Sentence	Clarification Question
R_WH - a generic *what* questions				
	default	Utt[1..error-1] what?	The doctor will most likely prescribe XXX	the doctor will most likely prescribe WHAT?
R_VB - a syntactic *what* questions				
1	VB after error & VB and error share a parent	Utt[1..error-1] what Utt[error+1 ..end]?	When was the XXX contacted?	When was WHAT contacted?
2	VB after error & error is followed by POS=to	Utt[1..error-1] what Utt[error+1 ..end]?	we need to have XXX to use this medication	we need to have what to use this medication?
3	POS(error-1)=TO; POS(error) = VB	Utt[1..error-1] do what?	we will not be able to XXX	we will not be able to do what?
4	POS(error-1)=MD; POS(error) = VB	Utt[1..error-1] do what?	if you stay quiet you can XXX down on any insurgents	if you stay quiet you can do what?
R_NMOD - *which* questions				
	DEP TAG error = NMOD & parent POS = NN | NNS	which <parent word>	Do you have anything other than these XXX plans	Which plans?
R_START - *what about* questions				
	error occurs in words 1,2, or 3 & no content words before error & at least 3 words after the error	What about Utt[error+1..end]	XXX are you going	What about "are you going"?
R_NE - *NE-realated* questions				
1	Entity=LOCATION, prepi=index of preceeding(IN, TO,AT)	Utt[1..i] where? (i=index of preceeding IN|TO|AT)	how long have people been selling drugs in XXX?	how long have people been selling drugs WHERE?
2	Entity=PERSON, dep TAG error=OBJ	Utt[1..error-1] whom?	I know your XXX	I know your WHOM?
3	Entity=PERSON, dep TAG error=not OBJ	Utt[1..error-1] who?	Hello mister XXX it's nice to meet you	Hello mister WHO?

Table 2. Question Generation Rules. Utt[a..b] indicates a subset of words from the utterance in index range from a to b

Utterance 2 contains the verb *win* after the error. However, a clarification question which includes the post-error context is incoherent, as shown in 2-A. A question generated by R_WH is more appropriate in this case, as in 2-B.

R_VB2 applies when an infinitival verb follows the error word. For a sentence like 3.

3 *We need to have XXX to use this medication.*
3-A *We need to have what?*
3-B *We need to have what to use this medication?*

R_WH would generate the question in 3-A. While this question is not incoherent, we claim that R_VB2 generates a more coherent clarification question with more information, as in 3-B. Our goal is to generate a clarification question which contains the most information from the original sentence while remaining coherent.

Rules R_VB3 and R_VB4 address the coherence of the question when the error word itself is a verb by appending *do what?* to the pre-error context. R_VB3 applies when an error word is an infinitival verb. R_VB4 applies when an error word is a verb preceded by a modal verb.

The rule R_NMOD applies when an error word is a noun modifier and the parent constituent of both is the NP. This rule generates a question by prepending *which* to the parent word of the error. For an utterance like 4, R_NMOD generates the question 4-A:

4 *Do you have anything other than these XXX plans?*
4-A *Which plans?*

R_NMOD generates short questions which directly target the error word. Questions generated using this rule contain minimal information from the sentence. They will be coherent unless the dependency parser fails or a parent word is also misrecognized and the error detection module fails to identify it as an error.

Rule R_START handles cases when an error occurs at the beginning of an utterance. In this case, there is no preceding context to include in asking a *what* question. This rule generates a *what about* question using post-error context Utt[error+1..end]. An example of this case is show in 5:

5 *XXX arrives tomorrow*
5-A *What about arrives tomorrow?*

The named entity based rule (R_NE) use the (hypothesized) named entity type of the error word to select a question word (*where, who, or whom*). Named entity rules are not evaluated in this experiment because there are only a small number of named entity errors in this data set.

The rules desdribed above are applied in the following order: R_START, R_NMOD, R_VB4, R_VB3, R_VB2, R_VB1, R_WH Table 3 shows an algorithm for applying the rules to each input sentence.

5 EVALUATION

We compared our automatically generated questions to the corpus of human-generated questions to assess the performance of our algorithm. First, we generated questions automatically using the algorithm described above for 84 randomly selected sentences with a single speech recognition error in a content word. These were se-

```
If error starts in words 1,2, or 3
  and there are no content words before error
  and there are at least 3 words after the error
    Apply R_START
else
If DEP TAG of error=NMOD and error's parent POS=NN|NNS
     Apply R_NMOD
else
If Entity of error = LOCATION
     Apply R_NE.1
else
If Entity of error = PERSON and DEP TAG of error = OBJ
     Apply R_NE.2
else
If Entity of error = PERSON and DEP TAG of error != OBJ
     Apply R_NE.3
else
if POS(error) = VB AND POS(error-1)=MD AND
     Apply R_VB4
else
if POS(error) = VB AND POS(error-1)=MD
     Apply R_VB3
else
if POS(error+1) = TO AND utt contains verb after error
     Apply R_VB2
else
if utt contains verb VB' after error AND PARENT(VB') = PARENT(error)
     Apply R_VB1
else
    Apply R_WH
```

Table 3. Question generation algorithm

lected from the same set of sentences we collected human clarification questions for using the Mechanical Turk data collection method described in Section 3. Next, we asked human raters to rank all the questions for syntactic and semantic quality on a Likert scale. In addition, we asked raters if they would prefer to ask a different clarification question for each question/utterance pair. Table 4 shows the questionnaire used for this evaluation.

rating	Question	Scale
Meaningful	The question is meaningful	1 - 5
Natural	The question is natural	1 - 5
Syntactic	The question is syntactically correct	1 - 5
Logical	The question clarifies the missing part (XXX) of the sentence.	1 - 5
Preference	In a dialogue, I prefer to hear **the above question** over being asked to **repeat the whole sentence**	1 - 5
AskDiff	Would you ask a clarification question differently?	yes/no

Table 4. Evaluation Criteria

We performed this experiment with two groups of subjects: general AMT workers[4] and also from native Standard American English (SAE) speakers recruited in our lab from students with no knowledge of the experiment. In the rest of the paper we refer to these groups as *Mturk* and *Recruited* subject groups. Interestingly, we find that the subjects' ratings in some of the aspects differed between the two subject groups. We compared the subjective ratings of the subjects for computer-generated questions and human-generated upper baseline. We also analyzed the difference between *Mturk* and *Recruited* annotators.

For the *Recruited* subject group, six native speaker of SAE rated the corpus of human- and computer-generated questions. Each question was annotated by three raters in each condition. Each rater annotated either a human or a computer-generated question for each generated sentence. The type of the question (human or computer) was unknown to the raters. Both labeler types used the same interface. Hence, for each question we obtained six scores: three from Mturk subjects and three from Recruited subjects working in the lab.

We used the annotation by the Lab subjects to compute inter-annotator agreement. We computed Kendall's W coefficient with correction for ties on the subset of the data annotated by 3 subjects. Table 5 shows that all of the measures have a moderate to substantial agreement.

rating	Kendall's W
Meaningful	.57
Natural	.59
Syntactic	.57
Logical	.69
Preference	.66
	fleiss-kappa
AskDiff	.43

Table 5. Inter-annotator agreement.

5.1 Compare Human and Computer Questions

For each sentence we generate one question using our rule set for question generation. Generic rule R_WH generated 58% of the questions in the test set, R_NMOD and all R_VB each generated 18% of questions and R_START generated 6% of questions, as shown in Table 6.

Rule ID	Count	Fraction
Total	84	100%
R_WH	49	58%
R_NMOD	15	18%
R_VB	15	18%
R_VB.1	2	
R_VB.2	5	
R_VB.3	7	
R_VB.4	1	
R_START	5	6%

Table 6. Number and percentage of times each rule was triggered

While overall the subjects rated human-generated questions significantly higher on the *Correct* and *Natural* categories, they rated computer-generated questions higher for the *Logical* category, as shown in Table 7. The difference between overall human- and computer-generated questions for *Meaningful, Preference* and *AskDiff* categories is not statistically significant. It is interesting to note that even human-generated questions do not receive perfect scores in any of the categories, averaging below 4 on the Likert scale for logical, meaningful, and preference over generic categories. Participants chose to ask a different question for 43% of human questions compared to 44% of computer questions. In the past, studies comparing human and computer-generated output also found that human-generated text failed to receive the highest possible scores [18].

Figure 2 graphically illustrates the difference between human-generated scores and the scores for each of the automatic rules. We

[4] Although we specified a requirement for the users to be native American English speakers, actual native language of the AMT subjects is not possible to verify.

type(number)	Correct	Logical	Meaningful	Natural	Pref	AskDiff
Human (84)	* 4.15 (.35)	3.59 (.78)	3.89 (.62)	* 4.02 (.46)	3.32 (.70)	0.44 (.28)
Computer (84)	3.72 (.54)	* 3.86 (.55)	3.99 (.49)	3.65 (.68)	3.37 (.61)	0.43 (.29)
Computer Generation Rules						
R_WH (49)	3.65	3.84	4.01	3.60	3.36	0.45
R_NMOD (15)	4.27	4.28	4.22	4.31	4.00	0.26
R_VB (15)	3.73	3.87	4.04	3.61	3.16	0.42
R_START (5)	2.73	2.80	3.03	2.37	2.13	0.87

Table 7. Overall means and standard deviations for human- and computer-generated questions. * indicates significantly higher score (or lower proportion of 'ask different') between human and computer means at a (p<.05) level.

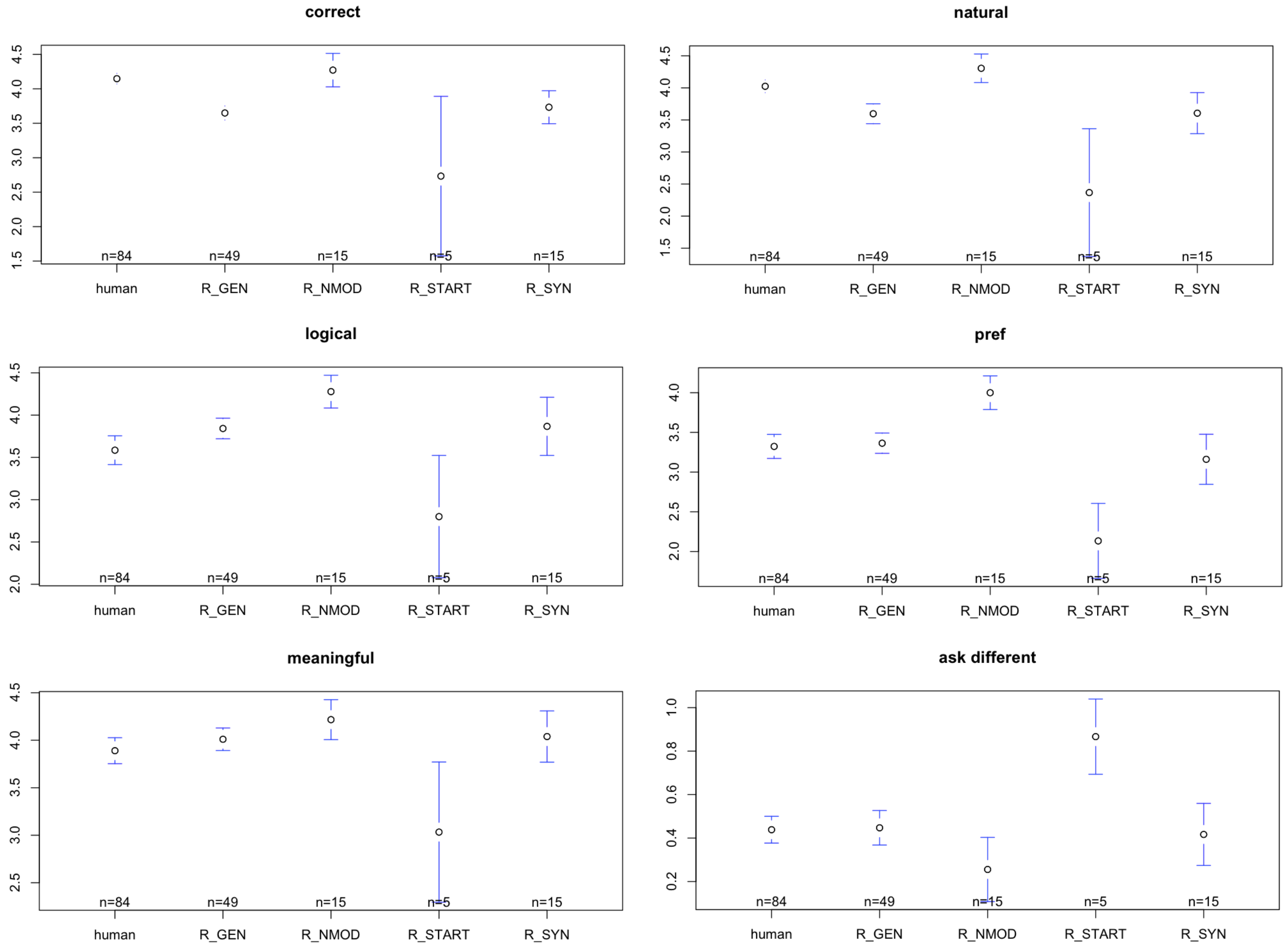

Figure 2. Scores for each question type with 5% confidence interval.

observe that the scores for computer-generated questions R_START are the lowest in all categories, showing that this rule generally does not work as well as others, although it did not trigger very often in our test set. Computer-generated questions with rules R_WH, R_NMOD, and R_VB outperformed human generation on the *meaningful* and *logical* categories. R_NMOD outperformed human generation on all categories.

5.2 Compare Recruited and Mturk Subjects

The subjects used for the evaluation were drawn from different populations. We analyzed differences in the scores each group assigned the generated sentences by these two groups. Table 8 show the mean values for the Recruited and Mturk subjects. Both subject pools rated human-generated questions significantly higher for correctness and naturalness. Both subject pools rated computer-generated questions more logical than human-generated questions but this difference is significant only for the Recruited subjects. Ratings for *Meaningful* and *Pref* categories did not differ significantly for either of the sub-

type	Correct	Logical	Meaningful	Natural	Pref	AskDiff
			Mturk			
computer	3.80	3.93	4.07	3.80	3.36	0.41
human	*4.12	3.82	4.00	*4.12	3.40	*0.31
			Recruited			
computer	3.64	*3.80	3.91	3.51	3.38	*0.45
human	*4.17	3.35	3.78	*3.93	3.24	0.57

Table 8. Mturk and students mean scores for human- and computer-generated questions. * indicates significantly higher score (or lower proportion for ask different) beetween human and computer means for a category(p<.05)

ject pools.

Overall, the subjects choose to specify a different question in 43% of cases for human- and in 44% of cases for computer-generated questions as we see in Table 7. However, mturk subjects prefer to ask a different question significantly more frequently for computer-generated questions (41%) than for human-generated questions (31%). Recruited subjects, on the other hand, prefer to ask a different question significantly more frequently for human-generated questions (57%) than for computer-generated questions (45%).

Tables 9 and 10 show examples where Mturk and Recruited subjects disagreed on the decision to ask a different clarification question. Table 9 shows examples of computer-generated questions for which three Recruited subjects chose to *ask a different question* but none of the Mturk subjects did. Mturk subjects in examples 1, 2, and 3 chose to shorten a question (*My what? Had to be what? Lift up what?*). In example 3, one of the alternative questions changes the pronoun orientation from *you* to *I*. Another subject chose to syntactically invert the sentence by moving *what* to the beginning: *What should I ask a person to lift up?*. In examples 1 and 4, the subjects chose to replace a question word from *what* to *who* and *where*.

Table 10 shows examples of human-generated questions for which at least two out of three Mturk subjects chose to *ask a different question* but none of the Recruited subjects did. The Recruited subjects chose to make the question longer and repeat more words from the original sentence in a question in examples 1 and 3. Recruited subjects were also more sensitive to matching an attribution discourse relation in a sentence. In Example 2, a sentence contains an attribution relation *I have heard that ...* but the human-generated question does not capture this attribution. Two of the three annotator modifications of this question inserted the attribution into the question, thus making it closer to the original sentence *What did you hear... What have you heard ...?*. In Example 4, the sentence does not contain an attribution but a human-generated question does. The question modification removes the attribution relation from the question. We observe that in several cases Recruited subjects change a human-generated questions to the question an automatic rules R_WH would have generated (e.g. *The set up is what by a professional?, The utility prices are what?*), suggesting that our rules perform well.

As in most natural language generated tasks, there is more than one correct way to generate a clarification question. When a subject chooses to specify a different question, it is not necessarily an indication that a question is bad but may be an indication of their personal preference to improve a question. By using subjects from different population pools, we capture different Natural Language Generation preferences for these subjects. The importance of modeling user preference has been shown by [8].

6 DISCUSSION

Our experiments have shown that a set of simple transformation rules can generate targeted clarification questions that human subjects rate with scores comparable to scores for human-generated clarification questions. What are the qualities of an optimal clarification question? We hypothesize that, besides syntactic correctness, conciseness and specificity of the question play a role in our ratings. We define conciseness simply as the length of a question. We define specificity as the number of concepts from the original sentence that a question mentions. By definition, more specific questions are less concise. Our automatic rules differed in conciseness and specificity of the questions that they generated.

The most concise of our rules, R_NMOD, out-performed all other rules. Recall that R_NMOD rule constructs a short question using a *which* + *head-word* construction. The success of this rule suggests that conciseness is a desirable property of a clarification question. Questions generated with the R_VB rule are less concise and more specific than R_WH because they contain additional post-error context. Despite being less concise, R_VB performed no worse than R_WH. Adding more specific information to a question does not necessarily lower the questions' ratings, so long as that information is appropriate.

Our findings suggest that an optimal clarification question is one that is concise yet specific enough to be coherent and meaningful. In the future, we would like to further investigate the trade-off between conciseness and specificity of clarification questions. We will examine which types of information are optional and which are essential when generating good clarification question.

Our findings reflect the fact that, in natural language generation, there is often more than one correct answer and that different users may have different preferences for the syntactic and lexical content of generated text. The most interesting of our findings is the finding that the scores for computer-generated questions for meaningful and logical categories are not lower than the score for human-generated questions. This indicates that even targeted questions that are not fully syntactically correct may still be perceived as logical or understandable to a user and thus can be interpreted and answered.

ACKNOWLEDGEMENTS

This material is based in part upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract # HR0011-12-C-0016. Any opinions, findings and conclusions or recommendations expressed are those of the authors and do not necessarily reflect the views of DARPA.

REFERENCES

[1] K. Acomb, J. Bloom, K. Dayanidhi, P. Hunter, P. Krogh, E. Levin, and R. Pieraccini, 'Technical support dialog systems: issues, problems, and

	sentence	human-gen question	recruited # ask diff	Mturk # ask diff	alternate questions by recruited subject
1	the set up is obviously XXX by a professional	what type of set up is this?	3/3	0/3	The set up is what by a professional? The set up is obviously what by a professional? it's obviously what?
2	i have heard that the utility prices are XXX	what are the utility prices?	3/3	0/3	What have you heard the utility prices are? What did you hear about the utility prices? The utility prices are what?
3	we should not have any XXX difficulties	what type of difficulties?	3/3	0/3	What type of difficulties should we not have? what difficulties?
4	have you taken any XXX	what do you think i took?	3/3	0/3	any what? Have I taken any what?

Table 9. Examples of human-generated questions preferred by the Mturk and dis-preferred by the Recruited subjects.

	sentence	computer-gen qustion	gen rule	Recruited # ask diff	Mturk # ask diff	alternate questions by Mturk subject
1	do your XXX have suspicious contacts	do your what have suspicious contacts?	R_VB.1	0/3	3/3	my what? What has suspicious contacts? Who?
2	how much of the building had to be XXX	how much of the building had to be what?	R_WH	0/3	2/3	had to be what?
3	you should ask a person to lift up their XXX	you should ask a person to lift up their what?	R_WH	0/3	3/3	I should ask a person to lift up their what? What should you ask a person to lift up? Lift up what?
4	deliver the supplies XXX	deliver the supplies what?	R_WH	0/3	2/3	Where? Deliver the supplies where?

Table 10. Examples of computer-generated questions preferred by Recruited and dispreferred by Mturk subjects.

solutions', in *Proceedings of the Workshop on Bridging the Gap: Academic and Industrial Research in Dialog Technologies*, p. 2531, (2007).

[2] M. Akbacak et al., 'Recent advances in SRI's IraqComm$^{\text{tm}}$ Iraqi Arabic-English speech-to-speech translation system', in *ICASSP*, pp. 4809–4812, (2009).

[3] J. Allwood, 'An activity based approach to pragmatics', in *Gothenburg Papers in Theoretical Linguistics*. John Benjamins, (1995).

[4] Amazon Mechanical Turk. http://aws.amazon.com/mturk/, accessed on 28 may, 2012, 2012.

[5] D. Bohus and E. Horvitz, 'Dialog in the open world: platform and applications', in *Proceedings of the 2009 international conference on Multimodal interfaces*, p. 3138, (2009).

[6] D. Bohus and A. I. Rudnicky, 'A principled approach for rejection threshold optimization in spoken dialog systems', in *INTERSPEECH*, pp. 2781–2784, (2005).

[7] H.H. Clark, *Using Language*, Cambridge University Press, 1996.

[8] G. Di Fabbrizio, A. J. Stent, and S. Bangalore, 'Referring expression generation using speaker-based attribute selection and trainable realization (ATTR)', in *Proceedings of the Fifth International Natural Language Generation Conference*, p. 211214, (2008).

[9] J. Ginzburg and R. Cooper, 'Clarification, ellipsism and the nature of contextual updates', *Linguistics and Philosophy*, **27**(3), (2004).

[10] J. Goldberg, M. Ostendorf, and K. Kirchhoff, 'The impact of response wording in error correction subdialogs', in *ISCA Tutorial and Research Workshop on Error Handling in Spoken Dialogue Systems*, (2003).

[11] S. Kalady, A. Elikkottil, and R. Das, 'Natural language question generation using syntax and keywords', in *Proceedings of QG2010: The Third Workshop on Question Generation*, p. 110, (2010).

[12] T. Koulouri and S. Lauria, 'Exploring miscommunication and collaborative behaviour in human-robot interaction', in *SIGDIAL Conference*, pp. 111–119, (2009).

[13] P. Lendvai, A. van den Bosch, E. Krahmer, and M. Swerts, 'Improving machine-learned detection of miscommunications in human-machine dialogues through informed data splitting', in *Proceedings of the ESSLLI Workshop on Machine Learning Approaches in Computational Linguistics*, p. 115, (2002).

[14] D. Litman, J. Hirschberg, and M. Swerts, 'Characterizing and predicting corrections in spoken dialogue systems', *Computational linguistics*, **32**(3), 417438, (2006).

[15] D. J. Litman and S. Silliman, 'Itspoke: an intelligent tutoring spoken dialogue system', in *Demonstration Papers at HLT-NAACL 2004*, HLT-NAACL–Demonstrations '04, pp. 5–8, Stroudsburg, PA, USA, (2004). Association for Computational Linguistics.

[16] A. Nasr, F. Béchet, J.F. Rey, B. Favre, and J. Le Roux, 'MACAON: an NLP tool suite for processing word lattices', in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Systems Demonstrations*, pp. 86–91. Association for Computational Linguistics, (2011).

[17] E. Pincus, S. Stoyanchev, and J. Hirschberg, 'Exploring features for localized detection of speech recognition errors', in *Proceedings of the SIGDIAL 2013 Conference*, (2013).

[18] P. Piwek and S. Stoyanchev, 'Data-oriented monologue-to-dialogue generation'.

[19] M. Purver, *The Theory and Use of Clarification Requests in Dialogue*, Ph.D. dissertation, King's College, University of London, 2004.

[20] A. Raux, B. Langner, A. Black, and M Eskenazi, 'Let's go public! taking a spoken dialog system to the real world', in *Proceedings of Eurospeech*, (2005).

[21] V. Rus, B. Wyse, P. Piwek, M. Lintean, S. Stoyanchev, and C. Moldovan, 'The first question generation shared task evaluation challenge', in *Proceedings of the 6th International Natural Language Generation Conference*, p. 251257, (2010).

[22] A. Liu S. Stoyanchev and J. Hirschberg, 'Clarification questions with feedback 2012.', in *Interdisciplinary Workshop on Feedback Behaviors in Dialog*, (2012).

[23] G. Skantze, 'Exploring human error handling strategies: Implications for spoken dialogue systems', in *ISCA Tutorial and Research Workshop on Error Handling in Spoken Dialogue Systems*, (2003).

[24] G. Skantze, 'Exploring human error recovery strategies: Implications for spoken dialogue systems', *Speech Communication*, **45**(2-3), 325–341, (2005).

[25] S. Stoyanchev, A. Liu, and J. Hirschberg, 'Modelling human clarification strategies', in *Proceedings of the SIGDIAL 2013 Conference*, (2013).

[26] S. Stoyanchev, P. Salletmayr, J. Yang, and J. Hirschberg, 'Localized detection of speech recognition errors.', in *SLT*, pp. 25–30. IEEE, (2012).

[27] B. A. Weiss et al., 'Performance evaluation of speech translation systems', in *LREC*, (2008).

[28] J. D. Williams and S. Young, 'Characterizing task-oriented dialog using a simulated ASR channel', in *Proceedings of the ICSLP, Jeju, South Korea*, (2004).