Modelling Human Clarification Strategies

Svetlana Stoyanchev, Alex Liu, Julia Hirschberg
Columbia University, New York NY 10027
sstoyanchev, al3037, julia @cs.columbia.edu

Abstract

We model human responses to speech recognition errors from a corpus of human clarification strategies. We employ learning techniques to study 1) the decision to either stop and ask a clarification question or to continue the dialogue without clarification, and 2) the decision to ask a targeted clarification question or a more generic question. Targeted clarification questions focus specifically on the part of an utterance that is misrecognized, in contrast with generic requests to 'please repeat' or 'please rephrase'. Our goal is to generate targeted clarification strategies for handling errors in spoken dialogue systems, when appropriate. Our experiments show that linguistic features, in particular the inferred part-of-speech of a misrecognized word are predictive of human clarification decisions. A combination of linguistic features predicts a user's decision to continue or stop a dialogue with accuracy of 72.8% over a majority baseline accuracy of 59.1%. The same set of features predict the decision to ask a targeted question with accuracy of 74.6% compared with the majority baseline of 71.8%.[1]

1 Introduction

Clarification questions are common in human-human dialogue. They help dialogue participants maintain dialogue flow and resolve misunderstandings. Purver (2004) finds that in human-human dialogue speakers most frequently use *reprise* clarification questions to resolve recognition errors. Reprise clarification questions use portions of the misunderstood utterance which are thought to be correctly recognized to *target* the part of an utterance that was misheard or misunderstood. In the following example from (Purver, 2004), Speaker B has failed to hear the word *toast* and so constructs a clarification question using a portion of the correctly understood utterance — the word *some* — to query the portion of the utterance B has failed to understand:

A: Can I have **some** *toast* please?
B: Some?
A: Toast.

Unlike human conversational partners, most dialogue systems today employ generic 'please repeat/rephrase' questions asking a speaker to repeat or rephrase an entire utterance. Our goal is to introduce reprise, or targeted, clarifications into an automatic spoken system. Targeted clarifications can be especially useful for systems accepting unrestricted speech, such as tutoring systems, intelligent agents, and speech translation systems. Using a reprise question, a user can correct an error by repeating only a portion of an utterance. Targeted questions also provide natural grounding and implicit confirmation by signalling to the conversation partner which parts of an utterance have been recognized.

In order to handle a misrecognition, the system must first identify misrecognized words (Stoyanchev et al., 2012), determine the type of question to ask, and construct the question. In this work, we address two points necessary for determining the type of question to ask:

- Is it appropriate for a system to ask a clarification question when a misrecognized word is detected?
- Is it possible to ask a targeted clarification question for a given sentence and an error segment?

To answer these questions, we analyze a corpus of human responses to transcribed utterances with missing information which we collected using Amazon Mechanical Turk (2012). Although the data collection was text-based, we asked annotators to respond as they would in a dialogue. In Section 2, we describe related work on error recovery strategies in dialogue systems. In Section 3, we describe the corpus used in this experiment. In Section 4, we describe our experiments on human clarification strategy modelling. We conclude in Section 5 with our plan for applying our models in spoken systems.

2 Related work

To handle errors in speech recognition, slot-filling dialogue systems typically use simple rejection ("I'm sorry. I didn't understand you.") when they have low confidence in a recognition hypothesis and explicit or implicit confirmation when confidence scores

[1]This work was partially funded by DARPA HR0011-12-C-0016 as a Columbia University subcontract to SRI International.

Proceedings of the SIGDIAL 2013 Conference, pages 137–141,
Metz, France, 22-24 August 2013.

are higher. Machine learning approaches have been successfully employed to determine dialogue strategies (Bohus and Rudnicky, 2005; Bohus et al., 2006; Rieser and Lemon, 2006), such as when to provide help, repeat a previous prompt, or move on to the next prompt. Reiser and Lemon (2006) use machine learning to determine an optimal clarification strategy in multimodal dialogue. Komatani et al. (2006) propose a method to generate a help message based on perceived user expertise. Corpus studies on human clarifications in dialogue indicate that users ask task-related questions and provide feedback confirming their hypothesis instead of giving direct indication of their misunderstanding (Skantze, 2005; Williams and Young, 2004; Koulouri and Lauria, 2009). In our work, we model human strategies with the goal of building a dialogue system which can generate targeted clarification questions for recognition errors that require additional user input but which can also recover from other errors automatically, as humans do.

3 Data

In our experiments, we use a dataset of human responses to missing information, which we collected with Amazon Mechanical Turk (AMT). Each AMT annotator was given a set of Automatic Speech Recognition (ASR) transcriptions of an English utterance with a single misrecognized segment. 925 such utterances were taken from acted dialogues between English and Arabic speakers conversing through SRI's *IraqComm* speech-to-speech translation system (Akbacak et al., 2009). Misrecognized segments were replaced by *"XXX"* to indicate the missing information, simulating a dialogue system's automatic detection of misrecognized words (Stoyanchev et al., 2012). For each sentence, AMT workers were asked to 1) indicate whether other information in the sentence made its meaning clear despite the error, 2) guess the missing word if possible, 3) guess the missing word's part-of-speech (POS) if possible, and 4) create a targeted clarification question if possible. Three annotators annotated each sentence. Table 1 summarizes the results. In 668 (72%) of the sentences an error segment corresponds to a single word while in 276 (28%) of them, an error segment corresponds to multiple words. For multiple word error segments, subjects had the option of guessing multiple words and POS tags. We scored their guess correct if any of their guesses matched the syntactic head word of an error segment determined from an automatically assigned dependency parse structure.

We manually corrected annotators' POS tags if the hypothesized word was itself correct. After this post-processing, we see that AMT workers hypothesized POS correctly in 57.7% of single-word and 60.2% of multi-word error cases. They guessed words correctly in 34.9% and 19.3% of single- and multi-word error cases. They choose to ask a clarification question in 38.3% /47.9% of cases and 76.1%/62.3% of these questions were targeted clarification questions. These results indicate that people are often able to guess a POS tag and sometimes an actual word. We observe that 1) in a single-word error segment, subjects are better at guessing an actual word than they are in a multi-word error segment; and 2) in a multi-word error segment, subjects are more likely to ask a clarification question and less likely to ask a targeted question. All three annotators agree on POS tags in 62% of cases and on hypothesized words in 25%. Annotators' agreement on response type is low — not surprising since there is more than one appropriate and natural way to respond in dialogue. In 39% of cases, all three annotators agree on the decision to stop/continue and in only 25% of cases all three annotators agree on asking a targeted clarification question. Figure 1 shows the annotator

	Single-word error	**Agree**	**Multi-word error**
Total sent	668 (72%)	-	276 (28%)
Correct POS	57.7%	62%	60.2%
Correct word	34.9%	25%	19.3%
Ask a question	38.3%	39%	47.9%
Targeted question	76.1%	25%	62.3%

Table 1: *Annotation summary for single-word and multi-word error cases. Absolute annotator agreement is shown for single-word error cases.*

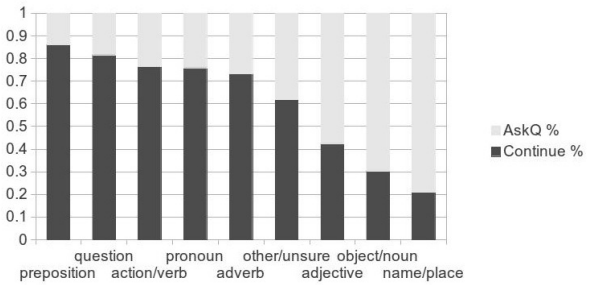

Figure 1: *Distribution of decisions to ask a question or continue dialogue without a question.*

distribution for asking a clarification question vs. continuing the dialogue based on hypothesized POS tag. It indicates that annotators are more likely to ask a question than continue without a question when they hypothesize a missing word to be a content word (noun or adjective) or when they are unsure of the POS of the missing word. They are more likely to continue when they believe a missing word is a function word. However, when they believe a missing word is a verb, they are more likely to continue, and they are also likely to identify the missing verb correctly.

Figure 2 shows a distribution of annotator decisions as to the type of question they would ask. The proportion of *targeted* question types varies with hypothesized POS. It is more prevalent than *confirm* and *generic* questions combined for all POS tags except preposition and question word, indicating that annotators are generally able to construct a targeted clarification question based on their analysis of the error segment.

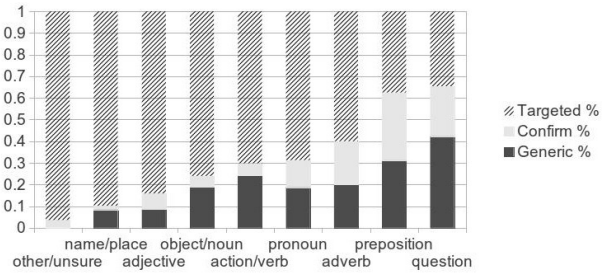

Figure 2: *Distribution of decisions for targeted, confirmation, and generic question types.*

4 Experiment

We use our AMT annotations to build classifiers for 1) choice of action: *stop* and engage in clarification vs. *continue* dialogue; and 2) type of clarification question (*targeted* vs. *non-targeted*) to ask. For the *continue/stop* experiment, we aim to determine whether a system should stop and ask a clarification question. For the *targeted vs. non-targeted* experiment, we aim to determine whether it is possible to ask a targeted clarification question.[2]

Using the Weka (Witten and Eibe, 2005) machine learning framework, we build classifiers to predict AMT decisions. We automatically assign POS tags to transcripts using the Stanford tagger (Toutanova and others, 2003). We compare models built with an automatically tagged POS for an error word (*POS-auto*) with one built with POS guessed by a user (*POS-guess*). Although a dialogue manager may not have access to a correct POS, it may simulate this by predicting POS of the error. We assign dependency tags using the AMU dependency parser (Nasr et al., 2011) which has been optimized on the Transtac dataset.

We hypothesize that a user's dialogue move depends on the syntactic structure of a sentence as well as on syntactic and semantic information about the error word and its syntactic parent. To capture sentence structure, we use features associated with the whole sentence: POS ngram, all pairs of parent-child dependency tags in a sentence (*Dep-pair*), and all semantic roles (*Sem-presence*) in a sentence. To capture the syntactic and semantic role of a misrecognized word, we use features associated with this word: POS tag, dependency tag (*Dep-tag*), POS of the parent word (*Parent-POS*), and semantic role of an error word (*Sem-role*).

We first model individual annotators' decisions for each of the three annotation instances. We measure the value that each feature adds to a model, using annotators' POS guess (*POS-guess*). Next, we model a joint annotators' decision using the automatically assigned *POS-auto* feature. This model simulates a system behaviour in a dialogue with a user where a system chooses a single dialogue move for each situation. We run 10-fold cross validation using the Weka J48 Decision Tree algorithm.

[2]If any annotators asked a targeted question, we assign a positive label to this instance, and negative otherwise.

Feature	**Description**
Count	
Word-position	*beginning* if a misrecognized word is the first word in the sentence, *end* if it is the last word, *middle* otherwise.
Utterance-length	number of words in the sentence
Part-of-speech (compare)	
POS-auto	POS tag of the misrecognized word automatically assigned on a transcript
POS-guess	POS tag of the misrecognized word guessed by a user
POS ngrams	
POS ngrams	all bigrams and trigrams of POS tags in a sentence
Syntactic Dependency	
Dep-tag	dependency tag of the misrecognized word automatically assigned on a transcript
Dep-pair	dependency tags of all (parent, child) pairs in the sentence
Parent-POS	POS tag of the syntactic parent of the misrecognized word
Semantic	
Sem-role	semantic role of the misrecognized word
Sem-presence	all semantic roles present in a sentence

Table 2: Features

4.1 Stop/Continue Experiment

In this experiment, we classify each instance in the dataset into a binary *continue* or *stop* decision. Since each instance is annotated by three annotators, we first predict individual annotators' decisions. The absolute agreement on continue/stop is 39% which means that 61% of sentences are classified into both classes. We explore the role of each feature in predicting these decisions. All features used in this experiment, except for the *POS-guess* feature, are extracted from the sentences automatically. Variation in the *POS-guess* feature may explain some of the difference between annotator decisions.

Features	Acc	F-measure	%Diff
Majority baseline	59.1%		
All features	**72.8%** †	**0.726**	0.0%
less utt length	72.9% †	0.727	+0.1%
less POS ngrams	72.8% †	0.727	+0.1%
less Semantic	72.6% †	0.724	-0.3%
less Syn. Depend.	71.5% †	0.712	-1.9%
less Position	71.2% †	0.711	-2.0%
less POS	67.9% †	0.677	-6.7%
POS only	70.1% †	0.690	-5.0%

Table 3: Stop/Continue experiment predicting individual annotator's decision with *POS-guess*. Accuracy, F-measure and Difference of f-measure from *All feature*. †indicates statistically significant difference from the majority baseline ($p<.01$)

Table 3 shows the results of continue/stop classification. A majority baseline method predicts the most frequent class *continue* and has 59.1% accuracy. In comparison, our classifier, built with all features, achieves 72.8% accuracy.

Next, we evaluate the utility of each feature by removing it from the feature set and comparing the model built without it with a model built on all features. POS is the most useful feature, as we expected: when it is removed from the feature set, the f-measure decreases by 6.7%. A model trained on the *POS-guess* feature alone outperforms a model trained on all other features. Word *position* in the sentence is the next most salient feature, contributing 2% to the f-measure. The syntactic dependency features *Syn-Dep, Dep-pair,* and *Parent POS* together contribute 1.9%.[3]

Next, we predict a majority decision for each sentence. Table 4 shows the accuracy of this prediction. A majority baseline has an accuracy of 59.9%. When we use a model trained on the *POS-auto* feature alone, accuracy rises to 66.1%, while a combination of all features further increases it to 69.2%.

Features	Acc	F-measure
Majority baseline	59.9%	
POS	66.1% †	0.655
All features	69.2% †	0.687

Table 4: Stop/Continue experiment predicting majority decision, using *POS-auto*. †indicates statistically significant difference from the majority baseline ($p<.01$).

4.2 Targeted Clarification Experiment

In this experiment, we classify each instance into *targeted* or *not targeted* categories. The *targeted* category comprises the cases in which an annotator chooses to stop and ask a targeted question. We are interested in identifying these cases in order to determine whether a system should try to ask a targeted clarification question. Table 5 shows the results of this experiment. The majority baseline predicts *not targeted* and has a 71.8% accuracy because in most cases, no question is asked. A model trained on all features increases accuracy to 74.6%. POS is the most salient feature, contributing 3.8% to the f-measure. All models that use POS feature are significantly different from the baseline. The next most salient features are POS ngram and a combination of syntactic dependency features contributing 1% and .5% to the f-measure respectively.

Table 6 shows system performance in predicting a joint annotators' decision of whether a targeted question can be asked. A joint decision in this experiment is considered *not targeted* when none of the annotators chooses to ask a targeted question. We aim at identifying the cases where position of an error word makes it difficult to ask a clarification question, such as for a sentence *XXX somebody steal these supplies.* Using the automatically assigned POS (*POS-auto*) feature alone achieves an accuracy of 62.2%, which is almost 10% above the baseline. A combination of all features, surprisingly, lowers the accuracy to 59.4%. Interestingly, a combination of all features *less POS* increases accuracy above the baseline by 7.6% points to 60.1% accuracy.

[3] All trained models are significantly different from the baseline. None of the trained models are significantly different from each other.

Features	Acc	F-measure	%Diff
Majority baseline	71.8%		
All features	74.6% †	**0.734**	0.0%
All feature (POS guess)			
less Utt length	74.8% †	0.736	+0.3%
less Position	74.9% †	0.731	-0.4%
less Semantic	74.8% †	0.737	+0.4%
less Syn. Depend.	74.2% †	0.730	-0.5%
less POS ngram	74.2% †	0.727	-1.0%
less POS	74.0%	0.706	-3.8%
POS	74.1% †	0.731	-0.4%

Table 5: Targeted/not experiment predicting individual annotator's decision with *POS-guess*. Accuracy, F-measure and Difference of f-measure from *All feature*. †indicates statistically significant difference from the majority baseline ($p<.05$)

Features	Acc	F-measure
Majority baseline	52.5%	
POS only	62.2% †	0.622
All features	59.4% †	0.594
All features *less POS*	60.1% †	0.600

Table 6: Targeted/not experiment predicting majority decision, using POS tag feature *POS-auto*. †indicates statistically significant difference from the majority baseline.

5 Conclusions and Future Work

In this paper we have described experiments modelling human strategies in response to ASR errors. We have used machine learning techniques on a corpus annotated by AMT workers asked to respond to missing information in an utterance. Although annotation agreement in this task is low, we aim to learn natural strategies for a dialogue system by combining the judgements of several annotators. In a dialogue, as in other natural language tasks, there is more than one appropriate response in each situation. A user does not judge the system (or another speaker) by a single response. Over a dialogue session, appropriateness, or lack of it in system actions, becomes evident. We have shown that by using linguistic features we can predict the decision to either ask a clarification question or continue dialogue with an accuracy of 72.8% in comparison with the 59.1% baseline. The same linguistic features predict a targeted clarification question with an accuracy of 74.6% compared to the baseline of 71.8%.

In future work, we will apply modelling of a clarification choice strategy in a speech-to-speech translation task. In our related work, we have addressed the problem of automatic correction of some ASR errors for cases when humans believe a dialogue can continue without clarification In other work, we have addressed the creation of targeted clarification questions for handling the cases when such questions are appropriate. Combining these research directions, we are developing a clarification component for a speech-to-speech translation system that responds naturally to speech recognition errors.

References

M. Akbacak, Franco, H., M. Frandsen, S. Hasan, H. Jameel, A. Kathol, S. Khadivi, X. Lei, A. Mandal, S. Mansour, K. Precoda, C. Richey, D. Vergyri, W. Wang, M. Yang, and J. Zheng. 2009. Recent advances in SRI's IraqComm^{tm} Iraqi Arabic-English speech-to-speech translation system. In *ICASSP*, pages 4809–4812.

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

D. Bohus and A. I. Rudnicky. 2005. A principled approach for rejection threshold optimization in spoken dialog systems. In *INTERSPEECH*, pages 2781–2784.

D. Bohus, B. Langner, A. Raux, A. Black, M. Eskenazi, and A. Rudnicky. 2006. Online supervised learning of non-understanding recovery policies. In *Proceedings of SLT*.

Y. Fukubayashi, K. Komatani, T. Ogata, and H. Okuno. 2006. Dynamic help generation by estimating user's mental model in spoken dialogue systems. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*.

T. Koulouri and S. Lauria. 2009. Exploring miscommunication and collaborative behaviour in human-robot interaction. In *SIGDIAL Conference*, pages 111–119.

A. Nasr, F. Béchet, J.F. Rey, B. Favre, and J. Le Roux. 2011. 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*, pages 86–91. Association for Computational Linguistics.

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

V. Rieser and O. Lemon. 2006. Using machine learning to explore human multimodal clarification strategies. In *ACL*.

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

Svetlana Stoyanchev, Philipp Salletmayr, Jingbo Yang, and Julia Hirschberg. 2012. Localized detection of speech recognition errors. In *SLT*, pages 25–30. IEEE.

K. Toutanova et al. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*. Association for Computational Linguistics.

J. D. Williams and S. Young. 2004. Characterizing task-oriented dialog using a simulated ASR channel. In *Proceedings of the ICSLP, Jeju, South Korea*.

I. Witten and F. Eibe. 2005. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 2nd edition.