Can I finish? Learning when to respond to incremental interpretation results in interactive dialogue

David DeVault and **Kenji Sagae** and **David Traum**
USC Institute for Creative Technologies
13274 Fiji Way
Marina del Rey, CA 90292
{devault,sagae,traum}@ict.usc.edu

Abstract

We investigate novel approaches to responsive overlap behaviors in dialogue systems, opening possibilities for systems to interrupt, acknowledge or complete a user's utterance while it is still in progress. Our specific contributions are a method for determining when a system has reached a point of maximal understanding of an ongoing user utterance, and a prototype implementation that shows how systems can use this ability to strategically initiate system completions of user utterances. More broadly, this framework facilitates the implementation of a range of overlap behaviors that are common in human dialogue, but have been largely absent in dialogue systems.

1 Introduction

Human spoken dialogue is highly interactive, including feedback on the speech of others while the speech is progressing (so-called "backchannels" (Yngve, 1970)), monitoring of addressees and other listener feedback (Nakano et al., 2003), fluent turn-taking with little or no delays (Sacks et al., 1974), and overlaps of various sorts, including collaborative completions, repetitions and other grounding moves, and interruptions. Interruptions can be either to advance the new speaker's goals (which may not be related to interpreting the other's speech) or in order to prevent the speaker from finishing, which again can be for various reasons. Few of these behaviors can be replicated by current spoken dialogue systems. Most of these behaviors require first an ability to perform incremental interpretation, and second, an ability to predict the final meaning of the utterance.

Incremental interpretation enables more rapid response, since most of the utterance can be interpreted before utterance completion (Skantze and Schlangen, 2009). It also enables giving early feedback (e.g., head nods and shakes, facial expressions, gaze shifts, and verbal backchannels) to signal how well things are being perceived, understood, and evaluated (Allwood et al., 1992).

For some responsive behaviors, one must go beyond incremental interpretation and predict some aspects of the full utterance before it has been completed. For behaviors such as complying with the evocative function (Allwood, 1995) or intended perlocutionary effect (Sadek, 1991), grounding by demonstrating (Clark and Schaefer, 1987), or interrupting to avoid having the utterance be completed, one must predict the semantic content of the full utterance from a partial prefix fragment. For other behaviors, such as timing a reply to have little or no gap, grounding by saying the same thing at the same time (called "chanting" by Hansen et al. (1996)), performing collaborative completions (Clark and Wilkes-Gibbs, 1986), or some corrections, it is important not only to predict the meaning, but also the form of the remaining part of the utterance.

We have begun to explore these issues in the context of the dialogue behavior of virtual human (Rickel and Johnson, 1999) or embodied conversational agent (Cassell et al., 2000) characters for multiparty negotiation role-playing (Traum et al., 2008b). In these kinds of systems, human-like behavior is a goal, since the purpose is to allow a user to practice this kind of dialogue with the virtual humans in training for real negotiation dialogues. The more realistic the characters' dialogue behavior is, the more kinds of negotiation situations can be adequately trained for. We discuss these sys-

Proceedings of SIGDIAL 2009: the 10th Annual Meeting of the Special Interest Group in Discourse and Dialogue, pages 11–20,
Queen Mary University of London, September 2009.

tems further in Section 2.

In Sagae et al. (2009), we presented our first results at prediction of semantic content from partial speech recognition hypotheses, looking at length of the speech hypothesis as a general indicator of semantic accuracy in understanding. We summarize this previous work in Section 3.

In the current paper, we incorporate additional features of real-time incremental interpretation to develop a more nuanced prediction model that can accurately identify moments of maximal understanding within individual spoken utterances (Section 4). We demonstrate the value of this new ability using a prototype implementation that collaboratively completes user utterances when the system becomes confident about how the utterance will end (Section 5). We believe such predictive models will be more broadly useful in implementing responsive overlap behaviors such as rapid grounding using completions, confirmation requests, or paraphrasing, as well as other kinds of interruptions and multi-modal displays. We conclude and discuss future work in Section 6.

Figure 1: SASO-EN negotiation in the cafe: Dr. Perez (left) looking at Elder al-Hassan.

$$\begin{bmatrix} mood : declarative \\ sem : \begin{bmatrix} type : event \\ agent : captain - kirk \\ event : deliver \\ theme : power - generator \\ modal : [\, possibility : can \,] \\ speech - act : [\, type : offer \,] \end{bmatrix} \end{bmatrix}$$

Figure 2: AVM utterance representation.

2 Domain setting

The case study we present in this paper is taken from the SASO-EN scenario (Hartholt et al., 2008; Traum et al., 2008b). This scenario is designed to allow a trainee to practice multi-party negotiation skills by engaging in face to face negotiation with virtual humans. The scenario involves a negotiation about the possible re-location of a medical clinic in an Iraqi village. A human trainee plays the role of a US Army captain, and there are two virtual humans that he negotiates with: Doctor Perez, the head of the NGO clinic, and a local village elder, al-Hassan. The doctor's main objective is to treat patients. The elder's main objective is to support his village. The captain's main objective is to move the clinic out of the marketplace, ideally to the US base. Figure 1 shows the doctor and elder in the midst of a negotiation, from the perspective of the trainee. Figure A-1 in the appendix shows a sample dialogue from this domain.

The system has a fairly typical set of processing components for virtual humans or dialogue systems, including ASR (mapping speech to words), NLU (mapping from words to semantic frames), dialogue interpretation and management (handling context, dialogue acts, reference and deciding what content to express), NLG (mapping frames to words), non-verbal generation, and synthesis and realization. The doctor and elder use the same ASR and NLU components, but have different modules for the other processing, including different models of context and goals, and different output generators. In this paper, we will often refer to the characters with various terms, including "virtual humans", "agents", or "the system".

In this paper, we are focusing on the NLU component, looking at incremental interpretation based on partial speech recognition results, and the potential for using this information to change the dialogue strategy where warranted, and provide responses before waiting for the final speech result. The NLU output representation is an attribute-value matrix (AVM), where the attributes and values represent semantic information that is linked to a domain-specific ontology and task model (Hartholt et al., 2008). Figure 2 shows an example representation, for an utterance such as "we can provide you with power generators". The AVMs are linearized, using a path-value notation, as shown in Figure 3.

To develop and test the new incremental/prediction models, we are using a corpus of

```
<s>.mood declarative
<s>.sem.type event
<s>.sem.agent captain-kirk
<s>.sem.event deliver
<s>.sem.theme power-generator
<s>.sem.modal.possibility can
<s>.sem.speechact.type offer
```

Figure 3: Example NLU frame.

utterances collected from people playing the role of captain and negotiating with the virtual doctor and elder. In contrast with Figure A-1, which is a dialogue with one of the system designers who knows the domain well, dialogues with naive users are generally longer, and often have a fairly high word error rate (average 0.54), with many out of domain utterances. The system is robust to these kinds of problems, both in terms of the NLU approach (Leuski and Traum, 2008; Sagae et al., 2009) as well as the dialogue strategies (Traum et al., 2008a). This is accomplished in part by approximating the meaning of utterances. For example, the frame in Figure 3 is also returned for an utterance of *we are prepared to give you guys generators for electricity downtown* as well as the ASR output for this utterance, *we up apparently give you guys generators for a letter city don town*.

3 Predicting interpretations from partial recognition hypotheses

Our NLU module, mxNLU (Sagae et al., 2009), is based on maximum entropy classification (Berger et al., 1996), where we treat entire individual frames as classes, and extract input features from ASR. The training data for mxNLU is a corpus of approximately 3,500 utterances, each annotated with the appropriate frame. These utterances were collected from user sessions with the system, and the corresponding frames were assigned manually. Out-of-domain utterances (about 15% of all utterances in our corpus) could not be mapped to concepts in our ontology and task model, and were assigned a "garbage" frame. For each utterance in our corpus, we have both a manual transcription and the output of ASR, although only ASR is used by mxNLU (both at training and at runtime). Each training instance for mxNLU consists of a frame, paired with a set of features that represent the ASR output for user utterances. The specific features used by the classifier are: each word in the input string (bag-of-words representation of the input), each bigram (pairs of consecutive words), each pair of any two words in the input, and the number of words in the input string.

In the 3,500-utterance training set, there are 136 unique frames (135 that correspond to the semantics of different utterances in the domain, plus one frame for out-of-domain utterances).[1] The NLU task is then framed as a multiclass classification approach with 136 classes, and about 3,500 training examples.

Although mxNLU produces entire frames as output, we evaluate NLU performance by looking at precision and recall of the attribute-value pairs (or *frame elements*) that compose frames. Precision represents the portion of frame elements produced by mxNLU that were correct, and recall represents the portion of frame elements in the gold-standard annotations that were proposed by mxNLU. By using precision and recall of frame elements, we take into account that certain frames are more similar than others and also allow more meaningful comparative evaluation with NLU modules that construct a frame from subelements or for cases when the actual frame is not in the training set. The precision and recall of frame elements produced by mxNLU using complete ASR output are 0.78 and 0.74, respectively, for an F-score (harmonic mean of precision and recall) of 0.76.

3.1 NLU with partial ASR results

The simplest way to perform NLU of partial ASR results is simply to process the partial utterances using the NLU module trained on complete ASR output. However, better results may be obtained by training separate NLU models for analysis of partial utterances of different lengths. To train these separate NLU models, we first ran the audio of the utterances in the training data through our ASR module, recording all partial results for each utterance. Then, to train a model to analyze partial utterances containing N words, we used only partial utterances in the training set containing N words (unless the entire utterance contained less than N words, in which case we simply used the complete utterance). In some cases, multiple partial ASR results for a single utterance

[1] In a separate development set of 350 utterances, annotated in the same way as the training set, we found no frames that had not appeared in the training set.

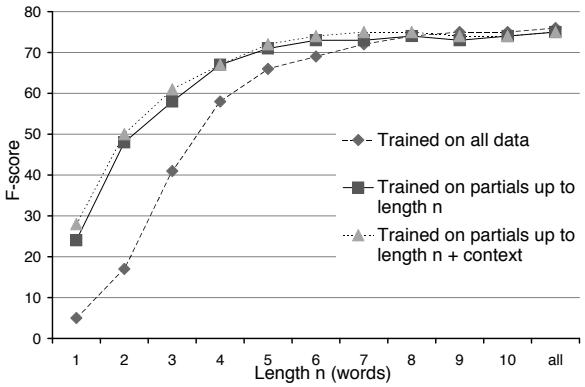

Figure 4: F-score for three NLU models on partial ASR results up to N words.

contained the same number of words, and we used the last partial result with the appropriate number of words.[2] We trained ten separate partial NLU models for N varying from one to ten.

Figure 4 shows the F-score for frames obtained by processing partial ASR results up to length N using three variants of mxNLU. The dashed line is our baseline NLU model, trained on complete utterances only, and the solid line shows the results obtained with length-specific NLU models. The dotted line shows results for length-specific models that also use features that capture aspects of dialogue context. In these experiments, we used unigram and bigram word features extracted from the most recent system utterance to represent context, but found that these context features did not improve NLU performance. Our final NLU approach for partial ASR hypotheses is then to train separate models for specific lengths, using hypotheses of that length during training (solid line in figure 4).

4 How well is the system understanding?

In this section, we present a strategy that uses machine learning to more closely characterize the performance of a maximum entropy based incremental NLU module, such as the mxNLU module described in Section 3. Our aim is to identify strategic points in time, as a specific utterance is occurring, when the system might react with confidence that the interpretation will not signif-

[2] At run-time, this can be closely approximated by taking the partial utterance immediately preceding the first partial utterance of length $N + 1$.

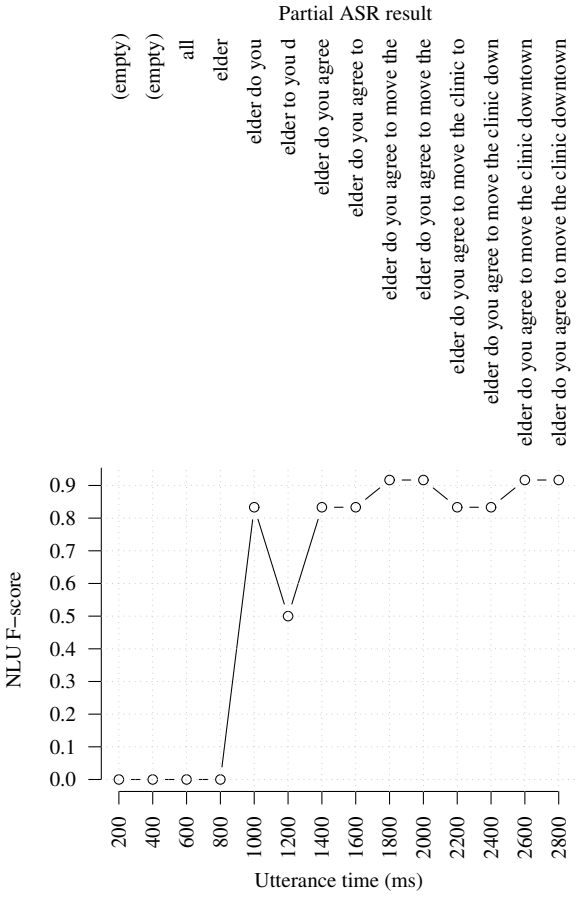

Figure 5: Incremental interpretation of a user utterance.

icantly improve during the rest of the utterance. This reaction could take several forms, including providing feedback, or, as described in Section 5 an agent might use this information to opportunistically choose to initiate a completion of a user's utterance.

4.1 Motivating example

Figure 5 illustrates the incremental output of mxNLU as a user asks, *elder do you agree to move the clinic downtown?* Our ASR processes captured audio in 200ms chunks. The figure shows the partial ASR results after the ASR has processed each 200ms of audio, along with the F-

score achieved by mxNLU on each of these partials. Note that the NLU F-score fluctuates somewhat as the ASR revises its incremental hypotheses about the user utterance, but generally increases over time.

For the purpose of initiating an overlapping response to a user utterance such as this one, the agent needs to be able (in the right circumstances) to make an assessment that it has already understood the utterance "well enough", based on the partial ASR results that are currently available. We have implemented a specific approach to this assessment which views an utterance as understood "well enough" if the agent would not understand the utterance any better than it currently does even if it were to wait for the user to finish their utterance (and for the ASR to finish interpreting the complete utterance).

Concretely, Figure 5 shows that after the entire 2800ms utterance has been processed by the ASR, mxNLU achieves an F-score of 0.91. However, in fact, mxNLU already achieves this maximal F-score at the moment it interprets the partial ASR result *elder do you agree to move the* at 1800ms. The agent therefore could, in principle, initiate an overlapping response at 1800ms without sacrificing any accuracy in its understanding of the user's utterance.

Of course the agent does not automatically realize that it has achieved a maximal F-score at 1800ms. To enable the agent to make this assessment, we have trained a classifier, which we call MAXF, that can be invoked for any specific partial ASR result, and which uses various features of the ASR result and the current mxNLU output to estimate whether the NLU F-score for the current partial ASR result is at least as high as the mxNLU F-score would be if the agent were to wait for the entire utterance.

4.2 Machine learning setup

To facilitate the construction of our MAXF classifier, we identified a range of potentially useful features that the agent could use at run-time to assess its confidence in mxNLU's output for a given partial ASR result. These features are exemplified in the appendix in Figure A-2, and include: K, the number of partial results that have been received from the ASR; N, the length (in words) of the current partial ASR result; Entropy, the entropy in the probability distribution mxNLU assigns to alternative output frames (lower entropy corresponds to a more focused distribution); $P_{\max}$, the probability mxNLU assigns to the most probable output frame; NLU, the most probable output frame (represented for convenience as fI, where I is an integer index corresponding to a specific complete frame). We also define MAXF (GOLD), a boolean value giving the ground truth about whether mxNLU's F-score for this partial is at least as high as mxNLU's F-score for the final partial for the same utterance. In the example, note that MAXF (GOLD) is true for each partial where mxNLU's F-score ($F(K)$) is $\geq$ 0.91, the value achieved for the final partial (*elder do you agree to move the clinic downtown*). Of course, the actual F-score $F(K)$ is not available at run-time, and so cannot serve as an input feature for the classifier.

Our general aim, then, is to train a classifier, MAXF, whose output predicts the value of MAXF (GOLD) as a function of the input features. To create a data set for training and evaluating this classifier, we observed and recorded the values of these features for the 6068 partial ASR results in a corpus of ASR output for 449 actual user utterances.[3]

We chose to train a decision tree using Weka's J48 training algorithm (Witten and Frank, 2005).[4] To assess the trained model's performance, we carried out a 10-fold cross-validation on our data set.[5] We present our results in the next section.

4.3 Results

We will present results for a trained decision tree model that reflects a specific precision/recall tradeoff. In particular, given our aim to enable an agent to sometimes initiate overlapping speech, while minimizing the chance of making a wrong assumption about the user's meaning, we selected a model with high precision at the expense of lower recall. Various precision/recall tradeoffs are possible in this framework; the choice of a specific tradeoff is likely to be system and domain-dependent and motivated by specific design goals.

We evaluate our model using several features which are exemplified in the appendix in Figure A-3. These include MAXF (PREDICTED), the trained MAXF classifier's output (TRUE or

[3] This corpus was not part of the training data for mxNLU.

[4] Of course, other classification models could be used.

[5] All the partial ASR results for a given utterance were constrained to lie within the same fold, to avoid training and testing on the same utterance.

FALSE) for each partial; K_{MAXF}, the first partial number for which MAXF (PREDICTED) is TRUE; $\Delta F(K) = F(K) - F(K_{\mathrm{final}})$, the "loss" in F-score associated with interpreting partial K rather than the final partial K_{final} for the utterance; $T(K)$, the remaining length (in seconds) in the user utterance at each partial.

We begin with a high level summary of the trained MAXF model's performance, before discussing more specific impacts of interest in the dialogue system. We found that our trained model predicts that MAXF = TRUE for at least one partial in 79.2% of the utterances in our corpus. For the remaining utterances, the trained model predicts MAXF = FALSE for all partials. The precision/recall/F-score of the trained MAXF model are 0.88/0.52/0.65 respectively. The high precision means that 88% of the time that the model predicts that F-score is maximized at a specific partial, it really is. On the other hand, the lower recall means that only 52% of the time that F-score is in fact maximized at a given partial does the model predict that it is.

For the 79.2% of utterances for which the trained model predicts MAXF = TRUE at some point, Figure 6 shows the amount of time in seconds, $T(K_{\mathrm{MAXF}})$, that remains in the user utterance at the time partial K_{MAXF} becomes available from the ASR. The mean value is 1.6 seconds; as the figure shows, the time remaining varies from 0 to nearly 8 seconds per utterance. This represents a substantial amount of time that an agent could use strategically, for example by immediately initiating overlapping speech (perhaps in an attempt to improve communication efficiency), or by exploiting this time to plan an optimal response to the user's utterance.

However, it is also important to understand the cost associated with interpreting partial K_{MAXF} rather than waiting to interpret the final ASR result K_{final} for the utterance. We therefore analyzed the distribution in $\Delta F(K_{\mathrm{MAXF}}) = F(K_{\mathrm{MAXF}}) - F(K_{\mathrm{final}})$. This value is at least 0.0 if mxNLU's output for partial K_{MAXF} is no worse than its output for K_{final} (as intended). The distribution is given in Figure 7. As the figure shows, 62.35% of the time (the median case), there is no difference in F-score associated with interpreting K_{MAXF} rather than K_{final}. 10.67% of the time, there is a loss of -1, which corresponds to a completely incorrect frame at K_{MAXF} but a completely correct frame at K_{final}. The converse also happens 2.52% of the time: mxNLU's output frame is completely correct at the early partial but completely incorrect at the final partial. The remaining cases are mixed. While the median is no change in F-score, the mean case is a loss in F-score of -0.1484. This is the mean penalty in NLU performance that could be paid in exchange for the potential gain in communication efficiency suggested by Figure 6.

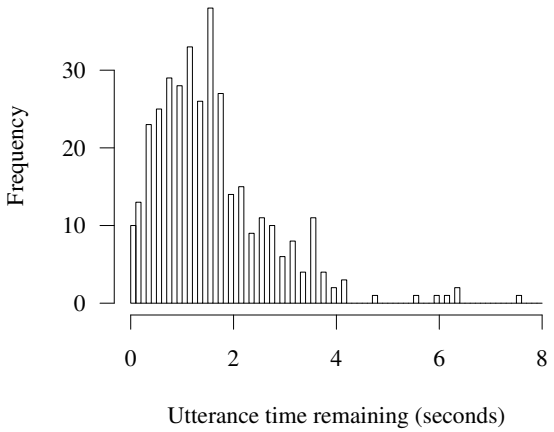

Figure 6: Distribution of $T(K_{\mathrm{MAXF}})$.

$\Delta F(K_{\mathrm{MAXF}})$ range	Percent of utterances
-1	10.67%
$(-1, 0)$	17.13%
0	62.35%
$(0, 1)$	7.30%
1	2.52%
mean($\Delta F(K_{\mathrm{MAXF}})$)	-0.1484
median($\Delta F(K_{\mathrm{MAXF}})$)	0.0000

Figure 7: The distribution in $\Delta F(K_{\mathrm{MAXF}})$, the "loss" associated with interpreting partial K_{MAXF} rather than K_{final}.

5 Prototype implementation

To illustrate one use of the techniques described in the previous sections, we have implemented a prototype module that performs *user utterance completion*. This allows an agent to jump in during a user's utterance, and say a completion of the utterance before it is finished, at a point when the agent

thinks it understands what the user means. This type of completion is often encountered in human-human dialogue, and may be used, for example, for grounding or for bringing the other party's turn to a conclusion.

We have equipped one of our virtual humans, Doctor Perez, with an ability to perform completions as follows. The first step is for the agent to recognize when it understands what the user wants to say. As discussed in Sections 3 and 4, this often happens before the user has completed the utterance. NLU is performed on partial ASR hypotheses as they become available, and MAXF decides whether the agent's understanding of the current partial hypothesis is likely to improve given more time. Once MAXF indicates that the agent's understanding is likely to be already maximized for the utterance, we take the current partial ASR hypothesis and attempt to generate text to complete it in a way that is fluent and agrees with the meaning of the utterance the user has in mind.

The generation of the surface text for completions takes advantage of the manual transcriptions in the corpus of utterances used to train the NLU module. For each frame that the agent understands, our training set contains several user utterances that correspond to the meaning in that frame. At the point where the agent is ready to formulate a completion, mxNLU has already predicted a frame for the user's utterance (even though it is still incomplete). We then consider only the set of known utterances that correspond to that frame as possible sources of completions. As a simple distance metric, we compute the word error rate (WER) between the current partial hypothesis for the user's utterance and a prefix of each of these known utterances. In our prototype, these prefixes have the same length as the current partial ASR hypothesis. We then select the utterance whose prefix has the lowest WER against the current partial ASR hypothesis. As a final step, we look in the prefix of our selected utterance for the last occurrence of the last word in the partial ASR, and if such a word is found, we take the remainder of the utterance as the agent's completion. Considering only the set of utterances that correspond to the frame predicted by mxNLU makes it likely that the completion will have the appropriate meaning. Since the completion is a suffix of a transcript of a previous user utterance, and this suffix follows the last word uttered by the user, it is likely to form a fluent completion of the user's partial utterance.

For 356 utterances in our development set, the procedure above generated completions containing at least one word for 177 utterances. We list below four representative examples of user utterance completions generated by Doctor Perez:

1. Partial utterance: *we need to*
   Predicted completion: *move your clinic*
   Actual user completion: *move the clinic*

2. Partial utterance: *I have orders*
   Predicted completion: *to move you and this clinic*
   Actual user completion: *to help you in moving the clinic to a new location*

3. Partial utterance: *the market*
   Predicted completion: *is not safe*
   Actual user completion: *is not a safer location*

4. Partial utterance: *we can also*
   Predicted completion: *give you medical supplies*
   Actual user completion: *build you a well*

Although agent completions for user utterances are often acceptably accurate (examples 1-3), occasionally they are incorrect (example 4). The frequency of appropriate completions depends on the accuracy of mxNLU and the chosen precision/recall tradeoff in MAXF. Finally, although the agent has the ability to generate these completions, clearly it should not complete the user's utterance at every opportunity. Determining a policy that results in natural behavior with respect to the frequency of completions for different types of agents is a topic under current investigation.

6 Summary and future work

We have presented a framework for interpretation of partial ASR hypotheses of user utterances, and high-precision identification of points within user utterances where the system already understands the intended meaning. Our initial implementation of an utterance completion ability for a virtual human serves to illustrate the capabilities of this framework, but only scratches the surface of the new range of dialogue behaviors and strategies it allows.

Immediate future work includes the design of policies for completions and interruptions that re-

sult in natural conversational behavior. Other applications of this work include the generation of paraphrases that can be used for grounding, in addition to extra-linguistic behavior during user utterances, such as head nods and head shakes.

Acknowledgments

The project or effort described here has been sponsored by the U.S. Army Research, Development, and Engineering Command (RDECOM). Statements and opinions expressed do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred. We would also like to thank Anton Leuski for facilitating the use of incremental speech results, and David Schlangen and the ICT dialogue group, for helpful discussions.

References

Jens Allwood, Joakim Nivre, and Elisabeth Ahlsen. 1992. On the semantics and pragmatics of linguistic feedback. *Journal of Semantics*, 9.

Jens Allwood. 1995. An activity based approach to pragmatics. Technical Report (GPTL) 75, Gothenburg Papers in Theoretical Linguistics, University of Göteborg.

Adam L. Berger, Stephen D. Della Pietra, and Vincent J. D. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Justine Cassell, Joseph Sullivan, Scott Prevost, and Elizabeth Churchill, editors. 2000. *Embodied Conversational Agents*. MIT Press, Cambridge, MA.

Herbert H. Clark and Edward F. Schaefer. 1987. Collaborating on contributions to conversation. *Language and Cognitive Processes*, 2:1–23.

Herbert H. Clark and Deanna Wilkes-Gibbs. 1986. Referring as a collaborative process. *Cognition*, 22:1–39. Also appears as Chapter 4 in (Clark, 1992).

Herbert H. Clark. 1992. *Arenas of Language Use*. University of Chicago Press.

B. Hansen, D. Novick, and S. Sutton. 1996. Prevention and repair of breakdowns in a simple task domain. In *Proceedings of the* AAAI-96 Workshop on Detecting, Repairing, and Preventing Human-Machine Miscommunication, pages 5–12.

A. Hartholt, T. Russ, D. Traum, E. Hovy, and S. Robinson. 2008. A common ground for virtual humans: Using an ontology in a natural language oriented virtual human architecture. In *Language Resources and Evaluation Conference (LREC)*, May.

A. Leuski and D. Traum. 2008. A statistical approach for text processing in virtual humans. In *26th Army Science Conference*.

Yukiko I. Nakano, Gabe Reinstein, Tom Stocky, and Justine Cassell. 2003. Towards a model of face-to-face grounding. In *ACL*, pages 553–561.

Jeff Rickel and W. Lewis Johnson. 1999. Virtual humans for team training in virtual reality. In *Proceedings of the Ninth International Conference on Artificial Intelligence in Education*, pages 578–585. IOS Press.

H. Sacks, E. A. Schegloff, and G. Jefferson. 1974. A simplest systematics for the organization of turn-taking for conversation. *Language*, 50:696–735.

M. D. Sadek. 1991. Dialogue acts are rational plans. In *Proceedings of the ESCA/ETR workshop on multi-modal dialogue*.

K. Sagae, G. Christian, D. DeVault, and D. R. Traum. 2009. Towards natural language understanding of partial speech recognition results in dialogue systems. In Short Paper Proceedings of NAACL HLT.

Gabriel Skantze and David Schlangen. 2009. Incremental dialogue processing in a micro-domain. In *Proceedings of EACL 2009*, pages 745–753.

D. Traum, W. Swartout, J. Gratch, and S. Marsella. 2008a. A virtual human dialogue model for non-team interaction. In L. Dybkjaer and W. Minker, editors, *Recent Trends in Discourse and Dialogue*. Springer.

D. R. Traum, S. Marsella, J. Gratch, J. Lee, and A. Hartholt. 2008b. Multi-party, multi-issue, multi-strategy negotiation for multi-modal virtual agents. In Helmut Prendinger, James C. Lester, and Mitsuru Ishizuka, editors, *IVA*, volume 5208 of *Lecture Notes in Computer Science*, pages 117–130. Springer.

I. H. Witten and E. Frank. 2005. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.

Victor H. Yngve. 1970. On getting a word in edgewise. In *Papers from the Sixth Regional Meeting*, pages 567–78. Chicago Linguistic Society.

A Appendix

1	C	Hello Doctor Perez.
2	D	Hello captain.
3	E	Hello captain.
4	C	Thank you for meeting me.
5	E	How may I help you?
6	C	I have orders to move this clinic to a camp near the US base.
7	E	We have many matters to attend to.
8	C	I understand, but it is imperative that we move the clinic out of this area.
9	E	This town needs a clinic.
10	D	We can't take sides.
11	C	Would you be willing to move downtown?
12	E	We would need to improve water access in the downtown area, captain.
13	C	We can dig a well for you.
14	D	Captain, we need medical supplies in order to run the clinic downtown.
15	C	We can deliver medical supplies downtown, Doctor.
16	E	We need to address the lack of power downtown.
17	C	We can provide you with power generators.
18	E	Very well captain, I agree to have the clinic downtown.
19	E	Doctor, I think you should run the clinic downtown.
20	D	Elder, the clinic downtown should be in an acceptable condition before we move.
21	E	I can renovate the downtown clinic, Doctor.
22	D	OK, I agree to run the clinic downtown, captain.
23	C	Excellent.
24	D	I must go now.
25	E	I must attend to other matters.
26	C	Goodbye.
26	D	Goodbye.
26	E	Farewell, sir.

Figure A-1: Successful negotiation dialogue between C, a captain (human trainee), D, a doctor (virtual human), and E, a village elder (virtual human).

		MAXF model training features					
Partial ASR result	$F(K)$	K	N	Entropy	$P_{\max}$	NLU	MAXF (GOLD)
(empty)	0.00	1	0	2.96	0.48	f82	FALSE
(empty)	0.00	2	0	2.96	0.48	f82	FALSE
all	0.00	3	1	0.82	0.76	f72	FALSE
elder	0.00	4	1	0.08	0.98	f39	FALSE
elder do you	0.83	5	3	1.50	0.40	f68	FALSE
elder to you d	0.50	6	3	1.31	0.75	f69	FALSE
elder do you agree	0.83	7	4	1.84	0.35	f68	FALSE
elder do you agree to	0.83	8	5	1.40	0.61	f68	FALSE
elder do you agree to move the	0.91	9	7	0.94	0.49	f10	TRUE
elder do you agree to move the	0.91	10	7	0.94	0.49	f10	TRUE
elder do you agree to move the clinic to	0.83	11	9	1.10	0.58	f68	FALSE
elder do you agree to move the clinic down	0.83	12	9	1.14	0.66	f68	FALSE
elder do you agree to move the clinic downtown	0.91	13	9	0.50	0.89	f10	TRUE
elder do you agree to move the clinic downtown	0.91	14	9	0.50	0.89	f10	TRUE

Figure A-2: Features used to train the MAXF model.

		MAXF model evaluation features		
K	$F(K)$	$\Delta F(K)$	$T(K)$	MAXF (PREDICTED)
1	0.00	-0.91	2.6	FALSE
2	0.00	-0.91	2.4	FALSE
3	0.00	-0.91	2.2	FALSE
4	0.00	-0.91	2.0	FALSE
5	0.83	-0.08	1.8	FALSE
6	0.50	-0.41	1.6	FALSE
7	0.83	-0.08	1.4	FALSE
8	0.83	-0.08	1.2	FALSE
9 ($= K_{\mathrm{MAXF}}$)	0.91	0.00 (=$\Delta F(K_{\mathrm{MAXF}})$)	1.0	TRUE
10	0.91	0.00	0.8	TRUE
11	0.83	-0.08	0.6	FALSE
12	0.83	-0.08	0.4	FALSE
13	0.91	0.00	0.2	TRUE
14	0.91	0.00	0.0	TRUE

Figure A-3: Features used to evaluate the MAXF model.