Incremental dialogue system faster than and preferred to its nonincremental counterpart

Gregory Aist[1] (gregory.aist@asu.edu), James Allen[2] (james@cs.rochester.edu), Ellen Campana[2,3,4,5] (ecampana@bcs.rochester.edu), Carlos Gomez Gallo[2] (cgomez@cs.rochester.edu), Scott Stoness[2] (stoness@cs.rochester.edu), Mary Swift[2] (swift@cs.rochester.edu), and Michael K. Tanenhaus[3] (mtan@bcs.rochester.edu)

[1]Department of Computer Science and Engineering Arizona State University P.O. Box 878809 Tempe AZ 85287

[2]Department of Computer Science University of Rochester P.O. Box 270226 Rochester NY 14627

[3]Department of Brain and Cognitive Sciences University of Rochester P.O. Box 270268 Rochester, NY 14627

[4]Arts, Media, and Engineering Program Arizona State University P.O. Box 878709 Tempe AZ 85287

[5]Department of Psychology Arizona State University P.O. Box 871104 Tempe AZ 85287

Abstract

Current dialogue systems generally operate in a pipelined, modular fashion on one complete utterance at a time. Evidence from human language understanding shows that human understanding operates incrementally and makes use of multiple sources of information during the parsing process, including traditionally "later" components such as pragmatics. In this paper we describe a spoken dialogue system that understands language incrementally, provides visual feedback on possible referents during the course of the user's utterance, and allows for overlapping speech and actions. We further present findings from an empirical study showing that the resulting dialogue system is faster overall than its nonincremental counterpart. Furthermore, the incremental system is preferred to its nonincremental counterpart – beyond what is accounted for by factors such as speed and accuracy. These results indicate that successful incremental understanding systems will improve both performance and usability.

Keywords: natural language understanding; dialogue systems; incremental processing.

Introduction

The standard model of natural language understanding for dialogue systems is pipelined, modular, and operates on complete utterances. By pipelined we mean that only one level of processing operates at a time, in a sequential manner. By modular, we mean that each level of processing depends only on the previous level. By complete utterances we mean that the system operates on one sentence at a time.

There is, however, considerable evidence that human language processing is neither pipelined nor modular nor whole-utterance (Marslen-Wilson 1993). Evidence is converging from a variety of sources, including particularly actions taken while speech arrives. For example, natural turn-taking behavior such as backchanneling (uh-huh) and interruption occur while the speaker is still speaking. Eye movements to possible referents also occur while listening: individuals process instructions incrementally, making saccadic eye movements to objects right after hearing relevant words in the instruction (Tanenhaus et al. 1995); verbs appearing earlier in sentences affect which objects are brought into context, as determined by hearer eye fixations (Altmann and Kamide 1999). Other actions can also be taken based on partial utterances.

Many different sources of knowledge are available for use in understanding. On the speech recognition side, commonly used sources of information include acoustics, phonetics and phonemics, lexical probability, and word order. In dialogue systems, additional sources of information often include syntax and semantics (both general and domain-specific.)

There are also however some sources of information that are less frequently programmed. These include such linguistic information as morphology and prosody. Knowledge-based features are also available, such as world knowledge (triangles have three sides), domain knowledge (here there are two sizes of triangles), and task knowledge (the next step is to click on a small triangle.) There is also pragmatic information available from the visual context (there is a small triangle near the flag.)

In this paper we discuss some of the progress we have made towards building methods for incremental understanding of spoken language by machines. We first discuss some of our and others' related work in this area. We then discuss the testbed domain that we have been developing, and show some of the characteristics of human dialogue in the domain. We then discuss the incremental architecture that we have been developing, highlighting its differences from traditional architectures. Finally, we present an experimental evaluation of the performance of the system showing that incremental systems are both faster than and preferred to their nonincremental counterparts.

Related Work

We have previously shown that incremental parsing can be faster and more accurate than non-incremental parsing (Stoness et al. 2005.) In addition, we have shown that in our testbed domain the relative percentage of language that is of a more interactive style also increases over time (Aist et al. 2005.)

1 okay so
2 we're going to put a large triangle with nothing into morningside
3 we're going to make it blue
4 and rotate it to the left forty five degrees
5 take one tomato and put it in the center of that triangle
6 take two avocados and put it in the bottom of that triangle
7 and move that entire set a little bit to the left and down
8 mmkay
9 now take a small square with a heart on the corner
10 put it onto the flag area in central park
11 rotate it a little more than forty five degrees to the left
12 now make it brown
13 and put a tomato in the center of it
14 yeah that's good
15 and we'll take a square with a diamond on the corner
16 small
17 put it in oceanview terrace
18 rotate it to the right forty five degrees
19 make it orange
20 take two grapefruit and put them inside that square
21 now take a triangle with the star in the center
22 small
23 put it in oceanview just to the left of oceanview terrace
24 and rotate it left ninety degrees
25 okay
26 and put two cucumbers in that triangle
27 and make the color of the triangle purple

Figure 1. Example human-human dialogue in the fruit carts domain.

Higashinaka et al. (2002) performed a linear regression experiment to find a set of features that predict performance of systems that understand utterances incrementally. The system evaluated by the authors is incremental in that dialogue states are updated as the sentence is processed. However this is a result of incrementally processing the input stream and not the type of continuous understanding we propose. In our approach we allow the parser to make use of information from different layers of processing (i.e. pragmatic constraints from verb-argument constructions, real world knowledge, etc).

Rosé et al. (2002) describe a reworking of a chart parser so that "as the text is progressively revised, only minimal changes are made to the chart". They found that incrementally parsing incoming text allows for the parsing time to be folded into the time it takes to type, which can be substantial especially for longer user responses. Our current work operates on spoken input as well as typed input and makes extensive use of the visual context and of pragmatic constraints during parsing.

DeVault and Stone (2003) describe techniques for incremental interpretation that involve annotating edges in a parser's chart with constraints of various types that must be met for to the edge to be valid. That has a clean and nice simplicity to it, but seems to impose uniformity on the sorts of information and reasoning that can be applied to the

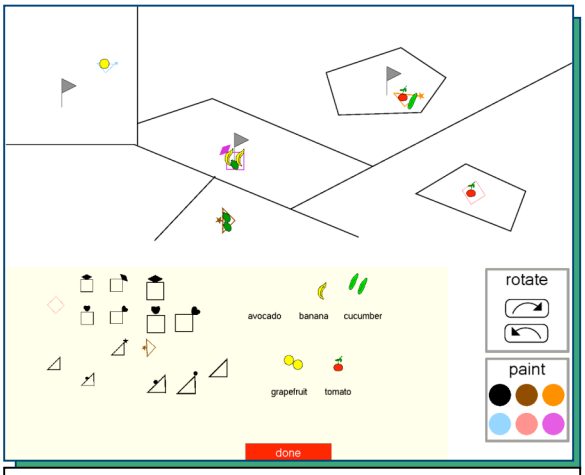

Figure 2. Fruit carts example screen.

parsing process. In our approach, advice to the parser is represented as modifications to the chart, and can thus be in any framework best for the source.

Previous work by Schuler (2001, 2002, 2003) has moved away from a pipeline architecture by accessing different sources of knowledge while parsing the sentence. Using real world knowledge about objects improves parsing and can only be achieved by analyzing the sentence from the start. Schuler makes use of potential referents from the environment much the same way that we have also done by the use of model-theoretic interpretations. Thus the system evaluates the logical expressions for all possible potential referents at each node of the tree to know whether they are possible in the current domain. The author provides an example where a prepositional phrase attachment ambiguity is resolved by knowing a particular fact about the world which rules out one of the two possible attachments. Thus this sort of knowledge comes into play during parsing. Even though the system described in the present paper shares the same goals in using more than just syntactic knowledge for parsing, our parser feedback framework does not require the rewriting of the grammar used for parsing to incorporate environment knowledge. This approach based on probability feedback directly affecting the parser chart is simpler and thus more applicable to and easily incorporated in a wider range of parsers and grammars.

Testbed Domain: Fruit Carts

To explore the effects of incremental understanding in human-computer dialogue, we devised a testbed domain (Figures 1 and 2) where a person gives spoken instructions to a computer in order to replicate a goal map (Aist 2004). On the map, there are named regions, some of which contain flags as landmarks; the screen also has two kinds of objects: abstract shapes such as triangles and squares, and "fruit" of various kinds (avocados, bananas, cucumbers, grapefruits, and tomatoes.) In this domain, certain steps were taken in order to reduce complexity and increase the

predictability of the spoken language. In particular, all objects and names of regions were chosen to be easy to name (or read) and easy for the speech recognizer to hear. Human-human dialogue collected in this domain was used in the construction of the dialogue system. An example of the human-human dialogue is shown in Figure 1.

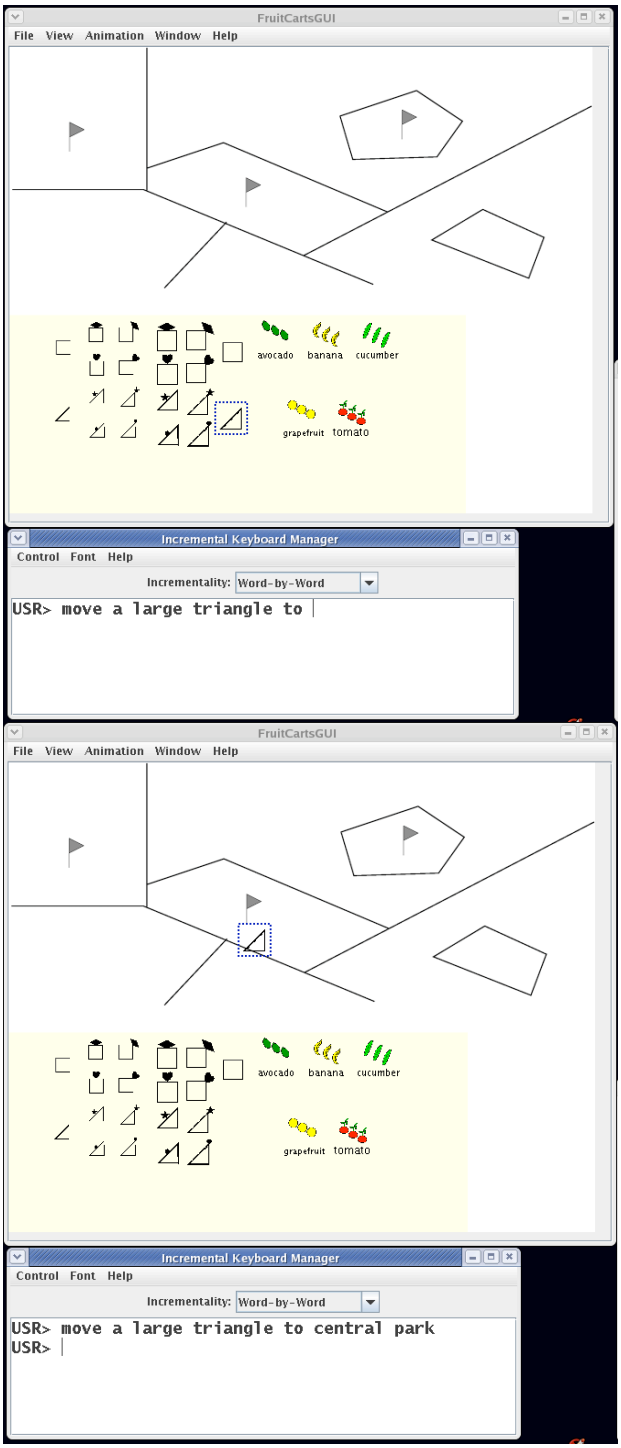

Figure 3. An example interaction with the incremental dialogue system. Note that in the top screenshot, halfway through the sentence, the large triangle is already highlighted.

We collected a set of dialogs from human-human conversation in this domain. Our observations included the following:

1. End-of-sentence boundaries tend to be fairly clear (at least to a human listener). Where a sentence begins, however, is quite difficult to say precisely, due to disfluencies, abandoned utterances, and so forth. This is in contrast to domains where speakers might tend to begin a sentence clearly, such as information retrieval ("Search for books by Kurt Vonnegut").

2. There seem to be two distinct strategies that people can employ: saying a direction all at once ("Put it one inch below the flag") or continuously ("Put it near the flag [pause] but down a bit [pause] a bit more [pause] stop.")

3. Besides a pure All-at-once and Continuous strategy, people sometimes switch between them, employing Both. For example, the director might tell the actor to place an object "right on the flag [pause] down a bit [pause] keep going [pause] stop." We see these as possibilities along a continuum, using the same language mechanisms yet according different emphasis to the strategies.

Our previous findings about these types of language include that continuous-style language uses fewer words per utterance than all-at-once language, and the words themselves are shorter in length as well. Furthermore, the use of continuous language increases over the course of the dialogs. Specifically, the relative percentage of continuous language increases over trials. The relative increase in continuous language over time is statistically significant (by logistic regression; style as outcome, subject as categorical, trial as numeric. $B=0.104 \pm 0.037$, $\exp(B) \approx 1.11$, $p < 0.01$) This is somewhat counterintuitive: it is well-known that interlocutors establish and refine referring expressions (Clark & Wilkes-Gibbs 1986), which ought to result in all-at-once language being easier to use; continuous language must be very compelling.

We used these human-human conversations to form the basis for formalizing various aspects of continuous understanding, and for gauging the behavior of the spoken dialog system that we built to operate in this testbed domain. The resulting system is capable of interactions as shown in Figure 3, where the user's utterance is processed as it is received, visual feedback is provided during the course of the utterance, and speech and actions can overlap. As in the human-human interactions, moving an object from one location to another takes time in the working system – that is, the objects are shown moving in a straight line from the beginning point (e.g. the bin at the bottom of the screen) to the end point (the flag in central park.)

Traditional vs. Incremental Architecture

Figure 4 shows a diagram of our incremental architecture for dialogue systems, as contrasted to a traditional dialogue system architecture.

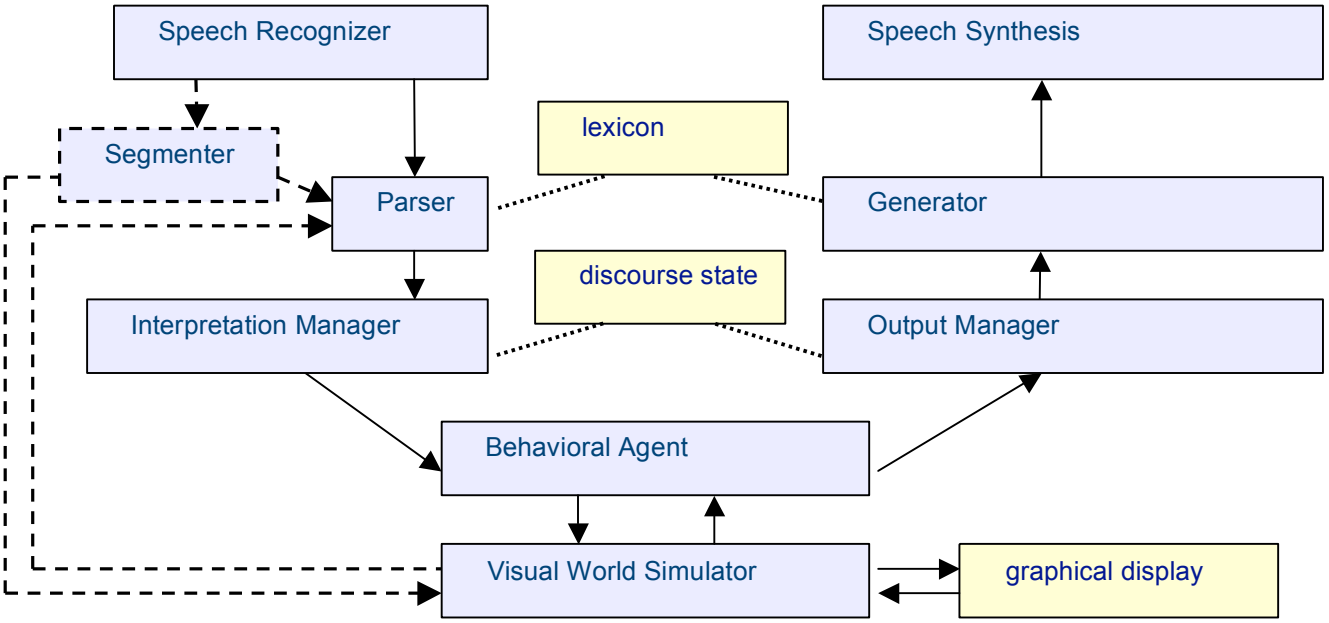

Figure 4. Changes to spoken dialogue system architecture to allow incremental understanding.
Boxes show components; lines show message flow.
In both types of systems, the lexicon and the discourse state are resources shared by input and output.
Components and connections new to the incremental system are shown in **dashed lines**.
Incremental understanding also places requirements on the speech recognizer (production of partial hypotheses), the parser (incremental construction of charts), the interpretation manager and behavioral agent (handling partial interpretations and actions), and the visual world simulator (incorporation of semantic models of partial actions) which are also important to the overall functioning of the system.
This paper focuses on incremental *understanding* and thus the changes are to the *understanding* aspects of the dialogue system, including action-taking as representing direct evidence of understanding.

Incremental language processing as we conceive it involves a number of fundamental and inter-related changes to the way in which language understanding and generation occurs:
(a) input sentences are processed before user turn ends, as opposed to processing only when turn is finished;
(b) components of the architecture operate asynchronously with several operating simultaneously, in contrast to a serial one where only one module at a time can be active;
(c) knowledge resources are available to several components at the same time, in contrast to a "pipeline" architecture where knowledge is sent from module to module;
(d) there is overlapping speech and graphical output ("action"), in contrast to presenting speech and other output sequentially;
(e) system and user turns and actions can overlap as appropriate for the dialogue.
We discuss some of the these distinctions in more detail below.

In a traditional dialogue system architecture, each component processes input from other components one utterance at a time. In our incremental architecture, each component receives input from other components as available, in whatever amounts arrive.

In a traditional system, each component feeds forward into other components. In our incremental architecture, each component advises other components as needed – and advice can flow both "forward" in the traditional directions and "backward" from traditionally later stages of processing (such as pragmatics) to traditionally earlier stages of processing (such as parsing.)

In a traditional system, the internal representations assume a strict division of time according to what's happening – the system is speaking, or the user is speaking, or the system is acting, and so forth. In our incremental architecture, representations can accommodate multiple events happening at once – such as the system acting while the user is still speaking.

In addition to these overall changes, our system incorporates a number of specific changes.

1. A Segmenter (Aist 2006) operates on incoming words, identifies pragmatically relevant fragments, and announces them to other system components such as the parser and the visual display.

2. Pragmatic information is provided to the parser in order to assist with ongoing parses (Stoness et al. 2005).

3. Modeling of actions and events is done by means of incremental semantics (Aist, Stoness, and Allen 2006), in

order to properly represent partial actions and allow for overlapping actions and speech.

4. Visual feedback is provided to the user about possible referents while the user is speaking (Figure 3).

Experiment 1: Speed of Incremental System vs. Nonincremental Counterpart

We conducted a controlled evaluation comparing incremental understanding to its nonincremental counterpart in our testbed domain. In the nonincremental system, speech and actions alternate; in the incremental system, the actions and speech overlap.

A total of 22 dialogues were collected, each of which consisted of two utterances and the corresponding system responses. Eleven of the dialogues were in the control (nonincremental) condition and eleven of the dialogues were in the experimental (incremental) condition. The utterances were in-domain and understandable by both the nonincremental and incremental versions of the system. The utterances were pre-recorded, and the same utterances were played to each version of the system. This technique allowed us to minimize variance due to extraneous factors such as interspeaker variability, acoustic noise, and so forth, and concentrate specifically on the difference between incremental understanding and its nonincremental counterpart. The resulting dialogues were recorded on digital video.

The incremental system was approximately 20% faster than the nonincremental system in terms of time to task completion, at 44 seconds per dialogue vs. 53 seconds for the control condition (single-factor ANOVA, F=10.72, df=21, p-value 0.004).

Experiment 2: Ratings of Incremental System vs. Nonincremental Counterpart

To further evaluate the effectiveness of the incremental system, we conducted an onlooker study where 18 subjects, mostly from the University of Rochester community, rated the interactions in the dialogues. First, each subject watched one video clip once and only once; then, the subject filled out written responses to questions about that video clip. In order to situate the present study with respect to other methods of evaluation of dialogue systems, we compared results from our experiment with the PARADISE model of dialogue system evaluation (Walker et al. 1997): that speed, accuracy, and match to user intentions well predict user satisfaction. Thus subjects provided responses for each dialogue video clip to each of four questions on speed, accuracy, match-to-intent, and satisfaction:

[FAST] "How fast did the computer respond?"
[ACC] "How accurately did the system understand?"
[ACT] "How well matched were the computer's actions to what the person wanted?"
[SAT] "If you had been the person giving the commands, how satisfied overall would you be with the interaction?"

Each response was provided on a Likert scale from 1 to 7, with 1 being "less fast", "less accurate", and so forth.

In order to check that people's responses were objectively correlated with actual system performance, four "wrong" system videos were included in the study, two for each condition (nonincremental control and incremental/experimental condition). That is, the user in the video said one thing, but the system did something else. To say that in another way, we experimentally manipulated the "right/wrong" response of the system to see how people would rate the system based on correctness.

Using a linear regression model as in the original PARADISE framework, we confirmed that a linear model with speed (FAST), accuracy (ACC), and match-to-actions (ACT) as input variables predicts well the output variable satisfaction (SAT) (R=.795, R Square=.631, Adj. R Square=.625; df=3, F=91.389, $p<0.001$; this and all subsequent statistical analyses performed in SPSS). Thus we replicated the main findings of Walker et al. with the experimental technique of the onlooker study.

Given the nature of the input and output variables – seven-item Likert scale responses – it turns out to be the case that ordinal regression models are a better match to the experimental setup than the linear regression models.

Table 1. Ordinal regression model showing relationship between satisfaction (SAT) and right/wrong system responses, taking various factors into account.

Variable	Parameter Estimate	Std. Error	Sig.	Notes
NTH	.188	.058	.001	
FAST	.770	.176	.000	
ACC	1.411	.341	.000	
ACT	.616	.304	.043	
RIGHT=0 (0=wrong, 1=right.)	-1.855	.903	.040	Negative number means wrong responses are negatively correlated with user satisfaction.
INC=0 (0=control 1=incr.)	-2.336	1.051	.026	Negative number means nonincremental processing is negatively correlated with user satisfaction.

Ordinal regression models are specifically designed for cases where the variables are a set of levels that are ordered (N+1>N) but not necessarily linear (1 to 2 may not be the same as 4 to 5.) We thus adopted ordinal regression models for the remainder of the analyses. In addition, since some of the subjects indicated in written comments that they got used to the behavior of the system over time, we included the dialogue number (NTH; 1=first seen, 22=last seen) as a covariate. And, since individual subjects tend to vary in their responses (some subjects being more critical in general than other subjects), we included subject (SUBJ) as an input variable as well.

The model we built to analyze the effects of right/wrong system response (RIGHT) and nonincremental vs. incremental processing (INC) was as follows. We built an ordinal regression model predicting satisfaction (SAT) by right/wrong (RIGHT) and nonincremental/incremental (INC) and subject (SUBJ) with FAST, ACC, and ACT as covariates. The model is shown in Table 1.

The first result we found was that there was a significant effect for RIGHT as a predictor of user satisfaction, in the expected direction: wrong responses predict lower satisfaction (or, equivalently, correct responses predict higher satisfaction.) These results serve as validation of the external reliability of the experimental design.

Next, to evaluate the effects of incremental vs. nonincremental processing, we examined the model coefficient for INC. In this case, nonincremental processing was a significant predictor of lower satisfaction (p=.026) – or, equivalently, incremental processing was a significant predictor of higher satisfaction.

Conclusion

Our results show that – at least for this task – incremental processing predicts higher user satisfaction. Why? The statistical model makes clear that this preference is the case after controlling for factors such as speed, accuracy, and match-to-intent. Explanatory factors that remain include particularly *naturalness* – that is, the ways in which incremental systems are more like human-human conversation than their nonincremental counterparts. Nonincremental dialogue systems require many artificial restrictions on what the user and the system can say and when they can say it, and therefore exclude many important characteristics of natural human dialogue. Incremental understanding has the potential to remove these obstacles. The work presented here suggests that successful incremental understanding systems will improve both performance and usability

References

Aist, G. (2004). Speech, gaze, and mouse data from choosing, placing, painting, rotating, and filling (virtual) vending carts. International Committee for Co-ordination and Standardisation of Speech Databases (COCOSDA) Workshop, Jeju Island, Korea, Oct. 4.

Aist, G.S., Campana, E., Allen, J., Rotondo, M., Swift, M., and Tanenhaus, M. 2005. Variations along the contextual continuum in task-oriented speech. Proceedings of the 27th Annual Conference of the Cognitive Science Society, Stresa, Italy, July. Paper number 769.

Aist, G. (2006). Incrementally segmenting incoming speech into pragmatic fragments. The Third Midwest Computational Linguistics Colloquium (MCLC-2006). Urbana-Champaign, Illinois. May 20-21.

Aist, G., Stoness, S., and Allen, J. (2006). Steps towards incremental semantics for spoken dialog systems. The Third Midwest Computational Linguistics Colloquium (MCLC-2006). Urbana-Champaign, Illinois. May 20-21.

Altmann, G. and Kamide, Y. (1999). Incremental interpretation at verbs: restricting the domain of subsequent reference. *Cognition* 73(3):247-264.

Clark, H.H. and Wilkes-Gibbs, D. (1986). Referring as a collaborative process. *Cognition* 22:1–39.

DeVault, D. and Stone, M. (2003). Domain inference in incremental interpretation. ICOS 2003.

Higashinaka, R., Miyazaki N., Nakano, M., and Kiyoaki, A. (2002). A method for evaluating incremental utterance understanding in spoken dialogue systems. ICSLP 2002.

Marslen-Wilson, W. D. (1993). Issues of process and representation in lexical access. In G. T. M. Altmann & R. Shillcock (Eds.), Cognitive models of speech processing: The second Sperlonga meeting (pp. 187-210). Hillsdale, NJ: Lawrence Erlbaum Associates Inc.

Rosé, C.P., Roque, A., Bhembe, D., and Van Lehn, K. (2002). An efficient incremental architecture for robust interpretation. HLT 2002.

Schuler, W. (2001). Computational properties of environment-based disambiguation. ACL 2001.

Schuler, W. (2002). Interleaved semantic interpretation in environment-based parsing. COLING 2002.

Schuler, W. (2003). Using model-theoretic semantic interpretation to guide statistical parsing and word recognition in a spoken language interface. ACL 2003.

Stoness, S.C., Allen, J., Aist, G., and Swift, M. (2005). Using real-world reference to improve spoken language understanding. AAAI Workshop on Spoken Language Understanding, Pittsburgh, Pennsylvania, July. pp. 38-45.

Tanenhaus, M.K., Spivey-Knowlton, M.J., Eberhard, K.M., and Sedivy, J.C. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science*, Vol. 268 (5217), 1632-1634.

Walker, M., Litman, D., Kamm C., and Abella, A. (1997). PARADISE: A Framework for Evaluating Spoken Dialogue Agents. ACL 1997.