Do We Need Deep Generation of Disfluent Dialogue?

Charles B. Callaway
ITC-irst, TCC Division
via Sommarive, 18
Povo (TN), Italy, I-38050
http://tcc.itc.it/people/callaway
callaway@itc.it

Abstract

Most Natural Language Generation systems focus on the production of perfectly written text for visual presentation. But as NLG tries to move into other areas of language production, such as speech in dialogue systems, it is not clear that approaches for traditional applications are best. Speech exhibits particular features like disfluency that are not found in written discourse. In this paper, we thus present an initial look at what may be necessary to allow current NLG systems to seamlessly make the transition to spoken language production.

1 Introduction

For decades, the primary interest of the Natural Language Generation community has been the production of perfect texts. Generation was envisioned to automatically create documentation, stories, newspaper summaries, informative presentations, *etc.*. But the last ten years have seen the introduction of interactive dialogue systems with a robust enough recognition capability that they are beginning to be in need of sophisticated language generation models to make full use of their abilities. But will this integration be as easy as plugging an existing NLG module into an existing dialogue system?

To mimic human behavior, dialogue systems must reproduce a linguistic phenomenon distinct from that which NLG systems currently try to emulate. Rather than speaking perfectly, human dialogue participants make mistakes, acknowledge, show signs of dynamic replanning in the middle of utterances, take or concede the floor, ask questions for grounding purposes, and notice that other conversational participants are uncomfortable, upset, argumentative, etc.[Traum and Hinkelman, 1992; Heeman and Allen, 1999]

This leads to a significant difference in the coverage and distribution of linguistic phenomena for dialogue compared to written discourse [Ward, 1996; McKelvie, 1998]. Thus NLG systems that perform well in traditional NLG tasks may be poor candidates for integration with dialogue systems. For example, traditional NLG systems are poorly adapted at producing "sentences" like: "Ok." "Mmm-hmmm." and "Well... that's one way... maybe... how about this." [VanLehn, 1998] But while it may or may not be desirable to employ deep generation techniques to create phrases such as these, there are many instances of disfluencies in which deep generation should be far superior to the types of shallow (template) generation techniques used by most dialogue systems.

But first, more research must be done to see if intentionally generating disfluency is a valuable use of resources. In psycholinguistic studies, [Oviatt, 1995; Brennan, 2000] have advocated changing the architecture of dialogue systems to account for specific categories of disfluencies, arguing that this could improve user satisfaction, interaction time, and accuracy. However, there have been no detailed examinations of specific approaches, implemented prototypes, or experimental data to show exactly what types of architectural changes should be made, or even sample scenarios where generating certain disfluencies would be beneficial.

One example application where generating disfluencies could be useful is in intelligent tutoring systems that have animated agents which provide feedback to the student. One requirement for such agents is *believability*, in which an illusion is created where the student associates personality and intelligence with the animated agent [Lester *et al.*, 2000]. One of the most destructive aspects of this rapport is the poor quality of text-to-speech synthesizers, and the automatic generation of speech disfluency could be one step toward making such agents more believable.

Treating disfluencies as a linguistic phenomenon to be reproduced in functioning dialogue systems could be approached from two directions:

- Researchers in NLG can ask themselves whether they can duplicate the type of utterances routinely found in human-human dialogue transcripts, and if there are syntactic patterns to the underlying linguistic data of disfluencies that can be treated in a rigorous manner, requiring simple updates to existing NLG infrastructure.
- Researchers in dialogue systems can ask themselves whether their systems need more advanced generation techniques, and if so, can they produce the types of representations typically needed by NLG systems to function (*e.g.*, rhetorical structure, or ontological and lexical informaiton).

In this paper, we treat the first topic: can an existing generation system be easily modified to reproduce the types of syntactic phenomena commonly found in human-human dialogue? We thus start by examining related research at the conjunction of these two areas, presenting examples of disfluent dialogue which is not currently reproducible by NLG systems, and ending with a proposal for modifying an existing, large-scale NLG grammar to produce such disfluencies.

2 Related Areas of Research

In order to investigate experimentally whether the generation of disfluencies in human-computer dialogue will be of any benefit, three areas of current research are important:

- **Implementated dialogue systems**: Systems that already engage in the understanding of human utterances and the generation of system responses (travel planning assistants, intelligent tutoring systems, etc.)
- **Implemented generation systems**: Systems that create written text either for textual presentation or for eventual use by a text-to-speech system and that typically produce monologues instead of taking part in dialogues.
- **Linguistic analysis of disfluencies**: Studies on the coverage, distribution, constraints, and syntactic, semantic and pragmatic considerations of disfluencies as exhibited by humans when conversing.

The first two topics have been readily covered in the literature. For example, workshops have been held on spoken dialogue systems (*Interactive Spoken Dialog Systems* at ACL 1997 and *Conversational Systems* at ANLP/NAACL 2000. Workshops and, more recently, conferences on NLG have been held at least every other year for the last two decades (*e.g.*, INLG, EWNLG).

A purposefully short enumeration of these systems might include tutoring systems with written output [Freedman, 1996; Eugenio *et al.*, 2002], task-based spoken systems [Smith and Hipp, 1994; Allen *et al.*, 1996; Rambow *et al.*, 2001], embodied conversational systems [Cassell, 2001], and written narratives and dramas with inter-character dialogue [Walker *et al.*, 1997; Callaway, 2001; Callaway and Lester, 2002]. Much rarer are works that explicitly address the problem of generating system responses for dialogue systems [Carletta *et al.*, 1995; Jokinen and Wilcock, 1999; Cassell *et al.*, 2000; McRoy *et al.*, 2000; Oh and Rudnicky, 2000; Rambow *et al.*, 2001].

Similarly, less work has been undertaken on the incorporation of disfluencies in either the parsing or generation of dialogue or its underlying linguistic or psycholinguistic analysis. Research on disfluency in spoken dialogue is still very rare as evidenced by the ongoing work of the last decade in still foundational issues.

For example, [Oviatt, 1995] collected a corpus of task-oriented simulated dialogues, and noted that the number of disfluencies was significantly related to the length of the utterance. Oviatt then suggested that applied dialogue systems could take this into account by structuring the dialogue to reduce sentence length. Thus given two systems, the one which was better organized so that its users would have to speak in short sentences would then be more efficient because there would be less disfluency.

Similarly, [Brennan, 2000] took a psycholinguistic approach to disfluencies, noting that disfluencies contain useful semantic information, that disfluent speech is in certain cases easier to understand than fluent speech, that listeners made fewer errors when there was less incorrect information in a speech repair, and were faster to respond to a target word when the edit interval before the repair was longer. By this reasoning, a system could seem to the user to be more "confident in its message" if it could modify its generated output with silent pauses, appropriate fillers, and proper intonation.

On the generation side, [Zechner, 2001] approached the summarization of open domain spoken dialogues (using machine learning techniques) where speech disfluency is a major problem in producing summaries in that often the disfluencies were reproduced in the summaries themselves. A dialogue corpus was tagged with parts of speech as normally done when parsing with machine learning approaches, but speech disfluencies were also tagged, enabling the learning algorithm to drop them (and the repaired parts as well) in the summary.

Most importantly for the generation community is a syntactic approach like [McKelvie, 1998] that could allow deep generation systems, which already operate at a high degree of syntactic complexity, to reproduce disfluencies given the right syntactic and semantic input. McKelvie analyzes the HCRC Map Task Corpus with emphasis on its disfluencies, and then describes three major approaches for analyzing self-repairs, presents new rules for parsing them, and provides experimental results from an implemented parser using that corpus.

What is currently missing in disfluency research then is an implemented deep generation system which operates as part of a spoken dialogue system and that is capable of producing disfluencies which empirically improve task efficacy and user satisfaction levels. But before undertaking such an enterprise, it is important to be sure the effort is worth the payoff. Would researchers who develop spoken dialogue systems be interested in generated responses that include disfluencies?

3 NLG and Disfluency

Spoken dialogue systems are modelled after observed human-human dialogue. But with current implemented systems, computer-produced dialogue displays different linguistic phenomena: it is perfectly written, without the disfluencies found in the former. For the spoken dialogue practictioner, then, disfluency could be defined to be anything that facilitates the dialogue but that isn't syntactically perfect. This contrasts with researchers in NLG, for whom the world is filled with perfect utterances.

But while this may be true for the corpora that NLG researchers most commonly work with (documents and other presentably written materials), spoken dialogue is a different matter. Rather than seeing what has been written (and edited in the case of most human-written texts), users of spoken dialogue systems never see the output. But the ear is more forgiving than the eye; hearers are much less likely to notice natural disfluencies than to notice misspellings.

The question is then if NLG systems were capable of producing text marked up for disfluency that was then sent to a text-to-speech system, would the users of that system notice disfluencies any differently than for natural disfluencies, and would this be desirable (*i.e.*, as Oviatt and Brennan suggest, could this improve interactions with the system on a number of measures)? If the answer is yes, then disfluency as a subtopic of NLG deserves to be studied.

Disfluent recognition is the norm in real-world dialogue systems, both with spoken [Allen *et al.*, 1996; Rich and Sidner, 1998] and written [Person and Mathews, 2000] systems. However, in generation, there is no analog for disfluent NLG production. But there is a difference between using an NLG system as a plug-in component of a dialogue system and using it to help reproduce existing dialogue from corpora. In the former case, it will continue to produce perfect texts without pauses, repairs, or other types of disfluencies.

To give a more concrete aspect to how difficult the problem of generating disfluencies is, we present a series of examples from the Physics Corpus 1 [VanLehn, 1998]. Utterances in dialogue are typically considered from the point of view of dialogue or speech acts, *i.e.*, something is a request, persuade, confirm, etc. However, to regenerate such utterances, an NLG researcher must ask, "What would it take to reproduce this utterance's surface form?" rather than "Why did this utterance occur in this place?" In other words, when examining an utterance, what would be necessary to reproduce that utterance in that context with a given implemented NLG system? (For example, consider what would be needed to generate the system responses in Figures 1 and 2 [Allen, 1993].) Without considering disfluencies, the problem is not substantially different than that faced by the typical NLG system in typical contexts.

Consider the following examples from this perspective (all examples in this section are taken from [VanLehn, 1998], where "T" represents "Tutor" and "S" represents "Student"):

- Many typical dialogue acts don't need any type of generation, deep or shallow. Examples include acknowlegements and simple backchanneling:

  T: mm hmm. T: OK

- Simple hesitations often follow syntactic boundaries, which is easier for NLG systems to generate. The following example if "written" would merely replace the dots with a comma:

  T: Well... what is it?

- One type of disfluency, which computers probably don't need to duplicate, involves hesitation and the "tip of the tongue" phenomenon, where almost the entire sentence is repeated when the word is recalled:

  T: Well, which way is the... Which way is the velocity?

- The following is an example of a standard repair, where the minimal constituent before the error is repeated, followed by the correction. This is more difficult to generate, as it requires keeping track of the last constitutent, which is not done in generation systems, and being able to regenerate its minimal element.

  T: Oh, you probably want parentheses around... the parenthesis before the v sub final.

- This example shows dynamic replanning to include information that at one point was considered 'grounded' [Traum, 1994], but then was reclassified as unknown to both conversants. The original sentence is stopped, and an entirely new sentence is substituted after a slight pause and break in the middle of a word to indicate the erasure. Examples of this type are more of a problem with timing in the speech synthesizer than for generators.

  T: It's opp... If something's stopping, its acceleration is opposite the direction of its velocity.

- Replanning, in this case to correct student misconceptions, sometimes requires a large amount of cognitive resources, which is often covered by a series of false starts. It is difficult to imagine a generation system encountering the types of difficulties leading to the production of these types of utterances.

  S: I have are hard time seeing why N isn't in a negative y direction.

  T: N is the force that... N is... see... It's the force... It's not the force of the... It's not the force of the blocks on the inclined plane, it's the force of the inclined plane on the blocks.

- Leading questions try to elicit information that the student needs in order to solve the problem. These can be obtained by not finishing the entire sentence, although the problem then becomes finding at what point to terminate.

  T: OK, so... looking at the y component...

- A different type of "trailing off' disfluency involves tutoring strategies that are intended to lead the student to put together knowledge he/she already has learned previously. Frequently they are only combinations of fillers and noun phrase hints, completely lacking verbs.

  T: But... so... T... well...

- Repetition is a type of disfluency in spontaneous dialogue that can be considered a tool for emphasis in written text, or eloquent in formal speeches. Deciding which element to repeat is not an easy task, although once that decision is made, it is a simple matter to repeat the element.

  T: That's right, minus a minus 10 meters... per second.

  T: Yeah, straight down, straight down.

- A different type of disfluency involves more complicated backchanneling, such as signalling an offer to give up the floor. In this case, the dialogue manager does not know exactly when (or if) the other participant will take the offer, and thus must mark all possible cutoff points.

  T: But the tension in the string... you're gonna have... Well, you'll see what happens. You're gonna have... you're gonna have two variables... You're gonna have two equations in which two unknowns appear. And for the equation for the big block.

- When such an offer is accepted, the most important issue becomes the timing and control of the speech synthesizer, as the generator can simply ignore the remaining part of the utterance (except in the case of user models,

```
T: OK, uh... hold on one second.
S: Actually, I... did I miss something?
T: uh... you... that... that is fine, but do you remember that what you are
   trying to write is the summation of forces equals ma?
S: Yeah, yeah.
T: So, you forgot the equals ma.
S: Yeah. I'm gonna retype that one.
T: mm hmm, OK.
```

Figure 1: Tutorial dialogue from VanLehn's physics corpus

```
utt86 : u:  I'll have a tanker in <sil> Corning
utt87 : s:  mm-hm
utt88 : u:  I drive the tanker <sil> from Corning to Elmira
utt89 : s:  you can't drive the tanker without an engine
utt90 : u:  a tanker
utt91 : s:  the + + tanker + needs an <sil> engine +
utt92 : u:  + u- +
utt93 :     + I- I thought a tanker + was <sil> jus(t)- just <sil> truck <sil>
            or is it a
utt94 : s:  it's one of those round things that gets pulled by an engine
utt95 : u:  oh with it's part of the train oh I see <sil> okay <sil>
            and then that that's why I was confused
utt96 :     okay <sil> so we'll start <sil> in
utt97 : s:  + start in <sil> Elmira +
```

Figure 2: Collaborative task planning dialogue from the Trains '93 corpus.

where the system must unmark "known" concepts if they were not actually delivered).

T: Either treating them as a composite object, which is actually easier, um,

- Other disfluencies relate to the physical environment of the utterance. If participants are interacting with a held, nonshareable resource, for instance, such as a keyboard, mouse, or map, a meta-dialogue may occur that concerns these physical objects. In the case of computer-human conversation, these objects will probably be on-screen rather than physical. In all of these cases, dialogue must be continually replanned as the system notes that the user has taken possession of or released these resources.

  T: Well... let's get... Here, let's hit return in equation mode and then get out of equation mode. And I'll show you how if...

  T: That's excellent. whoops.

The majority of these examples show that it is sufficient to drop the current utterance as it is delivered to the speech synthesizer and replan new phrases from scratch as necessary. Moreover, much more detailed timing information is needed from speech synthesizers, for example in cases where the system wants to indicate that an utterance is cancelled by cutting out in the middle of a word. The most interesting case from the generation perspective is probably that of indicating a willingness to give up the initiative.

4 Possible Approaches to Generating Disfluencies

Turning from instances in which the context of the conversation demands that disfluencies occur, we now examine disfluency generation strategies. Of the standard generation pipeline architecture, two modules stand out as needing modification: the text planner and the surface realizer. The former must be prepared to both replan from scratch as well as indicate potential semantic break points, while the later must be able to actually generate disfluencies (with speech markup rather than text markup) and indicate potential syntactic break points. It is difficult to imagine that other modules, such as pronominalization, revision, or lexical choice would be altered significantly.

At a more fundamental level, there are several possible directions:

- **Canned and template generators:** These are NLG systems which store and manipulate simple, non-linguistic text strings [Reiter, 1995].
- **Deep generators:** Systems that consider all linguistic levels of representations, and can perform multiple types of operations on those levels.
- **Hybrid generators:** Systems that combine template and deep generation and can thus customize the level of linguistic representation needed for a particular application [White and Caldwell, 1998].
- **Statistical generators:** A newer approach to NLG is to use corpora as a guide to constructing sentences from se-

mantic components [Langkilde and Knight, 1998]. Because they are newer and thus less studied, it is still unclear what advantages and disadvantages they may bring to dialogue generation.

- **Incremental generators:** Deep generation systems which are capable of returning initial fragments of sentences immediately, before the entire sentence is processed [Kantrowitz and Bates, 1992; Kilger and Finkler, 1995].

These types of systems differ with respect to how they handle the disfluency phenomena described above. For example, template generators are only capable of sending stop commands to the speech synthesizer, as they have no knowledge of what break points in the sentence would be semantically or syntactically meaningful (other than guessing at punctuation marks like commas). Meanwhile, a non-incremental generator might have problems because when a stop order arrives from the dialogue manager, it may have been processing the final part of a sentence instead of the first part, and thus not have any meaningful output to interrupt.

For dialogue systems that need to generate all of the above categories of disfluencies, it is necessary to have a model, statistical or syntactic, that describes accurately how humans produce disfluencies. Thus canned and template systems would have a much (though not impossible) more difficult time exhibiting them, and hybrid systems would have the same problems when generating dialogue with the template components.

For more deeply generated NLG, there is currently a tradeoff between incremental generation and standard deep generation. While incremental generators are likely to be more adept at producing disfluencies, the grammars available are much less advanced than for other deep generators. But on the other hand, there is no research to show how difficult it would be to modify existing, large-scale deep generation systems to be more incremental in behavior. Additionally, there have been no empirical studies to show how well incremental generation performs with dialogue systems.

On a final note, there are several freely available large-coverage deep generation systems available for anonymous download for research purposes. This has helped quickly advance the state of the art in NLG. However, there are no available dialogue systems for download. The creation of such shareable resources is a vital component to steady advancement in the field.

5 Conclusion

Most Natural Language Generation systems focus on the production of perfectly written text for visual presentation. But as NLG tries to move into other areas of language production, such as speech in dialogue systems, it is not clear that approaches for traditional applications are best. Speech exhibits particular features like disfluency that are not found in written discourse. In this paper, we have presented an initial look at what may be necessary to allow current NLG systems to seamlessly make the transition to spoken language production, including a survey of linguistic work in disfluency, annotated examples of disfluency and how NLG systems might interact with them, and how current NLG systems might be modified to produce disfluencies.

References

[Allen, 1993] James Allen. The trains '93 corpus. Technical report, University of Rochester, http://www.cs.rochester.edu/research/trains, 1993.

[Allen *et al.*, 1996] James F. Allen, Bradford W. Miller, Eric K. Ringger, and Teresa Sikorski. Robust understanding in a dialogue system. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pages 62–70, June 1996.

[Brennan, 2000] Susan E. Brennan. Processes that shape conversation and their implications for computational linguistics. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, Hong Kong, October 2000.

[Callaway, 2001] Charles Callaway. A computational feature analysis for multilingual character-to-character dialogue. In *Proceedings of the Second International Conference on Intelligent Text Processing and Computational Linguistics*, pages 251–264, Mexico City, Mexico, 2001.

[Callaway and Lester, 2002] Charles B. Callaway and James C. Lester. Narrative prose generation. *Artificial Intelligence*, 139(2):213–252, August 2002.

[Carletta *et al.*, 1995] Jean Carletta, Richard Caley, and Stephen Isard. A system architecture for simulating time-constrained language production. *Language and Cognitive Processes*, 10(3–4):357–361, 1995.

[Cassell, 2001] Justine Cassell. Embodied conversational agents: Representation and intelligence in user interface. *AI Magazine*, 22(3):67–83, 2001.

[Cassell *et al.*, 2000] J. Cassell, M. Stone, and H. Yan. Coordination and context-dependence in the generation of embodied conversation. In *International Natural Language Generation Conference*, Mitzpe Ramon, Israel, 2000.

[Eugenio *et al.*, 2002] Barbara Di Eugenio, Michael Glass, and Michael Trolio. Simple natural language generation and intelligent tutoring systems. In *Second International Natural Language Generation Conference*, pages 120–127, Harriman, NY, July 2002.

[Freedman, 1996] Reva K. Freedman. *Interaction of Discourse Planning, Instructional Planning and Dialogue Management in an Interactive Tutoring System*. PhD thesis, Northwestern University, 1996.

[Heeman and Allen, 1999] Peter Heeman and James Allen. Speech repairs, intonational phrases and discourse markers: Modeling speakers' utterances in spoken dialogue. *Computational Linguistics*, 25(4):527–571, 1999.

[Jokinen and Wilcock, 1999] Kristiina Jokinen and Graham Wilcock. Confidence-based adaptivity in response generation for a spoken dialogue system. In *Proceedings of the 2nd SIGdial Workshop on Discourse and Dialogue*, pages 80–89, Aalborg, Denmark, 1999.

[Kantrowitz and Bates, 1992] M. Kantrowitz and J. Bates. Integrated natural language generation systems. In R. Dale, E. Hovy, D. Rosner, and O. Stock, editors, *Aspects of Automated Natural Language Generation*, pages 247–262. Springer-Verlag, Berlin, 1992.

[Kilger and Finkler, 1995] Anne Kilger and Wolfgang Finkler. Incremental generation for real-time applications. Technical Report RR-95-11, DFKI, Saarbruecken, Germany, July 1995.

[Langkilde and Knight, 1998] Irene Langkilde and Kevin Knight. Generation that exploits corpus-based statistical knowledge. In *COLING-ACL-98: Proceedings of the Joint 36th Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics*, pages 704–710, Montréal, Canada, August 1998.

[Lester *et al.*, 2000] James C. Lester, Stuart G. Towns, Charles B. Callaway, Jennifer L. Voerman, and Patrick J. FitzGerald. Deictic and emotive communication in animated pedagogical agents. In J. Cassell, S. Prevost, J. Sullivan, and E. Churchill, editors, *Embodied Conversational Agents*, pages 123–154. MIT Press, Cambridge, 2000.

[McKelvie, 1998] David McKelvie. The syntax of disfluency in spontaneous spoken language. Technical Report HCRC/RP-95, Edinburgh University, Edinburgh, Scotland, May 1998.

[McRoy *et al.*, 2000] Susan McRoy, Songsak Channarukul, and Syed Ali. Text realization for dialog. In *Workshop Notes of the AAAI '00 Fall Symposium on Building Dialogue Systems for Tutorial Applications*, Cape Cod, MA, November 2000.

[Oh and Rudnicky, 2000] Alice Oh and Alexander Rudnicky. Stochastic language generation for spoken dialogue systems. In *Proceedings of the ANLP/NAACL 2000 Workshop on Conversational Systems*, pages 27–32, Seattle, WA, April 2000.

[Oviatt, 1995] Sharon Oviatt. Predicting spoken disfluencies during human-computer interaction. *Computer Speech and Language*, 9(1):19–35, 1995.

[Person and Mathews, 2000] Graesser A.C. Harter D. Person, N. and E. Mathews. Dialog move generation and conversation management in autotutor. In *Workshop Notes of the AAAI '00 Fall Symposium on Building Dialogue Systems for Tutorial Applications*, pages 45–51, Cape Cod, MA, November 2000.

[Rambow *et al.*, 2001] Owen Rambow, Srinivas Bangalore, and Marilyn A. Walker. Natural language generation in dialog systems. In *Proceedings of the First International Conference on Human Language Technology*, San Diego, CA, 2001.

[Reiter, 1995] Ehud Reiter. NLG vs. templates. In *Proceedings of the Fifth European Workshop on Natural Language Generation*, Leiden, The Netherlands, 1995.

[Rich and Sidner, 1998] Charles Rich and Candace L. Sidner. COLLAGEN: A collaboration manager for software interface agents. *User Modeling and User-Adapted Interaction*, 1998. In press.

[Smith and Hipp, 1994] Ronnie W. Smith and D. Richard Hipp. *Spoken Natural Language Dialog Systems*. Oxford University Press, Cambridge, Massachusetts, 1994.

[Traum, 1994] David Traum. *A Computational Theory of Grounding in Natural Language Generation*. PhD thesis, University of Rochester, 1994.

[Traum and Hinkelman, 1992] D. Traum and E. Hinkelman. Conversation acts in task-oriented spoken dialog. *Computational Intelligence*, 8(3):575–599, 1992.

[VanLehn, 1998] K. VanLehn. Physics Corpus 1. Technical report, University of Pittsburgh, http://www.pitt.edu/ circle/Archive/Physics1.htm, 1998.

[Walker *et al.*, 1997] Marilyn A. Walker, Janet E. Cahn, and Stephen J. Whittaker. Improvising linguistic style: Social and affective bases of agent personality. In *Proceedings of the First International Conference on Autonomous Agents*, 1997.

[Ward, 1996] Nigel Ward. Using prosodic clues to decide when to produce back-channel utterances. In *Proceedings of the 4th International Conference on Spoken Language Processing*, Philadelphia, PA, 1996.

[White and Caldwell, 1998] Michael White and Ted Caldwell. EXEMPLARS: A practical, extensible framework for dynamic text generation. In *Proceedings of the Ninth International Workshop on Natural Language Generation*, pages 266–275, Niagara-on-the-Lake, Ontario, August 1998.

[Zechner, 2001] Klaus Zechner. Automatic generation of concise summaries of spoken dialogues in unrestricted domains. In *Proceedings of the 24th ACM-SIGIR Information Retrieval Conference*, New Orleans, LA, September 2001.