Making Sense of Silence in Speech User Interfaces

Gina-Anne Levow

MIT AI Laboratory
gina@ai.mit.edu

CHI '97 Workshop: Speech User Interface Design Challenges

Short Biography

I have worked on a wide variety of spoken language interfaces. I developed the discourse manager for the EUCALYPTUS interface to the KOALAS aircraft carrier simulation system, which allowed speech, text, and mouse inputs integrated to allow for more natural use of the complex interface involving an AWACS style display. This project was carried out at the NRL NCARAI in Washington, D.C. I also worked on the SpeechActs project at Sun Microsystems laboratories, building grammars, modifying the speech synthesizer, and building applications for speech-in/speech-out access to calendar, mail, and weather applications over the telephone. I have also worked with Kurzweil Applied Intelligence to design speech systems to voice-enable both text and graphical editors. I am currently collaborating on a project with Sharon Oviatt at OGI to precisely model hyperarticulation by users of a speech user interface in the course of error resolution. My Ph.D. thesis work underway at MIT is on the use of intonation in human-computer dialog to produce more natural interaction and more comprehensible computer speech.

The Silence Problem

In a recent study of a voice-in/voice-out user interface, one the most common laments was the perceived slowness of the interaction. One of the system characteristics which led to this sensation was the presence of lengthy silences, particularly after the user felt she had completed her turn and was waiting for the system to respond. This problem also occurred after system completed its output.

Typical human-human inter-turn latencies in telephone conversation exceed one second only five percent of the time, as noted by Robert Hopper in Telephone Conversation. When this time limit is exceeded, one party or another acts to get the conversation moving again, even if this means taking back a relinquished turn. In contrast, many speech recognition systems wait until a one second silence has elapsed since the last detected user input before even beginning the recognition and analysis process. Given slower than real-time recognition and possible delays in application processing and response generation, substantial waits are possible, during which the system maintains a perplexing silence. Finally, after the the system does respond it again waits silently for user input.

Ambiguous Silences: What They Can Mean

Not only are these delays irritating in and of themselves, but they also make it particularly difficult for the user to form a coherent model of the system's state, with direct adverse effects on system usability.

As outlined above, there are several states that can produce this ambiguous silence:

- start of input detected, end not yet detected
- utterance endpointed, recognition incomplete
- recognition completed, language analysis incomplete
- recognition and analysis successful, application processing incomplete
- pause in incomplete system response
- system response completed, waiting for user input, and
- system crashed.

Given this wide variety of causes, the user's frustration and confusion in the face of long system silence is quite understandable.

Pauses and Turn Transitions

These states, however, can be viewed as falling into two major classes based on the turn transition which is taking place. The first four states are related to user confusion as they attempt to pass control TO the system; the remainder of the non-failure states are tied to problems with the user's ability to identify, or more specifically the system's ability to convey, when the transition to user control takes place.

Considered in this light, it is not surprising that these difficulties are most evident when there is a change in the initiative structure of the dialog, for instance, when control is returned to the user after a rigidly computer-driven clarification subdialog. Many people are initially surprised by the finding that pauses in telephone conversation are generally substantially shorter than those in face-to-face conversation. They believe that the presence of a wide variety of cues, such as glance, posture and facial expression, would provide clear evidence for the interlocutor's state and thus speed the transfer of turns. However, it is because of the presence of numerous cues to one's partner's continued attention that relatively long pauses are tolerated in face-to-face conversation, whereas in the restricted medium of telephone conversation verbal/acoustic feedback is the only channel available.

Spoken language interfaces to computers, even when some visual display is present, often have more in common with the dynamics of telephone conversation, since no current system is able to imitate or even approximate the myriad cues that a face-to-face conversant provides. As a result, as in human-human telephone conversations, lengthy ambiguous silences create significant problems for conversational flow, and ameliorating this confusion is an important challenge for speech interface designers.

Turn Transition: User To System

Perhaps the most problematic case arises when the user has spoken and then, because of a seemingly excessive delay, repeats their request to the system. A study of human-human telephone conversations found 95% of inter-turn pauses are less than one second in duration and inter-turn pauses average less than 200 milliseconds. In contrast, an analysis of a specific SUI system [4] indicated that even a very experienced user could expect average delays of 2.6 seconds between the input and start of response, with some latencies exceeding 6 seconds. Here the user believes that the system did not hear or understand their original input. While this may indeed be the case, it is also often the case that the system has heard the utterance, but

a) has not yet endpointed it,

b) has not yet completed recognition, or
c) has not yet produced the response.

Each of these circumstances poses a different problem. In the first instance, the result is often a recognition error, typically a rejection, since the doubled utterance is unlikely to constitute a grammatical input to the speech recognizer. Conditions b and c produce essentially the same outcome. Since the system is silent, it is likely to be accepting audio input. If the system does not support barge-in, the repetition will be interpreted as a second, separate input to the system. This is particularly disconcerting since it leads to two successive system responses, often identical, in response to what the user believes to be a single input. In a barge-in equipped system, it is possible to repeatedly interrupt system processing so that no response is ever provided, as the user interrupts processing again and again.

Turn Transitions: System to User

Another cluster of difficulties surrounds the user's determining when the system has finished speaking. During a response, the system may pause (briefly, one hopes) between items in lists, between paragraphs in long texts, between stages in a response, or even permanently due to a system failure. All of the breaks occur in addition to the expected halt at the close of the completed response. Given the state of the art in speech synthesis, there is little information to aid the user in determining which condition holds, other than the length of the pause. Two problematic circumstances may arise: when the user erroneously expects the system to continue and when the user mistakenly believes that the system has finished. The first essentially sets up a stalemate where the user waits for the system, but the system is actually waiting for the user. The user often becomes frustrated and confused. Sometimes they tentatively attempt input, ask "hello?", or even try to use telephone keypad input to get the system to escape from this unresponsive state. These attempts often lead to misrecognitions and further frustration. In a system without barge-in this can result in another input being added to the list to be processed, or the second input may be ignored altogether. If the system is equipped with barge-in, the problems may be even more severe, since the user may remain unaware that there was more information yet to come and may miss something important. Clearly understanding what a silence means is vital to determining whose turn it is, and providing smooth interaction.

Misinterpreting Silence: What Goes Wrong

Information Loss

An instance of misunderstood silence resulting in repeated user interrupts is demonstrated in the following scenario. The system is an over-the-telephone voice and keypad interface with a barge-in facility. The system reads several long menus of more than six choices which must be heard fully in order to identify all of the possible commands and invocations. A new user listens to the menu and the barge-in system incorrectly identifies an unintentional interruption (such as a loud breath by the user) but, naturally, fails to recognize the utterance and reports an error. The user at this point does not know whether or not any additional choices remain. The user, not knowing that more options exist but only those that were heard, concludes that the desired facility is absent. This situation serves to illustrate the importance of conveying to the user what state the system is in when it is producing output. It is also an example where the technical problem - over-sensitive barge-in - may be extremely difficult to solve in general, but a better interface design could save the user from being misled.

Spoke-Too-Soon

A particularly irritating problem is the "spoke-too-soon" problem. Here the user believes that the system has finished and is ready for input. The difficulty is that completion of response is not necessarily equivalent to readiness for input - particularly in systems with prompt tones. In these cases, a recognition error frequently occurs, when the user speaks after the silence, but before the prompt tone. Even experienced users are subject to this mistake, since it imposes an additional, unnatural step in the interaction.

The Challenge

Unfortunately, not all these problems can be solved simply by technical means in improving recognition or analysis speed. Delays due to information retrieval and output pauses are inevitable, and recognition delays are likely to remain a part of interactions with speech systems for some time. One also does not want to replace confusing, lengthy silences with irritating, time-consuming prompts from the system. So, the challenge remains to find an informative, yet unobtrusive way of disambiguating silences in human-computer spoken dialog systems.

References

Hopper, R. (1992) Telephone Conversation. University of Indiana Press.

Thomason, W.R. (1990) The relationship between transition relevance and speaker change in two conversational pause environments. Paper presented at annual meeting of Speech Communication Association, November, Chicago.

Thomason, W. R. and Hopper, R. (1992) Pauses, transition relevance and speaker change. Human Communication Research, 18, 429-44.

Yankelovich, N., Levow, G-A., and Marx, M. (1995) Designing SpeechActs: Issues is Speech User Interfaces. CHI ’95 Conference on Human Factors in Computing Systems, Denver, CO, May 7-11, 1995, SMLI94-0394.
