INTELLIGENT BARGE-IN IN CONVERSATIONAL SYSTEMS

N. Ström & S. Seneff

MIT Laboratory for Computer Science [1]
200 Technology Square
Cambridge, Massachusetts 02139 USA
{nikko, seneff}@sls.lcs.mit.edu

ABSTRACT

In this paper we present novel solutions to problems related to barge-in in telephony-based conversational systems. In particular we address recovery from falsely detected barge-in events and a method for signaling to the user that barge-in is disallowed at a particular dialogue state. The mechanisms and signals used to manage turn taking are similar to those in human-human conversation, which makes them easy to understand for users without explanation or prior training.

1. INTRODUCTION

In telephony-based spoken language systems, it is desirable to let users interrupt system output at any time, in particular if the output is based on erroneous understanding or contain superfluous information. Thus, enabling barge-in, i.e., the ability for the user to start speaking before system output has ended, can significantly enhance the user experience. However, users' new freedom also poses new challenges. One challenge is sorting out true user barge-in from background noise and non-speech sounds like coughs, and in telephony-based systems it is non-trivial to separate the user's voice from system output (echo cancellation). Updating the discourse history appropriately is also significantly harder after a barge-in because the user has heard only part of the system output. Furthermore, there may be dialogue states where it is desirable to prohibit barge-in (e.g., commercial advertisements, terms-of-usage messages, disclaimers, etc.) and this condition must be gracefully made apparent to the user.

This paper describes novel solutions to these challenges and their implementation in our mixed initiative conversational systems. The methods presented, based on features of human-human prosody and discourse, are domain-independent in nature and have been applied to several domains (weather, flight booking, and traffic) [1][2]. We approach barge-in in three phases, detection, verification, and recovery. Detection is based on short-term acoustic measurements of energy and voicing. Verification relies on the confidence score from the speech recognizer and the natural language component to classify the input as an intended user input or not. Finally, if verification fails the system must recover gracefully

2. DETECTION

A fast and reliable speech detection algorithm is important in any conversational system, but in particular in conjunction with barge-in. Fast responses to user barge-in are more important than to regular, non-barge-in utterances, because a real-time decision has to be made whether to turn off the system output. Accuracy is important because the necessary dialogue repair after a false barge-in detection is potentially more complex than that of a regular recognition error (recovery strategies are discussed in section 4).

The detection algorithm used in this study is based on signal energy and periodicity, where periodicity is defined as the autocorrelation coefficient corresponding to the fundamental frequency, normalized by dividing by the zeroth coefficient. The fundamental frequency is estimated from the autocorrelation analysis as well, but it is not explicitly used as a feature for detection.

A 50 ms frame rate is used, and a frame is marked as speech if both energy and periodicity exceed their respective thresholds. Most unvoiced and/or soft phones go undetected by these features, but vowels are relatively robustly detected. To capture also the consonants, 200 ms are added at the beginning and end of detected utterances, and 900 ms bridges are allowed between marked frames.

[1] This research was supported by DARPA under Contract N66001-99-1-8904 monitored through Naval Command, Control and Ocean Surveillance Center.

3. VERIFICATION

The verification phase starts when the end of a barge-in utterance has been detected. The purpose is to determine if the recorded audio is a valid user utterance, or merely background noise or a non-speech sound from the user, such as a cough. To determine this, a threshold is used on the recognizer confidence score [3] as well as the requirement that the natural language processing (NLP) component [4] must be able to generate an understanding of the utterance in the current context. For an utterance to be verified it must pass both tests, otherwise it will be rejected and treated as a falsely detected barge-in.

The rationale behind the NLP test is a cost-benefit one. The relative cost is low, because accepting a barge-in in this condition leads to an un-interpreted user utterance typically followed by a clarification sub-dialogue. The benefit is of course that all falsely detected barge-in in this condition are avoided.

4. RECOVERY

The last phase of the intelligent barge-in is entered when a barge-in is detected but later rejected in the verification phase.

4.1 Simplistic recovery schemes

The most straightforward strategy is to always stop system output as soon as a speech starting point is detected, and treat failed verifications in the same manner as regular user utterances that are not understood by the system. However, the subsequent dialogue repair can be challenging after a false barge-in. It is therefore advantageous to devise a scheme to backtrack to the interrupted system turn, in order to arrive at a well-defined dialogue state.

By simply postponing the decision to stop system output until after verification, the problem with rejected detections can be avoided — if the barge-in is verified, output is stopped at that time, otherwise the detection is simply ignored. However, this has serious disadvantages. First, it is uncomfortable for the user to have to compete with the system output for more than a very short period of time, and it may make her uncertain whether barge-in is allowed. Second, in a telephony-based system, the audio quality is typically degraded significantly in this cross-talk condition, which reduces recognition accuracy.

4.2 Strategy I: reduce and listen

An improved scheme is to reduce the volume of the system output when a barge-in is detected, rather than fully stopping the output. In case verification is later successful the system output is terminated, otherwise the volume is reset to normal (see Figure 1). This technique has several advantages**: i)** the reduced loudness is a signal to the user that a barge-in is detected, **ii)** a falsely detected barge-in does not stop the output, and **iii)** the reduction in the volume of the system output may improve the audio quality of the recorded user utterance in telephony-based systems, because the effect of cross-talk is lessened.

Even the reduced system output does degrade the audio quality of the user utterance, so to avoid long cross-talk segments the system output is turned off fully after 1000 ms (see Figure 1).

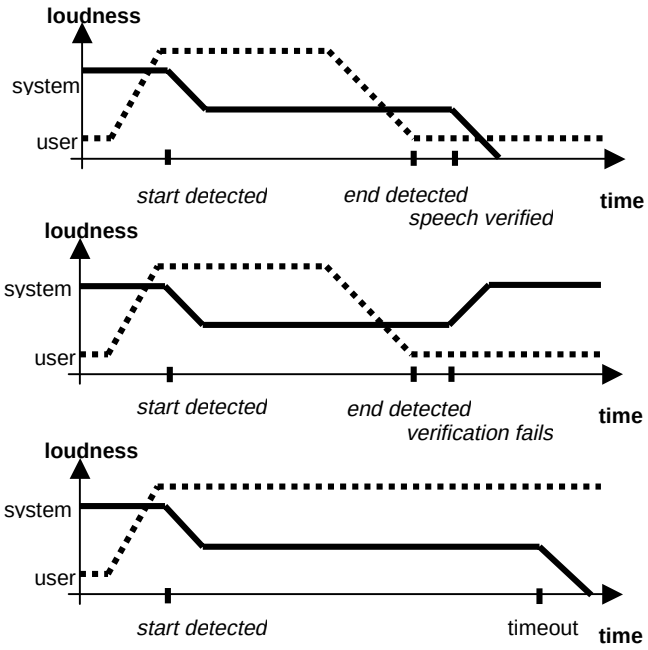

Figure 1. Barge-in recovery. Strategy I. *Top:* System output is reduced at detection time and turned off when the barge-in is verified. *Middle:* System output is reduced at detection time but later restored when verification fails. *Bottom:* System output is reduced at detection time and turned off after a timeout (1000 ms).

4.3 Strategy II: rewind and resume

Strategy I is a technically feasible method and it addresses the problems associated with the simplistic schemes. However, in informal user tests, the behavior was perceived as rather artificial. In response to this, we have devised a novel method that has the same advantages and is in addition more human-like. This method stops output immediately when a barge-in is detected. If the verification later rejects the input, a disfluency marker, <um>, is played, and system output is resumed from the last phrase boundary, with the appearance of a

momentary pause or hesitation (see Figure 2). Users do not expect synthetic voices to utilize this type of prosodic mechanisms to manage turn taking, and some find it quite amusing, but users do understand the function of the signal without explanation because it is sufficiently close to mechanisms used in human-human conversation. Timing is very important for naturalness in conjunction with this strategy. As a rule of thumb, system output should be resumed within a few syllable beats after the end of the user utterance, or else it may be better to fall back to a different recovery strategy.

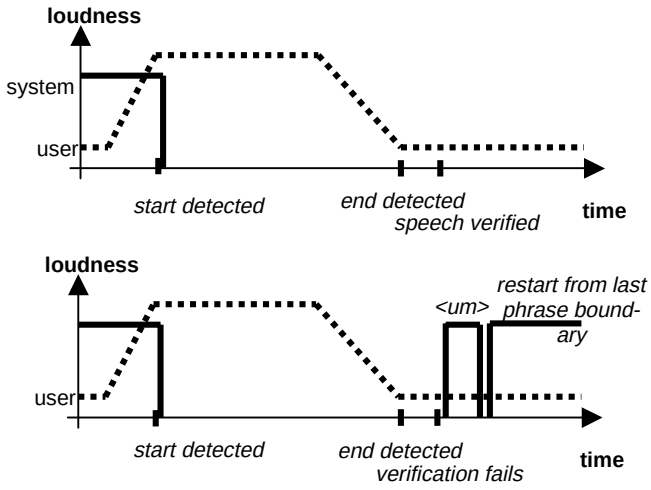

Figure 2. Barge-in recovery. Strategy II. *Top:* System output is turned off when a barge-in is detected. When the barge-in is verified, the remaining system output is discarded. *Bottom:* System output is turned off when a barge-in is detected but later resumed following a disfluency marker when the barge-in fails to be verified.

5. DECLINING BARGE-IN

There may be dialogue states where it is desirable to prohibit barge-in. In a commercial system, one important class of such states may be advertisements. Other examples are terms-of-usage messages, disclaimers, etc. To avoid confusing the user, this condition must be gracefully made apparent.

Our solution is, like the solution for false barge-in detection recovery, inspired by human-human dialogue. The system's response to a user barge-in attempt in a dialogue state where barge-in is not allowed is to increase the loudness of the system output for a short period of time to "keep the floor" (see Figure 3). This is an intuitive signal that most users understand without explanation. To strengthen the impression of increased vocal effort, we also apply pre-emphasis to the output signal to boost high frequencies of the speech. In a telephony system with limited signal amplitude and dynamic range, this has the additional advantage of boosting output energy without increasing signal amplitude.

For telephony-based systems we use the pre-emphasis filter $s'_i = s_i - 0.25\ s_{i-1}$, and an amplitude gain factor of 3.0, i.e, the total transformation is: $s'_i = 3.0s_i - 0.75\ s_{i-1}$. However, these parameter values are dependent on the spectral properties of the particular system output.

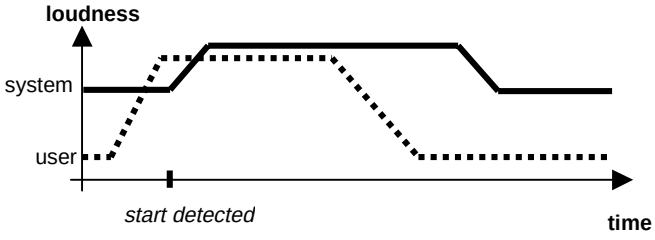

Figure 3. Declining barge-in. The loudness of the system output is increased for a short period of time (1500 ms) to indicate that barge-in is not allowed in this dialogue state.

6. SYSTEM BARGE-IN

In a mixed initiative conversational system it is not inconceivable to allow the system to barge in under some circumstances. An example could be an out of domain query:

U: *How much are the tickets to the next…*
S: *I'm sorry, I don't have ticket information, I know only of recent sports results.*

However, implementing the general case of system barge-in requires accurate, continuously updated, real-time understanding of partial utterances, which is currently beyond our capabilities. Nevertheless, we use system barge-in in one particular situation. Very long user utterances are often misinterpreted by the system. Therefore, after a state dependent amount of time, a timeout occurs, causing the system to interpret the partial user utterance recorded so far. At this point the system may barge-in on the user who may still be speaking at the onset of the system response to the truncated utterance. This has the advantage that the user is made aware that the final part of her utterance is not processed by the system. In a symmetric model, we would give the user the opportunity to decline the system's barge-in by simply continue speaking and perhaps raise her voice. This is an attractive feature, in particular when the type of barge-in of the example above can be handled, but it has not yet been implemented.

7. DISCOURCE MANGEMENT

We have seen (in section 3) an example of how NLP can be coupled with intelligent barge-in handling. The coupling becomes even more evident when we consider how to properly update discourse context when barge-in occurs. Users may interrupt a system to correct erroneous understanding, change the topic, or make a choice from options presented in the current prompt. Discourse context for the dialogue would be updated in different ways dependent on the timing of the barge-in and the intention of the utterance. To get an understanding of the complexity of the impact on discourse management, consider the case where a user interrupts a list of choices. It may seem sensible to update the discourse context based only on what the user had heard until the barge in occurred. For example:

U: *What places do you know about in China?*
S: *I know of the following places in China, please select one: Beijing, Guangzhou, Harbin…*
U: *Guangzhou.*

At this point it would be highly unlikely for the user to say for example "Shanghai". However, although updating based only on what the user heard is a reasonable first approximation; it is not a rule without exceptions. In particular, the redundancy of language often makes unheard constituents highly predictable. Consider:

U: *What places do you know about in Europe?*
S: *I know of the following places in Europe, please select one: Northern Europe, Southern Europe, Eastern Europe…*
U: *Western Europe.*

Here, because of the particular context, a choice that is not yet heard makes perfect sense. The latter type of interaction is very common in system-directed dialogue systems, in particular with experienced users who already know the list of choices. A refined rule would update the discourse context based on *what the user heard plus an estimate of what the user may have inferred*. Clearly this makes the process significantly more challenging and perhaps it suggests handling discourse management in a stochastic framework. This problem will be explored further in the future.

8. CONCLUSION

We have implemented a novel set of signals and behavioral patterns for mixed initiative conversational systems to handle barge-in. The model has been applied to several domains: the Jupiter weather information system [1], the Mercury flight booking system [2], and the Voyager traffic information system.

Because signals and behavioral patterns are inspired by human-human dialogue, the system is intuitive to use, and users typically understand the function of any particular system behavior without explanation or prior training. Prosodic features, like changing the amplitude of system output, are used when possible to signal the state of the turn taking model to the user. This information is perceived in parallel with the literal content of the system output and is therefore a very efficient type of communication.

It is clear that natural language processing can play an important role in generating appropriate responses to barge-in, and for determining when a system barge-in is appropriate. We view the limited role played by such techniques in this study as merely a starting point in the development of a comprehensive model for turn taking in conversational systems.

The methods of this paper cover only a very small fraction of the rich system of signals and behaviors that governs turn taking in human-human dialogue. We appreciate the importance of intonation, pausing, back-channeling, and even such complex behavior as finishing each other's sentences. Nevertheless, we consider this study a first step towards a model for barge-in and turn taking, where ultimately the same intuitive rules, familiar from human-human dialogue, apply for the user as well as the system in the turn taking game.

REFERENCES

[1] V. Zue, S. Seneff, J. Glass, J. Polifroni, C. Pao, T. J. Hazen, and L. Hetherington, "JUPITER: A Telephone-Based Conversational Interface for Weather Information," IEEE *Transactions on Speech and Audio Processing*, **Vol. 8**, No. 1, January 2000.

[2] S. Seneff and J. Polifroni, "Dialogue Management in th Mercury Flight Reservation System," *Proc. ANLP/NAACL 2000 Workshop on Conversational Systems*, Seattle, May 2000.

[3] T. J. Hazen, T. Burianek, J. Polifroni and S. Seneff, "Recognition Confidence Scoring for Use in Speech Understanding Systems," *To appear in The ISCA ASR2000 Tutorial and Research Workshop.*

[4] S. Seneff, "The Use of Linguistic Hierarchies in Speech Understanding," *Proc. ICSLP 98, Sydney, Australia*, November 1998.