INVESTIGATION OF CONFIRMATION STRATEGIES FOR SPEECH RECOGNITION APPLICATIONS

Cheryl Kotan
IBM Corp.
Boca Raton, FL

James R. Lewis
IBM Corp.
Boca Raton

Guidelines for speech user interfaces generally promote the use of delayed confirmation in speech recognition applications that require users to provide multiple elements of information. In our initial investigation of a simple delayed confirmation method, we discovered a significant design flaw (requiring users to review a fairly large amount of correct input multiple times when asked to correct two errors). To avoid this flaw, we designed two new methods. In one (Serial Collection/Correction), users named an item that needed correction, then made that correction before naming the next item to change. In the other (Batch Collection/Correction), users named all items that needed correction first, then changed the named items in sequence. An experiment comparing the methods indicated no significant difference in user preference, but significantly fewer memory errors occurred when using the Serial Collection/Correction method. Thus, we recommend the use of the Serial Collection/Correction method. We also recommend including an option to replay the information if a user has remembered the need to change an additional item, but does not remember which item to change.

INTRODUCTION

Guidelines for speech user interfaces (Balentine and Morgan, 2002; IBM, 2001) generally promote the use of delayed confirmation in speech recognition interactive voice response (IVR) applications that require users to provide multiple elements of information. This paper describes two experiments designed to investigate confirmation strategies for such speech recognition applications.

In the first experiment, we investigated a simple delayed confirmation method. This investigation revealed a significant design flaw, prompting the development of two alternative methods for delayed confirmation. In the second experiment, we compared these new alternatives. In one method (Serial Collection/Correction), the system directed users to name an item that needed correction, and then to make that correction before naming the next item to change. In the other (Batch Collection/Correction), the system directed users first to name all items that needed correction, and then to change the named items in sequence.

EXPERIMENT 1: IMMEDIATE VERSUS DELAYED CONFIRMATION

In this experiment we investigated two basic confirmation strategies for data input with a speech recognition interactive voice response (IVR) system: immediate and delayed. Figures 1-4 illustrate the confirmation styles. These examples show that the advantage of delayed over immediate confirmation is evident only when the user makes no corrections, at least for this simple delayed confirmation strategy.

Figure 1. Immediate Confirmation with No Errors (Sys = System; Hum = Human)

Sys: *Date?*
Hum: May 3rd.
Sys: *Was that May 3rd?*
Hum: Yes.
Sys: *Time?*
Hum: 2:00 PM.
Sys: *Was that 2:00 PM?*
Hum: Yes.

Figure 2. Delayed Confirmation with No Errors (Sys = System; Hum = Human)

Sys: *Date?*
Hum: May 3rd.
Sys: *Time?*
Hum: 2:00 PM.
Sys: *Let's review. Was that May 3rd at 2:00 PM?*
Hum: Yes.

Figure 3. Immediate Confirmation with Two Errors (Sys = System; Hum = Human)

Sys: *Date?*
Hum: May 3rd.
Sys: *Was that May 1st?*
Hum: No.
Sys: *Date?*
Hum: May 3rd.
Sys: *Was that May 3rd?*
Hum: Yes.
Sys: *Time?*
Hum: 2:00 PM.
Sys: *Was that 3:00 PM?*
Hum: No.
Sys: *Time?*
Hum: 2:00 PM.
Sys: *Was that 2:00 PM?*
Hum: Yes.

Figure 4. Delayed Confirmation with Two Errors (Sys = System; Hum = Human)

Sys: *Date?*
Hum: May 3rd.
Sys: *Time?*
Hum: 2:00 PM.
Sys: *Let's review. Was that May 1st at 2:00 AM?*
Hum: No.
Sys: *What needs to change? Select date or time.*
Hum: Date.
Sys: *New date?*
Hum: May 3rd.
Sys: *Let's review. Was that May 3rd at 2:00 AM?*
Hum: No.
Sys: *What needs to change? Select date or time.*
Hum: Time.
Sys: *New time?*
Hum: 2:00 PM.
Sys: *Let's review. Was that May 3rd at 2:00 PM?*
Hum: Yes.

Method

Three IBM employees participated in this study. The participants completed four tasks with a prototype bill-paying application. After each task, participants completed an After Scenario Questionnaire (ASQ – Lewis, 1995). The first two tasks required no corrections, and the second two tasks each required two corrections. Participants attempted both pairs of tasks first with the immediate confirmation strategy, then the delayed.

Experimental Design. The primary independent variable of interest in this study was the confirmation strategy (immediate or delayed). Due to the small scale of the study (only three participants), the less interesting independent variables (number of errors and payment task) were aliased, with the no-errors task bound to payment by electronic check and the two-error task bound to payment by credit card. Thus, the only reasonable inferential statistics were *t*-tests comparing the results within each of the two combinations of payment task and number of errors.

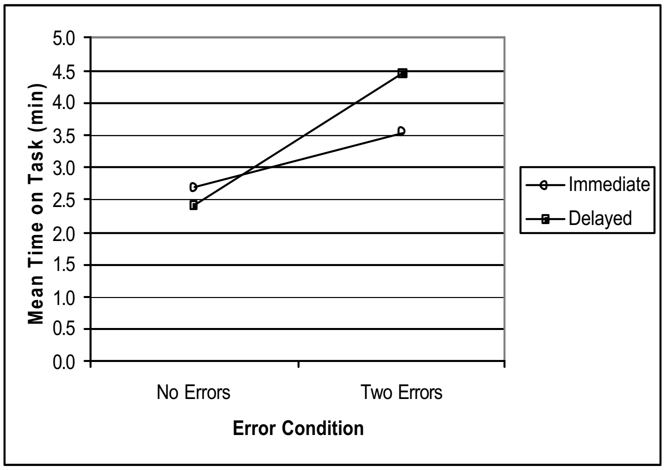
Results

Turns Required to Complete Tasks. For these analyses, a "turn" is a system prompt and the corresponding response from the human. Participants needed to complete 15 turns with the prototype application to complete Task 1 (immediate confirmation, no corrections). Task 2 (delayed confirmation, no corrections) required 11 turns. Task 3 (immediate confirmation, two corrections) required 24 turns. Task 4 (delayed confirmation, two corrections) required 23 turns.

Preference. Neither of the planned ASQ comparisons indicated statistical significance (no errors: $t(2) = 2.00$, $p = .18$; two errors: $t(2) = 1.00$, $p = .42$).

Time-on-Task. Figure 5 shows the times-on-task for the combinations of confirmation style and number of corrections. The only planned comparisons were within the two task/error combinations. For no corrections, the statistically significant difference in time ($t(2) = 16.00$, $p = .004$) was relatively small (about 15 seconds), but very consistent. The time difference for two corrections was also statistically significant ($t(2) = 9.04$, $p = .01$). When there were no corrections, delayed confirmation was faster than immediate confirmation, but when there were two corrections, immediate confirmation was the faster strategy by about a minute.

Figure 5. Mean Times-on-Task for Each Task



Discussion

These results indicated a potentially serious design flaw for a simple delayed confirmation strategy in the presence of two or more recognition errors because it required users to review a fairly large amount of correct input multiple times when asked to make two corrections, making this an unacceptable strategy. Because the use of immediate confirmation has its own well-known user acceptance issues (a feeling of slowness and excessive caution, see Balentine and Morgan, 2001), we decided to develop and test two new strategies for delayed confirmation, as described in Experiment 2.

EXPERIMENT 2: SERIAL VERSUS BATCH COLLECTION/ CORRECTION

Based on the results obtained in Experiment 1, we developed two new delayed confirmation strategies, illustrated in Figures 6 and 7 – Serial Collection/Correction and Batch Collection/ Correction. Both of these strategies, like the simple immediate confirmation strategy used in Experiment 1, require nine turns to complete in the given examples, but do not require users to listen repeatedly to the full confirmation message or to a complete menu of items to change for each correction, solving the major usability problem observed for the simple delayed confirmation strategy.

Figure 6. Example of Serial Collection/ Correction

Sys: *Date?*
Hum: May 3rd.
Sys: *Time?*
Hum: 2:00 PM.
Sys: *Let's review. Was that May 1st at 2:00 AM?*
Hum: No.
Sys: *What needs to change? Select date or time.*
Hum: Date.
Sys: *New date?*
Hum: May 3rd.
Sys: *Was that May 3rd?*
Hum: Yes.
Sys: *Anything else need to change?*
Hum: Time.
Sys: *New time?*
Hum: 2:00 PM.
Sys: *Was that 2:00 PM?*
Hum: Yes.

Figure 7. Example of Batch Collection/ Correction

Sys: *Date?*
Hum: May 3rd.
Sys: *Time?*
Hum: 2:00 PM.
Sys: *Let's review. Was that May 1st at 2:00 AM?*
Hum: No.
Sys: *What needs to change? Select date or time.*

Hum: Date.
Sys: We'll make that change in a moment.
Change anything else?
Hum: Yes, the time.
Sys: New date?
Hum: May 3rd.
Sys: Was that May 3rd?
Hum: Yes.
Sys: New time?
Hum: 2:00 PM.
Sys: Was that 2:00 PM?
Hum: Yes.

It wasn't clear which of the two delayed confirmation strategies would have the least demand on working memory during task completion. The Serial Collection/Correction design requires users to make the first correction before selecting the second item to change. Performing the first correction action could disrupt the maintenance of additional items to change in working memory. The Batch Collection/Correction design lets users select all items to change before beginning the correction step, but does not conform to the stereotypical correction steps that users seem to expect, which could have a disruptive effect on task performance.

The purpose of Experiment 2 was to conduct an experiment to investigate these two confirmation strategies.

Method

Participants. Eight users (five employees of a temporary employment agency and three IBM employees) participated in this study. Four participants were female (3 < 40 years of age; 1 > 40 years of age) and four were male (2 < 40 years of age; 2 > 40 years of age). All participants had previously used credit cards or checks to make online purchases, had given credit card or check information over the phone, and had called or used a speech recognition system at least once in the past.

Apparatus and Materials. Participants completed bill-paying tasks using two prototype voice applications. Both applications used the same introduction and data collection steps. After collecting the data, the system read back each item the user entered in a batch review. From that point on, the two programs were different, in accordance with the serial and batch collection/correction methods.

Procedure. Participants received instructions to use two different voice applications to pay a bill. They were to pay once with an electronic check and once with a credit card. The systems were programmed with two recognition errors. Participants made what they felt were the appropriate changes, then proceeded to process the transaction and exit the system. Across the eight participants, the experimental design counterbalanced the order of presentation for correction strategy (Serial vs. Batch), the order of presentation of the payment task (Check vs. Credit Card), and the pairing of correction strategy and payment task. After each task, participants completed an ASQ (Lewis, 1995).

Results

Preference. Two-tailed t-tests indicated no statistically significant difference (with the lowest value of p equal to .40) between the programs for any of the ASQ overall ratings or item ratings. 75% of participants indicated that Serial Collection/Correction was their favorite, with an adjusted-Wald 90% binomial confidence interval ranging from 40% to 93.7% (see Sauro and Lewis, 2005, for details about this new method for constructing binomial confidence intervals).

Successful Completions. All participants successfully completed the bill payment task using Serial Collection/Correction. Three of the eight participants failed to complete the payment task with Batch Collection/Correction. All three failures occurred when participants were trying to correct misrecognitions. A statistical test for differences in non-independent samples in 2 x 2 tables (Steel and

Torrie, 1980, p. 506-507) was marginally significant (*?* ^{2}(1) = 3, *p* = .08).

Participant Comments. Some participants specifically said they liked the Serial Collection/Correction style better, describing it as "good error recovery for mistaken data entry." Participants also said they liked having the ability to review the information they entered and make any necessary changes. A few suggested that they would like to be able to request the review at any point to prevent losing track of their changes. Many participants reported liking check payment better because they didn't have to enter as many items and the account numbers weren't as long.

RECOMMENDATIONS AND DISCUSSION

Recommendation 1: Use the Serial Collection/Correction method for delayed confirmation of multiple items of information in speech recognition IVRs.

Rationale: Despite our initial belief that the Batch Collection/Correction method would have lower memory demand on users, we found that significantly fewer task failures occurred when using the Serial Collection/Correction. There were no differences in the ASQ ratings for the two methods, but when asked directly, most participants preferred the Serial method.

Recommendation 2: Provide an option to replay the information if a user has remembered the need to change an additional item, but does not remember which item to change.

Rationale: Some participants indicated that it would be useful to be able to request a review of the entered data at any time during the process of correction. For example, it would be possible to include this option as the final option when listing the items available for change in the confirmation procedure.

Recommendation 3: If an application requires a final review of all data before accepting the data for processing, then do not engage in immediate confirmation during changes. If there is no need for a final comprehensive review of entered data, then use immediate confirmation of changes.

Rationale: In this study, participants using the new confirmation strategies engaged in immediate confirmation of items after making a change, and never had to review the entire set of entered data. This might be appropriate in some applications, but in other applications (especially financial) it might be necessary to have users go through a final review before accepting the data for processing.

Note: These results are specific to speech applications (not applicable to graphical user interfaces, which do not have the same memory demands on users).

REFERENCES

Balentine, B., and Morgan, D. M. (2001). *How to build a speech recognition application: A style guide for telephony dialogs* (2nd ed.). San Ramon, CA: Enterprise Integration Group.

International Business Machines, Corp. (2001). *IBM VoiceXML Programmer's Guide*. Author.

Lewis, J. R. (1995). IBM computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, *7*, 57-78.

Sauro, J., and Lewis, J. R. (2005). Estimating completion rates from small samples using binomial confidence intervals: Comparis ons and recommendations. In *Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting* (pp. 2100-2104). Santa Monica, CA: Human Factors and Ergonomics Society.

Steel, R. G. D., and Torrie, J. H. (1980). *Principles and procedures of statistics*. New York, NY: McGraw-Hill.