Empirical Evaluation of Interactive Multimodal Error Correction

Bernhard Suhm
Interactive Systems Laboratories
Carnegie Mellon University, Pittsburgh (USA) and
Karlsruhe University (Germany)
bsuhm@cs.cmu.edu

Abstract - Recently, the first commercial dictation systems for continuous speech have become available. Although they generally received positive reviews, error correction is still limited to choosing from list of alternatives, speaking again or typing. We developed a set of multimodal interactive correction methods which allow the user to switch modality between continuous speech, spelling, handwriting and pen gestures. We integrated these correction methods with our large vocabulary speech recognition system to build a prototypical multimodal listening typewriter. We designed an experiment to empirically evaluate the efficiency of different error correction methods. The experiment compares multimodal correction with methods available in current speech recognition applications. We confirm the hypothesis that switching modality can significantly expedite corrections. However in applications where a keyboard is acceptable, typing correction remains the fastest method to correct errors for users with good typing skills. If the keyboard is not desired, either due to application constraints or user preferences, our multimodal error correction enables state-of-the-art speech recognition technology to deliver keyboard-free text input which beats fast unskilled typing in input speed, including the time necessary to correct errors.

1. Introduction

Our research focuses on the problem of designing usable speech user interfaces despite the unreliability of automatic speech recognition technology. Although there is evidence that baseline accuracy is the main factor determining user acceptance of speech recognition applications [1], we believe the ease of error correction is another important factor which to date hasn't received the attention it deserves. We believe that more intuitive methods of recovering from errors will raise user tolerance towards recognition errors. We address the issue by developing different *multimodal interactive correction* methods allowing the user to switch between different input modalities, such as continuous speech, oral spelling, cursive handwriting, hand-drawn gestures, choosing among a list of alternatives, and typing. In previous work [2], a high fidelity wizard-of-oz simulation suggested that switching modality in repeated errors should significantly expedite error correction and alleviate user frustration.

To empirically evaluate our multimodal correction methods and test this hypothesis, we engineered a prototypical multimodal listening typewriter. Details of the design and algorithms to increase accuracy of recognizing multimodal repairs are described elsewhere [3,4]. This paper lays the foundation for a systematic empirical evaluation of error correction in speech user interfaces. We describe an experiment which compares multimodal correction with correction methods available in current commercial systems. Our study confirms that multimodal flexibility can expedite error correction and that users develop good intuitions regarding the accuracy of a particular mode.

In addition, we present a new interactive correction technique which allows to perform repairs on the level of letters within a word. At least in large vocabulary tasks, many recognition errors consist of substitution, deletion or insertion of one or two letters. In such cases, requiring the user to repeat the whole word is clearly not efficient. Instead, we allow the user to replace, insert or delete letters within a partially correct word. Our experiment shows that such *partial word correction* can significantly increase accuracy of repair.

2. Partial Word Correction

Error correction can be done on different levels: on a sentence or phrase level, on the level of single words or on the level of letters within a word. Which level is appropriate may depend on the modality to be used for correction, on constraints from the recognition technology or on efficiency constraints. For example, it is very natural to say multiple words, as opposed to spelling orally; current recognition technology allows only isolated word cursive handwriting recognition; and it may be faster to correct only the one or two letters which are wrong instead of having to repeat the whole word.

In addition to word-level correction methods, we implemented methods to select, delete, replace and insert one or more letters within a word. To maximize transparency and ease of use, modalities are triggered in the same way as for repair on the word level, which are similar to those used by text editing professionals. Only for selecting letters within a word we had to define a new gesture. Since speaking parts of a word continuously is not intuitive, we exclude continuous speech as modality for partial word correction.

To apply the concept of exploiting repair context [4] to partial word corrections, we use constraints on the word level in the following way. After letters within a word have been deleted or selected, decoding of the next repair input is limited to all words which complete the word fragment to a word within the vocabulary. This algorithm can dramatically reduce the number of possible alternatives for the repair input. For instance, in our dictation application, the vocabulary size typically decreases from 20,000 words to less than 100 words. A drawback of this algorithm is that it fails if the recognition error was caused by a word which is outside the vocabulary (out-of-vocabulary word, new word). As the pronunciation of parts of a word in general is not intuitive, we exclude speech as input modality for partial word correction, limiting it to spelling orally and handwriting.

3. The Experiment

3.1 Evaluation Measures

Early work of Baber et al. on modeling error correction [5] pointed out that correction techniques are difficult to compare because their performance is closely related to their implementation. A systematic evaluation framework for error correction has to define measures to evaluate error correction performance which overcome the dependence on implementation.

The user’s effort in correcting an error is a compound of time required by the user to provide repair input, response time of the system, accuracy of automatic interpretation of repair input and naturalness of interaction. We propose to combine accuracy and time factors into the normalized (by number or errors) error correction speed *V_correct* how many errors can be corrected successfully per minute. Implementation dependence of this measure can be overcome by separating user and modality specific factors from recognizer and interface implementation specific factors. A correction method *m* can be characterized by the word accuracy *WA(m)* of a single attempt to correct an error using *m* (which corresponds to the average numer of correction attempts until success *N(m)*), by the time *T_input(m)* necessary to provide one word of input in *m*, by how many times longer than real-time *R(m)* it takes to recognize the user input, and by the additional time *T_overhead(m*) the user needs to plan and initiate *m* and otherwise fiddle with the interface. Under some simplifying assumptions, the relation between these measures can be described by the three equations in Figure 1.

$$V_{correct}(m) = \frac{1}{Tattempt\ (m) \cdot N(m)}$$
$$T_{attempt}(m) = T_{overhead}(m) + T_{input}\ (m) \cdot R(m)$$
$$N(m) = \frac{1}{WA(m\ )}$$

Figure 1: Relationships between evaluation measures

3.2 Experiment Design

We designed a user study to compare three correction strategies: Correction limited to continuous speech and choice among list of alternatives (as available in current speech recognition applications), correction with keyboard and mouse (as in current text editors and dictation systems), and correction offering to switch between different non-keyboard input modalities. In addition, we evaluated whether partial word correction increases the efficiency of repair.

The task was to dictate sentences from the Wall Street Journal and to correct speech recognition errors using different methods. The task goal was to get every word correct. In addition to the available modalities to replace or insert words (by choosing from the list of alternatives, respeaking one or more words, spelling orally one word, handwriting one word or typing) the experimental conditions differ along two additional dimensions: whether simple hand-drawn gestures are available to delete words and position the cursor, and whether partial word correction (PWC) was allowed.

As there is high variation in recognition performance across subjects, we decided on a *within-subject, repeated measures design*. To limit the time required for the experiment, we chose a paragraph with only 200 words, and we instructed subjects to give up correcting any particular error after three failed attempts of providing repair input. From the 2^7 different possible combinations with seven factors (five different input modalities, and availability of gestures and partial word correction), we decided on a set of six correction methods, which are shown as rows in Table 1.

	Choice from N-best list	Respeaking	Spelling	Writing	Typing	Gesture	PWC
Respeak only	X	X				X	
Spell only	X		X			X	
Write only	X			X		X	
Free Choice	X	X	X	X		X	
Free Choice PWC	X	X	X	X		X	X
Emacs					X		

Table 1 : Experimental Conditions

For recognition, we used the JANUS continuous speech recognizer trained on WSJ [6], the connected letter recognizer Nspell [7] and the on-line cursive handwriting recognizer Npen++ [8], all with the standard 20,000 vocabulary from the November 1995 Hub 1 WSJ evaluation. We eliminated the new word problem by adding all out-of-vocabulary words occurring in the test paragraph. We feel the new word problem has to be addressed separately.

3.3 Results and Discussion

Six subjects, all with significant computer experience, participated in the present study. One subject was female, another had a foreign accent, and some subjects had prior exposure to speech recognition technology. Although this sample is not representative for the general public, it can be expected that this bias is irrelevant for the research questions under investigation.

Basic Correction Parameters. Pooling the data of all repair interactions across all experimental conditions, we estimated the various parameters of the performance model of error correction from section 3.1. Table 2 shows the size of the data sets in words, the input speed and the repair accuracy for corrections on the level of words.

	Respeaking	Spelling	Writing	Check N-best list	Typing
Words	603	689	887	548	204
Input Speed [wpm]	53	27	16	45	36
Accuracy [%]	19	80	74	26	95

Table 2: Basic correction modality parameters

Difficulty of recognizing speech repair. Our data establishes empirically why switching modalities can expedite error correction: the accuracy for recognizing repair by respeak is much lower than for initial dictation (-54%, $p<0.001$). The size of the effect is even more surprising as we employed a technique of adapting the language model context at the beginning and ending of a sentence to the current repair, which we proved elsewhere to improve accuracy of speech repairs significantly [3]. Recognizing speech repairs is difficult because the words were misrecognized on the first trial, and because they are hyperarticulated. Although the magnitude of the performance degradation probably depends on the used continuous speech recognizer, we are confident the problem is present in all state-of-the art systems.

Effectiveness of partial word correction. Partial word correction increases significantly ($p<0.05$) both input speed and repair accuracy. For partial spelling corrections, we measured 38 wpm input speed and 100% accuracy, and for partial handwriting correction, 20 wpm input speed and 81% accuracy.

Performance degradation of recognizers in “real use”. Although all of the recognizers used in this experiment perform at more than 90% accuracy on standard benchmark tasks [6,7,8,9], the numbers shown in Table 2 are significantly lower, ranging from a moderate 10% loss for spelling to a dramatic 71% loss for respeaking. Table 3 identifies three major sources for these losses: trading-off accuracy against speed to achieve close to real-time performance, evaluation on benchmark test sets versus data from real use, and difficulty of recognizing previously misrecognized input. We didn’t exclude corrupted input as it occurs during real use of a system. Additionally, our data is biased towards words which are more difficult to recognize for two reasons. First, our data includes many repeated attempts at correcting the same error. This is true for repair in general, but even more for our experiment design. Second, shorter words tend to be more difficult to recognize. Therefore, there is a higher frequency of short words in repair input.

Other factors lower the performance for specific modalities. For instance, for gesture recognition, 12% performance loss can be explained by errors in the automatic classification algorithm for pen input. For handwriting input, we were aware that there are usability problems in writing and gesturing the touchscreen we used for this study. In particular, the user couldn't rest his wrist on the screen while writing, and there was a large parallaxes due to the thickness of the touchscreen.

For the remaining unexplained performance losses we hypothesize lack of robustness of the recognition systems to environmental noise, different types of microphones and pen input devices etc.

	benchmark, any time	benchmark, close to real-time	real use, first attempts	real use, repeats
Continuous Speech	94%	81%	73%	19%
Spelling		93%	80%	60%
Handwriting		94%	77%	52%
(Pen) Gestures		98%	97%	84%

Table 3: Explanation of performance losses

Comparison of correction and input speeds. Table 4 summarizes the overall system performance from a task point of view. The first two rows show how fast error correction is, using different sets of correction methods (corresponding to the different experimental conditions). The first row is the actual correction speed V which we measured during our experiment, the second the predicted correction speed assuming real-time recognition was available for all modalities. In addition, the third row shows predictions for the total throughput of dictation systems which differ in the available correction methods. We calculated how many words per minute can be produced including times spent on corrections, assuming the recognition of the initial dictation was 90% accurate and in real-time.

As can be seen, correction with our prototype is about twice as fast if modality can be switched from respeaking to spelling or handwriting. Surprisingly, allowing the user to choose freely among modalities didn't perform best among the multimodal correction strategies. Subjects apparently failed to learn that repair by respeaking wasn't a very effective correction strategy with our current prototype, despite a 1.5 - 2 hour long tutorial and training session preceding the experiment. Instead, they frequently gave respeaking on try, in most cases unsuccessfully, and had to correct by spelling or writing in a second attempt. - Also against our expectations, the option of partial word correction overall decreased repair speed, due to the difficulties users had with selecting letters within a word. We expect a more usable writable display will remedy this problem.

Clearly, not interpretation speed, but accuracy is the problem of repair by respeak, since “Respeak only” remains the slowest correction method even if real-time recognition was available. For our subjects who typed at fast unskilled speed (40 wpm), correction using keyboard and mouse remained unchallenged in speed .

However, as can be seen in the last row, a dictation system equipped with multimodal correction would make keyboard-free text reproduction possible at a speed higher than fast unskilled typing.

	Respeak only	Spell only	Write only	Free Choice	Free Choice PWC	Emacs
V measured [epm]	3.0	5.8	5.3	5.2	4.8	16.3
V predicted [epm]	4.3	7.6	9.9	7.8	6.6	16.3
System Throughput [wpm]	34	50	60	51	46	78

Table 4: Correction speed (epm=error per minute) and total system throughput (wpm=words per minute) of different correction strategies

4. Conclusions and future work

For the first time, a systematic evaluation of different correction methods for speech user interfaces was performed. We show that switching modalities can significantly expedite corrections. Thus, with currently best speech recognizers, text *reproduction* without using a keyboard can beat fast unskilled typing in speed. However, the usefulness of an automatic dictation system for text *composition* tasks remains unclear since for such tasks, not input speed, but composition skill has shown to be the main limiting factor [10].

In future work, we will perform a more extended user study using a significantly easier-to-handle writable display and a real-time continuous speech recognizer. In addition, we will investigate whether automatic highlighting of likely recognition errors based on confidence measures can further speed up the correction process, since with our current prototype, users spend a significant amount of time on locating errors. - Furthermore, more research is necessary to improve the accuracy of recognizing repair input. For continuous speech, training specific acoustic models or using isolated word recognition could significantly improve the accuracy of repair by respeak. For handwriting recognition, we would need algorithms which increase recognition accuracy on short words. Thus, multimodal correction may eventually beat typing in correction speed.

Also, our prototype provides only a rudimentary solution to the new word problem: new words can be added interactively by typing their orthographic representation. The system automatically adds them to all recognizers involved. However, there is currently no easy way to distinguish whether some recognition error was due to poor modeling of a known words, or whether it was caused by an out-of-vocabulary word, and these two causes require completely different steps to remedy the problem.

5. Acknowledgments

This research was sponsored by the DARPA, Department of Navy, Office and Naval Research, under grant number N00014-93-1-0806. The views and conclusions in the document are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Navy or the U.S. government.

I want to acknowledge the support and guidance from my advisor Alex Waibel. My heartfelt gratitude extends to Brad Myers, Bernd Bruegge, Bonnie John and Suresh Bhavnani for many valuable discussions and advice. Also many thanks to my colleagues of the Interactive Systems Laboratories at Carnegie Mellon University and Karlsruhe University, and Dean Rubine, who made this research possible by developing the continuous speech recognizer JANUS, the connected letter recognizer Nspell, the cursive handwriting recognizer Npen++ and the gesture recognizer used in our prototype.

References

[1] Lai, J., and Vergo, J. “MedSpeak: Report Creation with Continuous Speech Recognition”, *Proceedings of CHI ´97*, pp. 431-438, Atlanta (USA)

[2] Oviatt, S., and VanGent, R. “Error Resolution During Multimodal Human-Computer Interaction”, *Proceedings of ICSLP´96*, Vol.2, pp. 204-207, Philadelphia (USA)

[3] Suhm, B., Myers, B., and Waibel, A. “Interactive Recovery from Speech Recognition Errors in Speech User Interfaces”, *Proceedings of ICSLP ´96*, Vol.2, pp. 861-864, Yokohama (Japan)

[4] Suhm, B., and Waibel, A. “Exploiting Repair Context in Interactive Error Recovery”, *Proceedings of EUROSPEECH ´97*, Vol. 3, pp. 1659-1662, Rhodes (Greece)

[5] Baber, C., and Hone, K.S. “Modeling error recovery and repair in automatic speech recognition”, *Int. J. Man-Machine Studies*, Vol. 39, pp.495-515, 1993

[6] Rogina, I., and Waibel, A. “The JANUS Speech Recognizer”, *ARPA Workshop on Spoken Language Technology*, Morgan Kaufmann, pp. 166-169, 1995

[7] Betz, M., Hild, H., and Waibel, A. “Recogntion of Spelled Names over the Telephone”, *Proceedings of ICSLP ‘96*, Philadelphia (USA)

[8] Manke, S., Finke, M., and Waibel, A. Npen++: “A Writer Independent, Large Vocabulary On-Line Cursive Handwriting Recognition System”, *Proceedings of Int. Conf. On Document Analysis and Recognition, Montreal*, 1995

[9] Rubine, D. “Specifying Gestures by Example”, *ACM Joural on Computer Graphics*, Vol. 25, No. 4, pp. 329-337, July 1991

[10] Gould, J.D. “How Experts Dictate”, *Journal of Experimental Psychology: Human Perception and Performance*, Vol. 4 (4), pp. 648-661, 1978