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 - Commercial dictation systems for continues speech have recently become available. Although they generally received positive reviews, error correction continuous to be limited to choosing from a list of alternatives, speaking or typing. We developed a set of interactive methods to correct errors without using keyboard or mouse, allowing the user to switch between the modalities continuous speech, spelling, handwriting and pen gestures. These correction methods were integrated with our large vocabulary speech recognition system to build a prototypical multimodal listening typewriter. The efficiency of different error correction methods was evaluated in a user study. The experiment compares multimodal correction with other methods available in current speech recognition applications. Results confirm the hypothesis that switching between modalities can significantly expedite corrections. Thus, state-of-the-art speech recognition technology with multimodal error correction makes it possible to input text at a faster speed than unskilled typing, including the time necessary to correct errors. In applications where a keyboard is acceptable, however, typing still remains the fastest way to correct errors for users with good typing skills.

1. Introduction

Our research focuses on the problem of designing more flexible speech user interfaces that overcome the unreliability of automatic speech recognition technology. There is evidence that baseline accuracy is the main factor determining user acceptance of speech recognition applications [1]. The ease of error correction is obviously another important factor, which has not received enough attention. We believe that developing more intuitive methods of recovering from errors will raise user tolerance of recognition errors. *Multimodal interactive correction* methods [3] allow the user to switch between different input modalities, including continuous speech, oral spelling, handwriting, hand-drawn gestures. A high-fidelity wizard-of-oz simulation [2] suggests that switching modalities for repeated errors should significantly expedite error correction and alleviate user frustration.

To evaluate multimodal correction methods and to test this hypothesis, we engineered a prototypical multimodal listening typewriter. Details of the design and the algorithms employed 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 that compares multimodal correction with correction methods available in current commercial systems. Our study confirms that multimodal flexibility can expedite error correction.

We also present a new interactive correction technique that allows the user to repair individual characters within a word. Many recognition errors in large vocabulary tasks consist of substitution, deletion or insertion of one or two letters within a word. Requiring the user to repeat the whole word is clearly inefficient. We developed an option 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: sentence or phrase, single words or characters within a word. The appropriate level 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 speak multiple words, as opposed to spelling them all; current recognition technology allows only isolated word 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.

We therefore implemented methods to select, delete, replace and insert one or more characters within a word. To maximize transparency and ease of use, each of the letter repair modalities are triggered in the same way as for whole words. The pen gestures employed to perform simple editing tasks (e.g., deleting or placing the cursor) are similar to those used by text editing professionals. We had to define a new pen gesture for selecting letters within a word.

To apply the concept of exploiting repair context [4] to partial word corrections, constraints on the word level are exploited 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 that complete the word fragment to a word within the vocabulary. This algorithm dramatically reduces 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 that 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 therefore must initially define measures that 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 of errors) error correction speed *V_correct* as representative of how many errors can be corrected successfully per minute. Separating user and modality specific factors from recognizer and interface implementation specific factors eliminates implementation dependence. A correction method *m* is characterized by the following four measures: First, the word accuracy *WA(m)* of a single attempt to correct an error using *m* (which is related to the average number of correction attempts until success *N(m)*). Second, the time *T_input(m)* necessary to input one word in *m*. Third, the real-time factor *R(m)* which indicates how many times longer it takes to recognize the user input. Finally, the additional time *T_overhead(m)* the user needs to plan and initiate. The relations between these measures are described by the three equations in Figure 1.

$$V_{correct}(m) = \frac{60\ \mathrm{sec}}{T_{attempt}(m) \cdot N(m)}$$

$$T_{attempt}(m) = T_{overhead}(m) + T_{input}(m) \cdot R(m)$$

$$N(m) = \frac{1}{WA(m)},\ if\ \ldots WA(m) = const\ .$$

Figure 1: Relationships between evaluation measures

3.2 Experiment Design

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

Participants had 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. Since recognition performance varies greatly across subjects, we decided on a *within-subject, repeated measures design*. To limit the time taken in the experiment, we chose a paragraph with only 200 words, and we instructed participants to give up correcting any particular error after three failed attempts of providing repair input. From the 2^7 different possible combinations of seven factors (five different input modalities, and availability of gestures and partial word correction), we decided on a set of six correction methods shown in Table 1. The rows correspond to different experiment conditions, and the columns to what correction methods are available in each condition. The “Respeak only” condition captures correction as available in current speech applications, and the “Emacs-like” condition corresponds to correction using keyboard and mouse. The other four conditions compare different multimodal correction methods.

	Choice from N-best list	Respeaking	Spelling	Writing	Keyboard/ Mouse	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-like					X		

Table 1 : Experimental Conditions (shown as rows)

The prototypical multimodal listening typewriter uses 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 words from the test paragraph to the system's vocabularies, because the new word problem should be addressed in a different experiment.

3.3 Results and Discussion

Six participants were evaluated, all had significant computer experience and good typing skills . One was female, another had a foreign accent, and some had prior exposure to speech recognition technology. Although this sample is not representative of the general public, it is reasonable to assume that this bias does not confound the results of this study.

Basic Correction Parameters. Pooling the data of all repair interactions across all experimental conditions, we estimated the various parameters of the error correction performance model 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 [%]	27	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 surprising since we employed a technique of adapting the language model context at the beginning and ending of a sentence to the current repair, which has been shown 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. The increase of performance compared to whole word correction is significant, although the data contains only few examples of partial word corrections (about 20 for each modality).

Performance degradation of recognizers in corrections. 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 65% 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 the experiment, 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 that 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 the effect is even more pronounced due to 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 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 were large parallaxes due to the thickness of the touchscreen.

	benchmark, any time	benchmark, real-time	User study, dictation	Correction, first attempts	Repeated Correction
Continuous Speech	94%	81%	73%	40%	0%
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 row shows how fast error correction is, using different sets of correction methods (corresponding to the different experimental conditions), assuming real-time recognition. The second row predicts the total throughput of dictation systems that 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 almost twice as fast if modality can be switched from respeaking to spelling or handwriting. Allowing the user to choose freely among modalities surprisingly did not perform best among the multimodal correction strategies. Participants 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 tried respeaking once. Since respeaking failed in most cases, they had to correct by spelling or writing in a second attempt. – In contrast to 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.

The predictions shown in the last row show that a dictation system equipped with multimodal correction would make keyboard-free text reproduction possible at a speed higher than fast unskilled typing (40 wpm).

	Respeak only	Spell only	Write only	Free Choice	Free Choice PWC	Emacs
V predicted [epm]	4.3	7.1	6.0	6.1	5.4	14.7
System Throughput [wpm]	34	47	42	42	39	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. Multimodal error correction is almost always successful in the first attempt. 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 touchscreen that is significantly easier to handle. We will also 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 further improve repair accuracy. 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 that increase recognition accuracy on short words.

Our prototype provides 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. There is however 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. The two causes require completely different steps to remedy the problem: a modeling problem can be solved with more robust recognition algorithms, whereas a new word needs to be added to the system's vocabularies.

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. Many thanks to all colleagues working at the Interactive Systems Laboratories at Carnegie Mellon University and Karlsruhe University, who made this research possible by developing the recognizers used in our prototype; in particular Michael Finke, Hermann Hild, Stefan Manke, Ivica Rogina and Monika Woszczyna. Finally, I wish to thank all participants in the study, who had to suffer through long experimental sessions.

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