Speech Dasher: A Demonstration of Text Input using Speech and Approximate Pointing

Keith Vertanen
Montana Tech
Butte, Montana, USA
kvertanen@mtech.edu

David J.C. MacKay
Cambridge University Engineering Department
Cambridge, UK
djcm1@cam.ac.uk

ABSTRACT

Speech Dasher is a novel text entry interface in which users first speak their desired text and then use the zooming interface Dasher to confirm and correct the recognition result. After several hours of practice, users wrote using Speech Dasher at 40 (corrected) words per minute. They did this using only speech and the direction of their gaze (obtained via an eye tracker). Despite an initial recognition word error rate of 22%, users corrected virtually all recognition errors.

Categories and Subject Descriptors

K.4.2 [**Computers and Society**]: Social Issues - assistive technologies for persons with disabilities.

Keywords

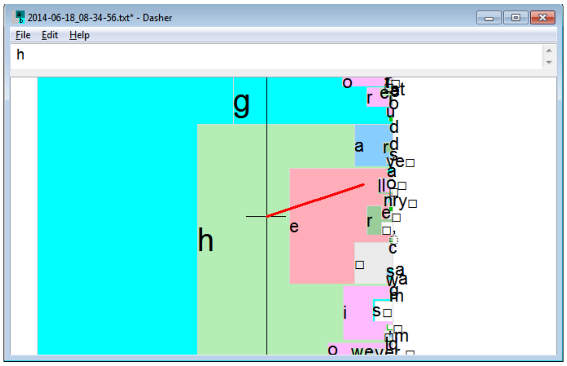
Speech recognition, eye tracking, error correction

1. INTRODUCTION

While people can dictate text to a computer quickly, correcting speech recognition errors can substantially reduce entry rates. Corrections can be made via speech, but recognizers tend to make similar mistakes when the same text is spoken during a correction attempt. Using other input modalities for correction such as a keyboard and a mouse can help avoid a frustrating cascade of errors. But such modalities often require precise motor control that some users lack.

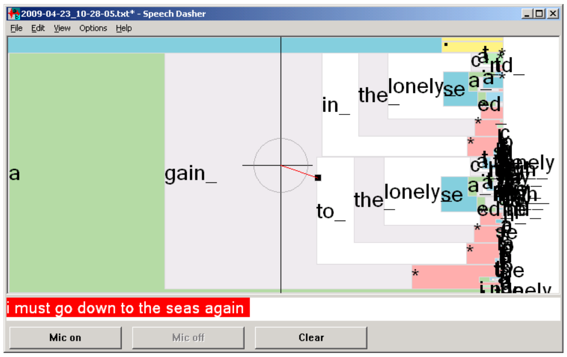
Dasher [4] is a text entry interface in which users write by navigating a world of nested boxes (Figure 1). Each box is labeled with a letter and a box's size is proportional to the letter's probability under a language model. Letters appear in alphabetical order from top to bottom. Users control Dasher using some type of pointing device (e.g. a mouse, stylus, or eye-tracker). Crucially, Dasher works well even when a user's pointing accuracy is poor. Currently Dasher is one of the fastest ways to enter text using an eye tracker [1].

In Speech Dasher, users first speak their desired text to a speech recognizer. Dasher's probability model is modified to predict not only the recognizer's best hypothesis but also its competing alternatives. Here we focus on the performance of Speech Dasher when driven using an eye tracker. Our presentation here is necessarily brief. For further details about the interface, model and evaluation, see [2, 3].


2014-06-18_08-34-56.txt* - Dasher

h

Figure 1: The Dasher interface. The user has currently written "h". The red line shows the direction a user would point in order to write "hello".


2009-04-23_10-28-05.txt* - Speech Dasher

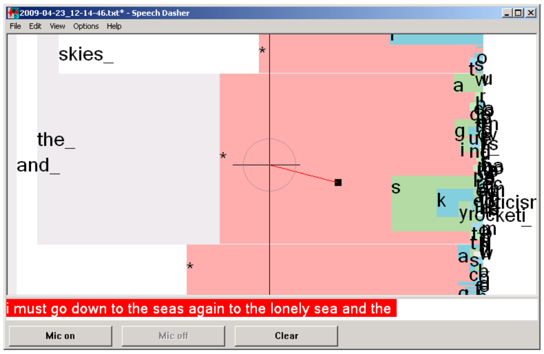
i must go down to the seas again

Figure 2: The Speech Dasher interface. The user is midway through the sentence "I must go down to the seas again to the lonely sea and the sky". The user must now choose between the word "in" or "to".

2. INTERFACE AND MODEL

In Speech Dasher, users first speak their intended text and then navigate using Dasher to confirm and correct the recognition result (Figure 2). *Primary predictions* are the words that Speech Dasher thinks are most probable at the current location. Primary predictions appear in alphabetical order and are always big and easy to navigate to. In Figure 2, the words "in" and "to" are the current primary predictions.

ASSETS'14, October 20–22, 2014, Rochester, NY, USA.
ACM 978-1-4503-2720-6/14/10.
http://dx.doi.org/10.1145/2661334.2661420.



2009-04-23_12-14-46.txt* - Speech Dasher

skies_

the_

and_

i must go down to the seas again to the lonely sea and the

Figure 3: The user wants to write “sky” but the primary prediction was “skies”. The escape box allows “sky” to be spelled using information from the recognition result and from a letter language model.

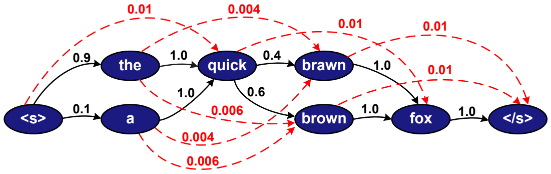


Figure 4: Lattice after the new edges in red were added to cover all one-word insertion errors.

The recognizer may also have a set of less probable word predictions. These *secondary predictions* appear below the primary predictions inside the *escape box*. The escape box is a red asterisk box appearing at word boundaries. Inside the escape box, the model offers the secondary word predictions as well as all the other letters of the alphabet (Figure 3). This makes it possible to write any word, regardless of whether it was predicted by the speech recognizer or not.

The backbone of Speech Dasher’s probability model is the word lattice obtained from the speech recognizer for a given utterance. A lattice is graph containing the word hypotheses explored during the recognizer’s search including acoustic and language model scores. We prune the lattice to remove unlikely hypotheses. We also convert the lattice scores to posterior probabilities. Finally we add edges that skip over words in order to cover all one-word insertion errors (Figure 4). The probability of skip edges was set to a constant multiplied by the probabilities of the skipped edges.

Each box in Dasher needs a probability distribution over all letters (including space). This is done by finding the set of lattice paths consistent with the current symbol history. Given the lattice in Figure 4, if the symbol history is “the_quick_br”, there is one path to “brawn” and one path to “brown”. Given these paths, the model predicts that the next symbol would be either “a” or “o”. A letter’s probability is based on the total penalties incurred by its path.

A sequence of letters may not be in the lattice, for example if the user spells out a word using the escape box. After completing the out-of-lattice word, Speech Dasher tries to get the user back on track somewhere in the lattice. We assume the recognizer has made a deletion or substitution error somewhere. We initiate a new search, allowing paths to make one error (Figure 5). Paths incur different penalties for using a deletion error or a substitution error. If no paths are found using one error, two errors are used, and so on. Using the paths allowed to make one or more errors, we calculate the probability distribution over all letters.

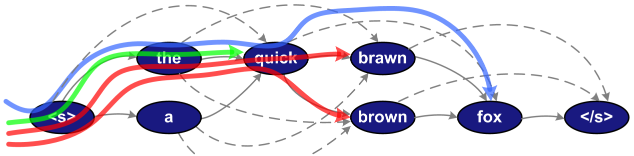


Figure 5: The user has written “the_quiet_”. A substitution at “quick” allows the red paths to reach “brawn” and “brown” and the blue path to reach “fox”. An insertion before “quick” allows the green path to reach “quick”. Currently we would predict the letters “b”, “f” and “q”.

3. FORMATIVE USER STUDY

We conducted a longitudinal study with three users we anticipated would have different levels of recognition accuracy due to their accent. The user denoted US1 was American, UK1 (the second author) was British, and DE1 was German.

Users completed 6–8 training sessions followed by 3 test sessions. In each session, users wrote newswire sentences for 15 minutes using normal Dasher or Speech Dasher. After a break, they wrote for 15 minutes in the other condition. The order of conditions was swapped between sessions. We used a Tobii P10 eye tracker calibrated at the start of each session. We give results on the final 3 test sessions.

Users’ initial recognition results had a word error rate (WER) of 22%. The WER varied significantly between users: 7.8% for US1, 12.4% for UK1, and 46.7% for DE1. In both conditions, we measured the error rate of the user’s final text. Users left few errors uncorrected. The final WER was 1.3% in Dasher and 1.8% in Speech Dasher.

Users’ average entry rate was 20 wpm in Dasher and 40 wpm in Speech Dasher. In Speech Dasher, users showed a wide range of entry rates, presumably due to their differing recognition error rates: US1 54 wpm, UK1 42 wpm, and DE1 23 wpm. On sentences with at least one recognition error, users still wrote at 30 wpm in Speech Dasher.

4. CONCLUSIONS

While our user study was small and used able-bodied users, preliminary results show Speech Dasher may be a promising input method for people who want to dictate text via speech but cannot use a conventional keyboard and mouse for correction. After four hours of practice, users were able to write nearly error-free at 40 wpm despite an initial speech-recognition error rate of 22%.

5. REFERENCES

[1] D. Rough, K. Vertanen, and P. O. Kristensson. An evaluation of Dasher with a high-performance language model as a gaze communication method. In *Proc. AVI*, May 2014.

[2] K. Vertanen. *Efficient Correction Interfaces for Speech Recognition*. PhD thesis, University of Cambridge, Cambridge, UK, April 2009.

[3] K. Vertanen and D. J. C. MacKay. Speech Dasher: Fast writing using speech and gaze. In *Proc. CHI*, pages 595–598, April 2010.

[4] D. J. Ward, A. F. Blackwell, and D. J. C. MacKay. Dasher - a data entry interface using continuous gestures and language models. In *Proc. UIST*, pages 129–137, 2000.