Human-centered compression for efficient text input*

Rani Nelken and Stuart M. Shieber
Division of Engineering and Applied Sciences
Harvard University
{nelken,shieber}@deas.harvard.edu

We describe a novel method for improving the efficiency of natural language text input under degraded conditions, for instance by disabled users or on mobile telephones. Previous approaches to this problem were based on *prediction* of the input text. Prediction, however, requires the user to take overt action to verify or select the system's predictions, causing an increased cognitive load that eliminates any speed advantages (Goodenough-Trepagnier, Rosen, and Galdieri, 1986).

We have developed an alternative method that takes advantage of the duality between prediction and *compression*. Using this method, users input text in a compressed form, which the system then automatically decodes to generate the full text. Because the system's operation is completely independent from the user's, the overhead from cognitive task switching and attending to the system's actions online is stet, enabling efficiency improvements.

Compression follows a simple rule that requires users to drop all vowels except at the beginning of a word, as well as any consecutive doubled consonants. For instance, the word "association" would be abbreviated "asctn".

Since multiple words might be abbreviated to the same character sequence, we have implemented a disabbreviation method that searches for the most likely decoding of the sentence. The method is implemented as a composition of several weighted finite-state transducers (Pereira and Riley, 1997). A key component of this model is a smoothed n-gram language model of the words. The model transduces word sequences, weighted according to the language model, to the corresponding abbreviated character sequences. Unseen words are handled using an n-gram model over letters. To disabbreviate a given input, we use Viterbi decoding to find the most likely path through the transducer that could have generated the abbreviated text. The system is implemented using the AT&T FSM and GRM libraries.

We have conducted an automated decoding study in which we first abbreviated a held-out corpus of Wall Street Journal text of about 840,000 words, yielding a character reduction of 26.4%. We then applied the decoding procedure, and compared the decoding result with the original text, yielding the residual error rates shown for a variety of language models in Table 1.

To assess the usability and the efficiency of the method, we performed a user study, which included 16 participants. After a tutorial on the abbreviation method and a short training period, subjects were asked to abbreviate 10 sentences and were timed in the process, via a specially designed web-site, and using a software-based keyboard. For comparison purposes,

*We thank Ellie Baker, Winston Cheng, Bryan Choi, Reggie Harris, and Emily Morgan for their help in various stages of the implementation of this project. This work was supported in part by grant IIS-0329089 from the National Science Foundation.

Dagstuhl Seminar Proceedings 05382
Efficient Text Entry
http://drops.dagstuhl.de/opus/volltexte/2006/517

Table 1: Performance of the disabbreviation method using a variety of language models

Model	*Average error rate* percent
uniform	51.36%
unigram	8.49%
bigram	4.67%
trigram model over words	4.57%
combined model (3-gram words, 10-gram letters)	3.30%

they were also asked to copy 10 sentences without dropping any of the characters. The abbreviated text was run through the decoding procedure and the result compared with the original. Any discrepancies (due either to user errors or to decoding errors) were highlighted and the user was asked to manually correct them. This process was repeated until no discrepancies remained. The entire experiment took about an hour per subject.

We measured the average speed-up of abbreviation over copying. Due to the effects of learning, the speed-ups increase from round to round. Averaging over all 10 rounds, we measured a speed-up of 12.24% of abbreviation over copying without correction. This speed-up peaked at 18.63% in the final round. Once we also take correction times into account, averaging over all rounds, abbreviation and copying operate at roughly the same speed. The speed-up increases, however, over the 10 rounds, yielding a speed-up of 8.98% in the final round.

References

Goodenough-Trepagnier, Cheryl, Michael J. Rosen, and Beth Galdieri. 1986. Word menu reduces communication rate. In *Proceedings of the Ninth Annual Conference on Rehabilitation Technology*, pages 354–356, Minneapolis, MN, June 23-26. RESNA.

Pereira, Fernando C. N. and Michael Riley. 1997. Speech recognition by composition of weighted finite automata. In Emmanuel Roche and Yves Schabes, editors, *Finite-State Devices for Natural Language Processing*. MIT Press, Cambridge, MA.