

Effect of Window Size on Rate of Communication in a Lexical Prediction AAC System

Horabail S. Venkatagiri

Department of Speech Communication, Iowa State University, Ames, Iowa, USA

The study evaluated the effect of three different "window" sizes (the number of words displayed in the word selection menu) on the efficiency of communication in a lexical prediction augmentative and alternative communication (AAC) program. The results showed that a 15-word window produces significantly higher predictions and fewer keystrokes than a 5-word window. The time required to type the messages was statistically equivalent across windows. However, mean keystroke duration was significantly higher in the 15-word window as compared to the 5-word window. This study shows that (1) a saving in keystrokes does not necessarily result in an increase in the rate of communication in lexical prediction programs; (2) searching a word prediction window for target words places significant cognitive/perceptual demands on even individuals with presumably normal cognitive and perceptual abilities (and probably much more so on AAC users); and (3) a 15-word window produces roughly the same rate of communication as the 5-word window but at a significantly greater saving in effort (keystrokes/switch activations).

KEY WORDS: acceleration, augmentative and alternative communication (AAC), computer effort, lexical prediction, rate, technology

Two major considerations in the implementation of an augmentative and alternative communication (AAC) system for an individual are (1) communication flexibility and (2) communication rate (Scull & Hill, 1988). Communication flexibility refers to the individual's ability to generate an unlimited number of novel utterances. A unique feature of human linguistic communication is that, barring a few phrases used in routine social interaction, we rarely use an utterance in exactly the same form more than once. Instead, we constantly generate unique utterances — utterances that differ from our own previous speech as well as that of others. Beukelman and Yorkston (1982) have shown that AAC users tend to produce unique sentences in their communication as well.

Rate of communication is also important for the successful use of AAC (Beukelman & Yorkston, 1982; Goodenough-Trepagnier & Rosen, 1988). A slow rate of communication tends to reduce the amount and the overall quality of interaction that takes place between AAC users and their conversational partners.

Unfortunately, flexibility and rate generally work against each other (Damper, 1986). An AAC system that is high on flexibility tends to have a slow rate and vice versa. To overcome this trade-off, several acceleration techniques have been developed (Damper, 1986; Vanderheiden & Kelso, 1987). Word prediction

is one of the most common communication acceleration techniques employed in the microprocessor-based communication programs (Heckathorne, Voda, & Leibowitz, 1987; Horstman & Levine, 1990; Scull & Hill, 1988; Swiffin, Arnott, Pickering, & Newell, 1987).

In its simplest form, the word prediction paradigm requires that the program provide a set of likely words (words weighted for frequency, for example) in response to user's keystrokes. If the word the user wants to type is among the words displayed by the program, the word may be incorporated in the message by typing a code, usually a number, associated with the word. This, of course, results in a saving of keystrokes necessary to complete predicted words.

The disadvantage of the word prediction technique is that the user needs to constantly search computer-generated word lists in order to make selections. This wastes time as well as places additional cognitive demands on the user.

The number of words displayed in the word prediction window probably has an effect on the rate of communication in a lexical prediction communication program. If a small number of words are displayed in the window, many short, less frequently used words may not be predicted, since the words are likely to be completely typed out by the user before they appear. If the window contains a large number of words, the

