

Towards technology-assisted co-construction with communication partners

Brian Roark[†], Andrew Fowler[†], Richard Sproat[†], Christopher Gibbons[°], Melanie Fried-Oken[°]

[†]Center for Spoken Language Understanding [°]Child Development & Rehabilitation Center
Oregon Health & Science University

{roark, fowlera, sproatr}@cslu.ogi.edu {gibbons, mfo}@ohsu.edu

Abstract

In this paper, we examine the idea of technology-assisted co-construction, where the communication partner of an AAC user can make guesses about the intended messages, which are included in the user's word completion/prediction interface. We run some human trials to simulate this new interface concept, with subjects predicting words as the user's intended message is being generated in real time with specified typing speeds. Results indicate that people can provide substantial keystroke savings by providing word completion or prediction, but that the savings are not as high as n-gram language models. Interestingly, the language model and human predictions are complementary in certain key ways – humans doing a better job in some circumstances on contextually salient nouns. We discuss implications of the enhanced co-construction interface for real-time message generation in AAC direct selection devices.

1 Introduction

Individuals who cannot use standard keyboards for text entry because of physical disabilities have a number of alternative text entry methods that permit typing. Referred to as keyboard emulation within augmentative and alternative communication (AAC), there are many different access options for the user, ranging from direct selection of letters with any anatomical pointer (e.g., head, eyes) to use of a binary switch – triggered by button-press, eye-blink or even through event related potentials (ERP) such as the P300 detected in EEG signals. These options allow the individual to indirectly select a symbol based on some process for scanning through alternatives (Leshner et al., 1998). Typing speed is a challenge, yet is critically important for usability, and as a result there is a significant line of research into

the utility of statistical language models for improving typing speed (McCoy et al., 2007; Koester and Levine, 1996; Koester and Levine, 1997; Koester and Levine, 1998). Methods of word, symbol, phrase and message prediction via statistical language models are widespread in both direct selection and scanning devices (Darragh et al., 1990; Li and Hirst, 2005; Trost et al., 2005; Trnka et al., 2006; Trnka et al., 2007; Wandmacher and Antoine, 2007; Todman et al., 2008). To the extent that the predictions are accurate, the number of keystrokes required to type a message can be dramatically reduced, greatly speeding typing.

AAC devices for spontaneous and novel text generation are intended to empower the user of the system, to place them in control of their own communication, and reduce their reliance on others for message formulation. As a result, *all* such devices (much like standard personal computers) are built for a single user, with a single keyboard and/or alternative input interface, which is driven by the user of the system. The unilateral nature of these high technology solutions to AAC stands in contrast to common low technology solutions, which rely on collaboration between the individual formulating the message and their communication partner. Many adults with acquired neurological conditions rely on communication partners for co-construction of messages (Beukelman et al., 2007).

One key reason why low-tech co-construction may be preferred to high-tech stand-alone AAC system solutions is the resulting speed of communication. Whereas spoken language reaches more than one hundred words per minute and an average speed typist using standard touch typing will achieve approximately 35 words per minute, a user of an AAC device will typically input text in the 3-10 words per minute range. With a communication partner guess-

ing the intended message and requesting confirmation, the communication rate can speed up dramatically. For face-to-face communication – a modality that is currently very poorly served by AAC devices – such a speedup is greatly preferred, despite any potential authorship questions.

Consider the following low-tech scenario. Sandy is locked-in, with just a single eye-blink serving to provide binary yes/no feedback. Sandy’s communication partner, Kim, initiates communication by verbally stepping through an imagined row/column grid, first by number (to identify the row); then by letter. In such a way, Sandy can indicate the first desired symbol. Communication can continue in this way until Kim has a good idea of the word that Sandy intends and proposes the word. If Sandy says yes, the word has been completed, much as automatic word completion may occur within an AAC device. But Kim doesn’t necessarily stop with word completion; subsequent word prediction, phrase prediction, in fact whole utterance prediction can follow, driven by Kim’s intuitions derived from knowledge of Sandy, true sensitivity to context, topic, social protocol, etc. It is no wonder that such methods are often chosen over high-tech alternatives.

In this paper, we present some preliminary ideas and experiments on an approach to providing technology support to this sort of co-construction during typing. The core idea is to provide an enhanced interface to the communication partner (Kim in the example above), which does not allow them to *directly* contribute to the message construction, but rather to *indirectly* contribute, by predicting what they believe the individual will type next. Because most text generation AAC devices typically already rely upon symbol, word and phrase prediction from statistical language models to speed text input, the predictions of the conversation partner could be used to influence (or adapt) the language model. Such adaptation could be as simple as assigning high probability to words or symbols explicitly predicted by the communication partner, or as complex as deriving the topic or context from the partner’s predictions and using that context to improve the model.

Statistical language models in AAC devices can capture regularities in language, e.g., frequent word collocations or phrases and names commonly used by an individual. People, however, have access to

much more information than computational models, including rich knowledge of language, any relevant contextual factors that may skew prediction, familiarity with the AAC user, and extensive world knowledge – none of which can be easily included in the kinds of simple statistical models that constitute the current state of the art. People are typically quite good at predicting what might come next in a sentence, particularly if it is part of a larger discourse or dialogue. Indeed, some of the earliest work looking at statistical models of language established the entropy of English by asking subjects to play a simple language guessing game (Shannon, 1950). The so-called “Shannon game” starts with the subject guessing the first letter of the text. Once they have guessed correctly, it is uncovered, and the subject guesses the next letter, and so on. A similar game could be played with words instead of letters. The number of guesses required is a measure of entropy in the language. People are understandably very good at this game, often correctly predicting symbols on the first try for very long stretches of text. No purely computational model can hope to match the contextual sensitivity, partner familiarity, or world knowledge that a human being brings to such a task.

A co-construction scenario differs from a Shannon game in terms of the time constraints under which it operates. The communication partner in such a scenario must offer completions and predictions to the user in a way that actually speeds communication relative to independent text generation. Given an arbitrary amount of time, it is clear that people have greater information at their disposal for predicting subsequent content; what happens under time constraints is less clear. Indeed, in this paper we demonstrate that the time constraints put human subjects at a strong disadvantage relative to language models in the scenarios we simulated. While it is far from clear that this disadvantage will also apply in scenarios closer to the motivating example given above, it is certainly the case that providing useful input is a challenging task.

The principal benefit of technology-assisted co-construction with communication partners is making use of the partner’s knowledge of language and context, as well as their familiarity with the AAC user and the world, to yield better predictions of likely continuations than are currently made by the kinds

of relatively uninformed (albeit state of the art) computational language models. A secondary benefit is that such an approach engages the conversation partner in a high utility collaboration during the AAC user's turn, rather than simply sitting and waiting for the reply to be produced. Lack of engagement is a serious obstacle to successful conversation in AAC (Hoag et al., 2004). The slow speed of AAC input is itself a contributing factor to AAC user dissatisfaction with face-to-face conversation, one of the most critical modes of human social interaction, and the one least served by current technology. Because of the slow turnaround, the conversation partner tends to lose focus and interest in the conversation, leading to shorter and less satisfying exchanges than those enjoyed by those using spoken language. A system which leverages communication partner predictions will more fully engage the conversation partner in the process, rather than forcing them to wait for a response with nothing to do.

Importantly, an enhanced interface such as that proposed here provides predictive input from the communication partner, but not direct compositional input. The responsibility of selecting symbols and words during text entry remains with the AAC user, as the sole author of the text. In the preliminary experiments presented later in the paper, we simulate a direct selection typing system with word prediction, and measure the utility of human generated word completions and predictions relative to n-gram models. In such a scenario, n-gram predictions can be replaced or augmented by human predictions. This illustrates how easily technology assisted co-construction with communication partners could potentially be integrated into a user's interface.

Despite the lack of speedup achieved versus n-gram models in the results reported below, the potential for capturing communication partner intuitions about AAC user intended utterances seems a compelling topic for future research.

2 Background and Related Work

Over the past forty years, there has been a vast array of technological solutions to aid AAC users who present with severe speech and physical impairments, from methods for generating possible responses, to techniques for selecting among responses. The simplest methods to generate lan-

guage involve the use of pre-stored phrases, such as "hello", "thank you", "I love you", etc., which are available on many AAC devices. Some studies have indicated that use of such phrases improves the perception of fluid communication (McCoy et al., 2007; Hoag et al., 2008).

Prediction options vary in AAC devices, ranging from letter-by-letter prediction – see Higginbotham (1992) and Leshner et al. (1998) for some reviews – to word-based prediction. Some systems can be quite sophisticated, for example incorporating latent semantic analysis to aid in the better modeling of discourse-level information (Wandmacher and Antoine, 2007). The WebCrawler project in Jeffrey Higginbotham's lab uses topic-related wordlists mined from the Web to populate a user's AAC device with terminology that is likely to be of utility to the current topic of conversation.

Going beyond word prediction, there has been an increased interest in *utterance-based* approaches (Todman et al., 2008), which extend prediction from the character or word level to the level of whole sentences. For example, systems like FrameTalker/Contact (Higginbotham and Wilkins, 1999; Wilkins and Higginbotham, 2006) populate the AAC device with pre-stored phrases that can be organized in various ways. In a similar vein, recent work reported in Wisenburn and Higginbotham (2008; 2009) proposed a novel method that uses automatic speech recognition (ASR) on the speech of the communication partner, extracts noun phrases from the speech, and presents those noun phrases on the AAC device, with frame sentences that the AAC user can select. Thus if the communication partner says "Paris", the AAC user will be able to select from phrases like "Tell me more about Paris" or "I want to talk about Paris". This can speed up the conversation by providing topically-relevant responses. Perhaps the most elaborate system of this kind is the *How Was School Today* system (Reiter et al., 2009). This system, which is geared towards children with severe communication disabilities, uses data from sensors, the Web, and other sources as input for a natural language generation system. The system acquires information about the child's day in school: which classes he or she attended, what activities there were, information about visitors, food choices at the cafeteria, and so forth. The data are then used

to generate natural language sentences, which are converted to speech via a speech synthesizer. At the end of the day, the child uses a menu to select sentences that he or she wants the system to utter, and thereby puts together a narrative that describes what he/she did. The system allows for vastly more rapid output than a system where the child constructs each sentence from scratch.

Perhaps the closest work to what we are proposing is the study of non-disabled adults in Cornish and Higginbotham (No Date), where one of the adults played the role of an AAC user, and the other a non-disabled communication partner. The participants completed a narrative, a map and a puzzle task. Of interest was the relative amount of co-construction of the other’s utterances by each partner, and in particular its relation to which of the partners was the one initiating the attempt to achieve a common ground with the other speaker — the “grounded contribution owner”. In all tasks both the communication partner and the AAC user co-constructed each other’s contributions, but there was the greatest asymmetry between the two users in the puzzle task.

In what follows, we will first describe a preliminary experiment of word completion for a simulated AAC user, using sentences from the Enron email corpus and the New York Times. We then will present results for word completion and prediction within the context of dialogs in the Switchboard corpus. While we ultimately believe that the potential for co-construction goes far beyond simple word completion/prediction, these experiments serve as a first indication of the challenges to an enhanced technology-assisted interface for co-construction with communication partners during novel text generation.

3 Preliminary experiment

In this section, we present a preliminary experiment to evaluate the potential utility of our technology-assisted co-construction scenario. The experiment is akin to a Shannon Game (Shannon, 1950), but with a time limit for guesses imposed by the speed of typing. For the current experiment we chose 5 seconds per keystroke as the simulated typing speed: target sentences appeared one character at a time, every five seconds. The subjects’ task was to provide a

```
with^the^nat
      nation
      national
      nation's
      natural
```

□

Figure 1: Preliminary experimental interface in terminal window, with 4 predicted completions and cursor below

completion for the current word. If the correct word is provided by the subject, it is selected by the simulated AAC user as the next keystroke.

For this preliminary experiment, we used a simple program running in the terminal window of a Mac laptop. Figure 1 shows a screenshot from this program in operation. The target string is displayed at the top of the terminal window, one character at a time, with the carat symbol showing white space word boundaries. Predicted word completions are made by typing with a standard qwerty keyboard; and when the enter key is pressed, the word that has been typed is aligned with the current incomplete word. If it is consistent with the prefix of the word that has been typed, it remains as a candidate for completion. When the current five second interval has passed, the set of accumulated predictions are filtered to just those which are consistent with the new letter that the user would have typed (e.g., ‘i’ in Figure 1). If the correct word completion for the target string is present, it is selected with the following keystroke. Otherwise the following letter will be typed (with the typical 5-second delay) and the interface proceeds as before.

Three able-bodied, adult, literate subjects were recruited for this initial experiment, and all three completed trials with both Enron email and New York Times target strings. The Enron data comes from the Enron email dataset (<http://www-2.cs.cmu.edu/~enron/>) and the NY Times data from the English Gigaword corpus (LDC2007T07). Both corpora were pre-processed to remove duplicate data (e.g., spam or multiple recipient emails), tabular data and other material that does not represent written sentences. Details on this normalization can be found in Roark (2009). Both corpora consist of written sentences, one heavily edited (newspaper), the other less formal (email); and both are large enough to allow for robust statistical language modeling.

Task	Ngram training		Testing		
	sents	words	sents	words	chars
NYT	1.9M	35.6M	10	201	1199
Enron	0.6M	6.1M	10	102	528

Table 1: Statistics for each task of n-gram training corpus size and test set size in terms of sentences, words and characters (baseline keystrokes)

The two corpora were split into training and testing sets, to allow for training of n-gram language models to compare word completion performance. To ensure fair comparison between n-gram and human word completion performance, no sentences in the test sets were seen in the training data. From each test corpus, we extracted sets of 10 contiguous sentences at periodic intervals, to use as test or practice sets. Each subject used a 10 sentence practice set from the NY Times to become familiar with the task and interface; then performed the word completion task on one 10 sentence set from the NY Times and one 10 sentence set from the Enron corpus. Statistics of the training and test sets are given in Table 1.

Language models were n-gram word-based models trained from the given corpora using Kneser-Ney smoothing (Kneser and Ney, 1995). We performed no pruning on the models.

We evaluate in terms of keystroke savings percentage. Let k be the baseline number of keystrokes without word completion, which is the number of characters in the sample, i.e., 1 keystroke per character. With a given word completion method, let c be the number of keystrokes required to enter the text, i.e., if the word completion method provides correct words for selection, those will reduce the number of keystrokes required¹. Then keystroke savings percentage is $100 * (k - c) / k$, the percentage of original keystrokes that were saved with word completion. Table 2 shows the keystroke savings percentage on our two tasks for three n-gram language models (unigram, bigram and trigram) and our three subjects.

It is clear from this table that the n-gram language models are achieving much higher keystroke savings than our three human subjects. Further, our three subjects performed quite similarly, not only in com-

¹Each word completion requires a selection keystroke, but saves the keystrokes associated with the remaining characters in the selected word.

Task	N-gram			Subject		
	1g	2g	3g	1	2	3
NYT	47.4	54.5	56.0	36.5	32.0	32.9
Enron	54.4	61.4	64.4	34.5	32.0	34.1

Table 2: Keystroke savings percentage for test set across models and subjects

parison with each other, but across the two tasks. On the face of it, the relatively poor performance of the human predictors might be surprising, given that the original Shannon game was intended to establish a lower bound on the entropy of English. The assumption has always been that people have better language models than we can hope to learn automatically. However, in contrast to the original Shannon game, our predictions are carried out with a fairly tight time limit, i.e., predictions need to be made within a fairly short period in order to be made available to individuals for word completion. The time limit within the current scenario is one factor that seems to be putting the subjects at a disadvantage compared to automated n-gram models on this task.

There are a couple of additional reasons why n-gram models are performing better on these tasks. First, they are specific domains with quite ample training data for the language models. As the amount of training data decreases – which would certainly be the case for individual AAC users – the efficacy of the n-gram models decrease. Second, there is a 1-character advantage of n-gram models relative to human predictions in this approach. To see this point clearly, consider the position at the start of the string. N-gram models can (for practical purposes) instantaneously provide predictions for that word. But our subjects must begin typing the words that they are predicting for this position at the same time the individual is making their first keystroke. Those predictions do not become operative until after that keystroke. Hence the time overhead of prediction places a lag relative to what is possible for the n-gram model. We will return to this point in the discussion section at the end of the paper.

There are some scenarios, however, where the subjects did provide word completions prior to the trigram language model in both domains. Interestingly, a fairly large fraction of these words were faster than n-gram for more than one of the three

	NY Times		Enron
company	cranbury	creditor	hearing
creditors	denied	facility	suggestions
foothill	jamesway	jamesways	stairs
plan	proposal	sandler	savings
stock	stockholders	warrants	

Table 3: Words completed using subject suggestions with fewer keystrokes than trigram model. Bold indicates more than one subject was faster for that word.

subjects. Table 3 shows the list of these words for our trials. These tended to be longer, open-class words with high topical importance. In addition, they tended to be words with common word prefixes, which lead to higher confusability in the n-gram model. Of course, common prefixes also lead to higher confusability in our subjects, yet they appear to be able to leverage their superior context sensitivity to yield effective disambiguation earlier than the n-gram model in these cases.

Based on these results, we designed a second experiment, with a few key changes from this preliminary experiment, including an improved interface, the ability to predict as well as complete, and a domain that is closer to a proposed model for this co-construction task.

4 Switchboard experiment

Based on the preliminary experiment, we created a new protocol and ran seven able-bodied, adult, literate subjects. We changed the interface and domain in ways that we believed would make a difference in the ability of subjects to compete with n-gram models in keystroke savings. What remained the same was the timing of the interface: characters for target strings were displayed every five seconds. Word completions were then evaluated for consistency with what had been typed, and if the correct word was present, the word was completed and revealed, and typing continued.

Data Our primary motivating case for technology-assisted co-construction comes from face-to-face dialog, yet the corpora from which target strings were extracted in the preliminary experiments were from large corpora of text produced under very different conditions. One corpus that does represent a varied-topic, conversational dialog scenario is the Switchboard corpus (Godfrey et al., 1992), which contains transcripts of both sides of telephone conversations.

The idea in using this data was to provide some number of utterances of dialog context (from the 10 previous dialog turns), and then ask subjects to provide word completions for some number of subsequent utterances.

While the Switchboard corpus does represent the kind of conversational dialog we are interested in, it is a spoken language corpus, yet we are modeling written (typed) language. The difference between written and spoken language does present something of an issue for our task. To mitigate this mismatch somewhat, we made use of the Switchboard section of the Penn Treebank (Marcus et al., 1993), which contains syntactic annotations of the Switchboard transcripts, including explicit marking of disfluencies (“EDITED” non-terminals in the treebank), interjections or parentheticals such as “I mean” or “you know”. Using these syntactic annotations, we produced edited transcripts that omit much of the spoken language specific phenomena, thus providing a closer approximation to the kind of written dialogs we would like to simulate. In addition, we de-cased the corpus and removed all characters except the following: the 26 letters of the English alphabet, the apostrophe, the space, and the dash.

Interface Figure 2 shows the graphical user interface that was created for these trials. In the upper box, ten utterances from the context of the dialog are presented, with an indication of which speaker (A or B) took the turn. Participants are asked to first read this context and then press enter to begin the session. Below this box, the current utterance is displayed, along with which of the two participants is currently producing the utterance. As in the previous experiment, the string is displayed one character at a time in this region. Below this is a text box where word completions and predictions are entered. Finally, at the bottom of the interface, Figure 2 shows two of the five rows of current word completions (left column) and next word predictions (right column).

Perhaps the largest departure from the preliminary experiment is the ability to not only complete the current word but also to provide predictions about the subsequent word. The subject uses a space delimiter to indicate whether predictions are for the current word or for the subsequent word. Words preceding a space are taken as current word completions; the first word after a space is taken as a

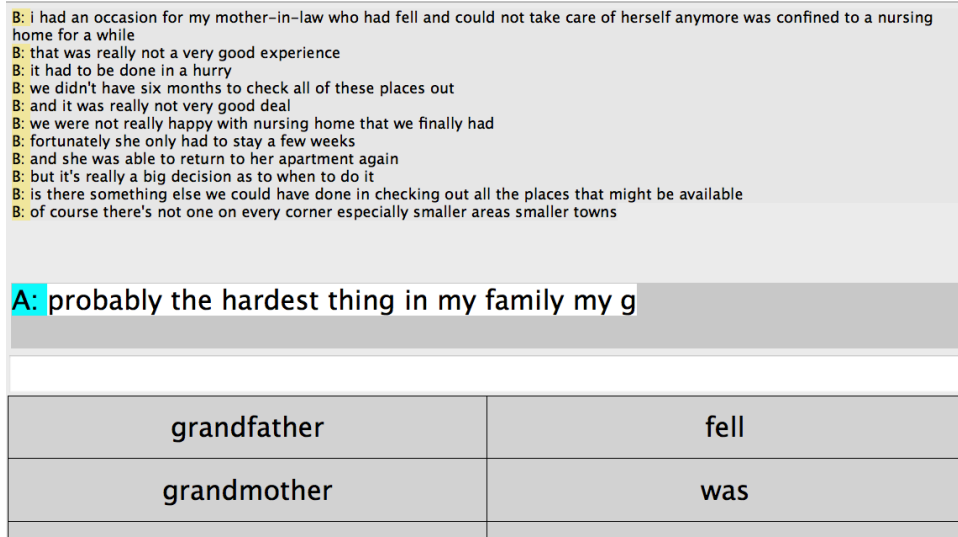


Figure 2: Experimental graphical user interface

subsequent word prediction. To just predict the subsequent word, one can lead with a space, which results in no current word completion and whatever comes after the space as next word prediction. Once the current word is complete, any words on the subsequent word prediction list are immediately shifted to the word completion list. We limited current and next word predictions to five.

We selected ten test dialogs, and subjects produced word completions and predictions for three utterances per dialog, for a total of thirty utterances. We selected the test dialogs to conform to the following characteristics:

1. Each group of three utterances was consecutive and spoken by the same person.
2. Each utterance contained more than 15 characters of text.
3. Each group of three utterances began turn-initially; the first of the three utterances was always immediately after the other speaker in the corpus had spoken at least two consecutive utterances of 15 characters or more.
4. Each group of three utterances was far enough into its respective conversation that there was enough text to provide the ten lines of context required above.

Language models used to contrast with human performance on this task were trained separately for every conversation in the test set. For each conversation, Kneser-Ney smoothed n-gram models were

built using all other conversations in the normalized Switchboard corpus. Thus no conversation is in its own training data. Table 4 shows statistics of training and test sets.

Table 5 shows the results for n-gram models and our seven subjects on this test. Despite the differences in the testing scenario from the preliminary experiment, we can see that the results are very similar to what was found in that experiment. Also similar to the previous trial was the fact that a large percentage of tokens for which subjects provided faster word completion than the trigram model were faster for multiple subjects. Table 6 shows the nine words that were completed faster by more than half of the subjects than the trigram model. Thus, while there is some individual variation in task performance, subjects were fairly consistent in their ability to predict.

5 Discussion

In this paper we presented two experiments that evaluated a new kind of technology-assisted co-construction interface for communication partners during time-constrained text generation. Results

Task	Ngram training		Testing		
	sents	words	sents	words	chars
SWBD	0.66M	3.7M	30	299	1501

Table 4: Statistics for the Switchboard task of n-gram training corpus size and test set size in terms of utterances, words and characters (baseline keystrokes)

Task	N-gram			Subject						
	1g	2g	3g	1	2	3	4	5	6	7
Switchboard	51.0	59.0	60.0	28.7	33.1	28.4	28.6	34.1	31.8	32.5

Table 5: Keystroke savings percentage for Switchboard test set across models and subjects

applied	can't	comes
every	failure	named
physics	should	supervisor

Table 6: Words completed in more than half of the Switchboard trials using subject suggestions with fewer keystrokes than trigram model.

from both experiments are negative, in terms of the ability of our human subjects to speed up communication via word prediction under time constraints beyond what is achievable with n-gram language models. These results are somewhat surprising given conventional wisdom about the superiority of human language models versus their simplified computational counterparts. One key reason driving the divergence from conventional wisdom is the time constraint on production of predictions. Another is the artificiality of the task and relative unfamiliarity of the subjects with the individuals communicating.

While these results are negative, there are reasons why they should not be taken as an indictment of the approach as a whole, rather an indication of the challenges faced by this task. First, we would stress the fact that we have not yet tested the approach in a situation where the user knows the speaker well, and therefore can be presumed to have knowledge well beyond general knowledge of English and general topical knowledge. In future work we are planning experiments based on interactions between people who have a close relationship with each other. In such a scenario, we can expect that humans would have an advantage over statistical language models, for which appropriate training data would not, in any case, be available.

None of the domains that we evaluated were a perfect match to the application: the text data was not dialog, and the dialogs were spoken rather than written language. Further, the tasks that we evaluated in this paper are quite rigid compared to what might be considered acceptable in real use. For example, our task required the prediction of a particular word type, whereas in actual use synonyms or other ways of phrasing the same information will likely be quite

acceptable to most AAC users. In such an application, the task is not to facilitate production of a specific word string, rather production of an idea which might be realized variously. We were interested in the tasks reported here as a first step towards understanding the problem, and among the lessons learned are the shortcomings of these very tasks.

Another take-away message relates to the utility of the new interface itself. The subjects in these trials had the difficult task of quickly predicting intended words; this is also a communication task that may be assisted. Providing access to what n-gram models are predicting may allow the communication partner to quickly select or winnow down the options. Further, it is apparent that single word completions or predictions is not where communication partners are going to achieve order-of-magnitude speedups in communication; rather such speedups may be realized in facilitation of larger phrase or whole utterance production, particularly when the communication is between familiar partners on known topics.

In summary, this paper presented preliminary results on the ability of human subjects to provide word completion and prediction information to users of AAC systems, through simulation of such a new interface concept. While the subjects were not able to match n-gram language models in terms of keystroke reduction, we did see consistent performance across many subjects and across several domains, yielding real keystroke reductions on the stimulus strings. Ultimately, the tasks were not as representative of real co-construction scenarios a we would have liked, but they serve to illustrate the challenges of such an application.

Acknowledgments

This research was supported in part by NIH Grant #1R01DC009834-01. Any opinions, findings, conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the NIH.

References

- D.R. Beukelman, S. Fager, L. Ball, and A. Dietz. 2007. AAC for adults with acquired neurological conditions: A review. *Augmentative and Alternative Communication*, 23(3):230–242.
- Jennifer Cornish and Jeffrey Higginbotham. No Date. Assessing AAC interaction III: Effect of task type on co-construction & message repair. AAC-RERC, available from http://aac-rerc.psu.edu/_userfiles/asha3.pdf.
- J.J. Darragh, I.H. Witten, and M.L. James. 1990. The reactive keyboard: A predictive typing aid. *Computer*, 23(11):41–49.
- J.J. Godfrey, E.C. Holliman, and J. McDaniel. 1992. Switchboard: A telephone speech corpus for research and development. In *Proceedings of ICASSP, volume I*, pages 517–520.
- D. Jeffery Higginbotham and David Wilkins. 1999. Frametalker: A system and method for utilizing communication frames in augmented communication technologies. US Patent No. 5,956,667.
- D. Jeffery Higginbotham. 1992. Evaluation of keystroke savings across five assistive communication technologies. *Augmentative and Alternative Communication*, 8:258–272.
- Linda A. Hoag, Jan L. Bedrosian, Kathleen F. McCoy, and Dallas Johnson. 2004. Informativeness and speed of message delivery trade-offs in augmentative and alternative communication. *Journal of Speech, Language, and Hearing Research*, 47:1270–1285.
- Linda A. Hoag, Jan L. Bedrosian, Kathleen F. McCoy, and Dallas Johnson. 2008. Hierarchy of conversational rule violations involving utterance-based augmentative and alternative communication systems. *Augmentative and Alternative Communication*, 24(2):149–161.
- R. Kneser and H. Ney. 1995. Improved backing-off for m-gram language modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 181–184.
- Heidi H. Koester and Simon Levine. 1996. Effect of a word prediction feature on user performance. *Augmentative and Alternative Communication*, 12(3):155–168.
- Heidi H. Koester and Simon Levine. 1997. Keystroke-level models for user performance with word prediction. *Augmentative and Alternative Communication*, 13(4):239–257.
- Heidi H. Koester and Simon Levine. 1998. Model simulations of user performance with word prediction. *Augmentative and Alternative Communication*, 14(1):25–36.
- G.W. Lesh, B.J. Moulton, and D.J. Higginbotham. 1998. Techniques for augmenting scanning communication. *Augmentative and Alternative Communication*, 14:81–101.
- J. Li and G. Hirst. 2005. Semantic knowledge in word completion. In *Proceedings of the 7th International ACM Conference on Computers and Accessibility*.
- Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.
- Kathleen F. McCoy, Jan L. Bedrosian, Linda A. Hoag, and Dallas E. Johnson. 2007. Brevity and speed of message delivery trade-offs in augmentative and alternative communication. *Augmentative and Alternative Communication*, 23(1):76–88.
- Ehud Reiter, Ross Turner, Norman Alm, Rolf Black, Martin Dempster, and Annalu Waller. 2009. Using NLG to help language-impaired users tell stories and participate in social dialogues. In *12th European Workshop on Natural Language Generation*, pages 1–8. Association for Computational Linguistics.
- B. Roark. 2009. Open vocabulary language modeling for binary response typing interfaces. Technical Report #CSLU-09-001, Center for Spoken Language Processing, Oregon Health & Science University. cslu.ogi.edu/publications/ps/roark09.pdf.
- C.E. Shannon. 1950. Prediction and entropy of printed English. *Bell System Technical Journal*, 30:50–64.
- John Todman, Norman Alm, D. Jeffery Higginbotham, and Portia File. 2008. Whole utterance approaches in AAC. *Augmentative and Alternative Communication*, 24(3):235–254.
- K. Trnka, D. Yarrington, K.F. McCoy, and C. Pennington. 2006. Topic modeling in fringe word prediction for AAC. In *Proceedings of the International Conference on Intelligent User Interfaces*, pages 276–278.
- K. Trnka, D. Yarrington, J. McCaw, K.F. McCoy, and C. Pennington. 2007. The effects of word prediction on communication rate for AAC. In *Proceedings of HLT-NAACL; Companion Volume, Short Papers*, pages 173–176.
- H. Trost, J. Matiasek, and M. Baroni. 2005. The language component of the FASTY text prediction system. *Applied Artificial Intelligence*, 19(8):743–781.
- T. Wandmacher and J.Y. Antoine. 2007. Methods to integrate a language model with semantic information for a word prediction component. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 506–513.
- David Wilkins and D. Jeffery Higginbotham. 2006. The short story of Frametalker: An interactive AAC device. *Perspectives on Augmentative and Alternative Communication*, 15(1):18–21.

- Bruce Wisenburn and D. Jeffery Higginbotham. 2008. An AAC application using speaking partner speech recognition to automatically produce contextually relevant utterances: Objective results. *Augmentative and Alternative Communication*, 24(2):100–109.
- Bruce Wisenburn and D. Jeffery Higginbotham. 2009. Participant evaluations of rate and communication efficacy of an AAC application using natural language processing. *Augmentative and Alternative Communication*, 25(2):78–89.