Programming Robots at the Museum

ABSTRACT

Author Keywords
assistive technology; prolonged engagement; collaborative conversation; story telling game

ACM Classification Keywords
H.5.2 Information Interfaces And Presentation: User Interfaces - Interaction styles

General Terms
Human Factors; Design

INTRODUCTION

INTRODUCTION

We collaborate with a motion and speech impaired individual, Henry, who enjoys conversing. The speech impairment is complete. Quadriplegia while severely limiting, does allow Henry to move his head in affirmation or negation. His hand can operate a mouse button. One of his communication modes is via a text-to-speech system (*tts*). A camera mounted on top of his laptop tracks a confetti sized white dot pasted on the lower left of his glasses. The resulting cursor control allow Henry to hunt down the keys of an onscreen keyboard. On a very good day the resulting speed is 15 words per minute. The tts produces sound once a sentence is complete. Uttering words as Henry types them would work, but this approach makes it difficult for listeners to track the very slowly evolving sentences in their minds. Poor tts performance for some words would additionally impede comprehension, which is supported by context when a full sentence is pronounced in a flow.

This slow communication channel results in very frustrating experiences during gatherings like parties. A guest will make a remark to Henry, who will go to work on an answer. To the conversation partner Henry looks frozen, peering at his laptop screen whose back surface reveals nothing to the expectant partner. Often the potential conversation partner wanders off bewildered before Henry can finish his response sentence.

One improvement would be to install a second display on the backside of Henry's laptop. Inexpensive, USB based options are available. This option would at least allow a listener to understand that information is forthcoming. The option does not help as much as possible in keeping the listener(s) actively involved in the conversation.

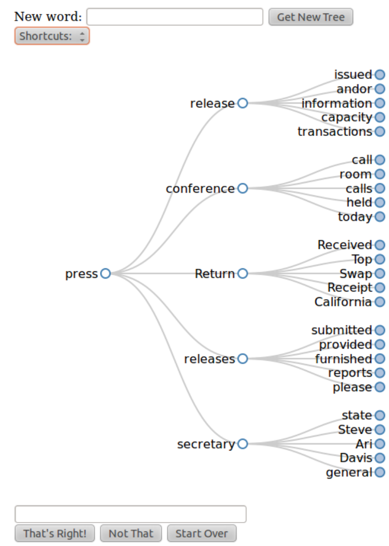

Figure 1. Screenshot of an EchoTree browser.

In an attempt to ameliorate the situation further we developed EchoTree. EchoTree is a distributed, collaborative word tree. Figure 1 shows an example.

USER EXPERIENCE

EchoTrees are browser applications that can be viewed anywhere, by multiple users. For example, the secondary display facing a conversation partner in the option discussed earlier could show an EchoTree. Alternatively, a conversation partner's smartphone could *tune in* to Henry's EchoTree. After describing the interface and some of its interaction affordances we explain how an EchoTree can be related to Henry's response activity.

The word on the left most node of Figure 1 is called the *root word*, in case of the Figure the root word is *press*. Links connect the root node to the five 'most frequently following' other words. 'Most frequently following' is measured in the context of an underlying text collection. We will discuss this dependency later.

Each of the word followers is connected to the five words that most frequently follow *it* in the underlying texts. Following, for example, the top branches in Figure 1 we find the sequence $press, release, issued$.

In the current implementation anyone viewing an EchoTree on their device may click or tap on one of the circles. In response all the displayed EchoTrees are *re-rooted*: the selected word becomes the root of a new tree. All follow words are recomputed, and a new tree is displayed on all browsers. Figure 2 shows the result of clicking on the word *today* in Figure 1's tree. Alternatively, one may type a new word into the text box at the top, and click on the button *Get New Tree*.

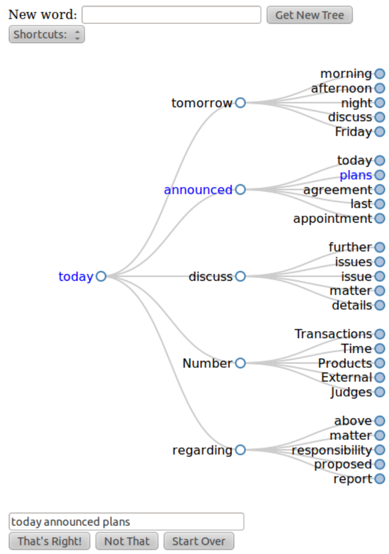

Figure 2. EchoTree re-rooted in 'today'. Blue words were user selected, adding them to the sentence box.

This action again creates a new tree, rooted at the new word, and displayed everywhere.

Instead of clicking/tapping on a circle, one may target one of the words with a click or tap. This action causes two changes in the display. First, the targeted word turns blue, and second, the word is added to the text field at the bottom of the display. This field is called the *sentence box*. Figure 2 shows the result of clicking on *today*, then *announced*, and finally *plans*.

The EchoTree facility can be used for a number of purposes. In the context of Henry interacting in a conversation, the facility may be used as follows.

Collaborative Conversing

As Henry types words, EchoTrees in the browsers of all tuned in listeners will evolve, the latest word always serving as root. Any word that Henry completes is additionally appended to the sentence box. Listeners, meanwhile may active think ahead and guess where Henry might be headed. After scanning the current EchoTree they can call out possibilities. If Henry hears a hit, he can click on the *That's Right* button, or nod. After a successful guess Henry can continue, skipping one of more words.

Of course, if Henry notices a word that matches his intention, he can click on that word in the EchoTree himself. If the content of the sentence box is hopelessly wrong, the *Start Over* button will clear the box, and turn off all blue (i.e. selected) words in the EchoTree.

The underlying machinery will not include a fixed set of stopwords in the EchoTrees. This filtering helps maximize use of the limited screen real-estate. Sometimes participants may wish to enrich sentences with fill words. The pull-down menu below the *New word* field fills that need (Figure 3). Selecting any of these words will enter them in the sentence box. Again, in the current implementation this addition appears in all views of the EchoTree. Note that the use of EchoTrees

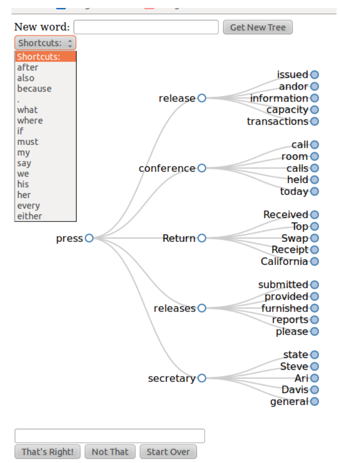

Figure 3. Shortcut words are available as fillers for the sentence box. These stopwords will not occur in EchoTrees.

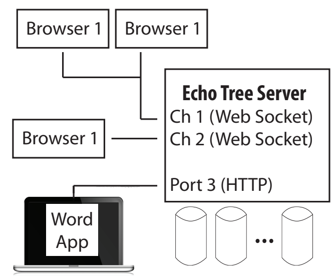

Figure 4. EchoTree architecture. Channels are implemented as WebSocket ports. Browsers tune in to different EchoTree channels.

for collaborative conversation is not limited to face-to-face situations, like parties. Communication with Henry via the telephone are also an option. The remote participant tunes into Henry's EchoTrees, and offers guesses over the phone. Since Henry nodding assent is not an option in this scenario, the *Not That* button can serve as a negative response.

Story Telling Game

Instead than supporting a directed conversational thread, EchoTrees can serve as a collaborative story telling facility. Geographically distributed players click on words in a starting EchoTree, collaboratively adding words to the sentence box. Various rules might govern the process. Participants might take turns, or work at speed without sequence limitations. Re-rooting might earn a demerit, while opening new possibilities.

This application is accessible to disabled and non-impaired participants. Again, the goal is mutual engagement. Appropriate rules in this scenario can provide satisfying interaction even in the presence of speed limitations.

Architecture

Figure 4 shows how the EchoTree system is constructed. Central, or distributed EchoTree servers each manage some number of distinct EchoTree channels. All facilities described above operated on one channel. All shared EchoTree views are refreshed, and request re-rooting on one channel.

Multiple, unrelated EchoTree sequences may be served by a single server, using different ports. In Figure 4 Browers 1 is separated from Browsers two and three, which share all EchoTree transmissions.

Browsers communicate with EchoTree servers via WebSocket connections, which are bi-directional. This bidirectionality enables the re-rooting requests from browsers back to the server.

Figure 4 also shows an HTTP port family. These ports can push new words to the echo server, triggering the multicast of a new EchoTree to all browsers on the respective channel. These HTTP connections are simpler than the more versatile WebSocket connections. They are provided for easy connection with word entry support applications on Henry's machine. For example, Henry uses an application that offers word completions as he types a word. The HTTP method of pushing words to the EchoTree server can be attached to this application. This method allows Henry to focus on typing in his usual environment, and not being forced to interact with a browser's *New Word* entry to push a new word (and consequently new EchoTree).

Figure 4 shows a series of databases with word pair frequencies that are the basis for the generation of the trees. Each database holds lists triples: a word, a follower word, and a frequency count. These *bigram* counts may originate from any text collection. The trees of the above figures are based on bigrams from the Enron collection **??**. In the following section we examine some aspects of these underlying collections, which strongly influence the induced EchoTrees.

EFFECTIVENESS EXPERIMENT

EchoTree could be effective along several dimensions. Each dimension implies a different evaluation method:

1. Conversation acceleration through word prediction.
2. Conversation acceleration by conveying intent.
3. Encouragement through bi- or multilateral engagement.
4. Fun

We measured the first item in an experiment, which we will describe in this section. This dimension works by accurately predicting the next word the current utterance originator is planning to type.

The second dimension in the above list contributes not through prediction, but by helping the listener imagine indirectly where the UO is heading. The inspiration might for example arise from associations with words that occur in the EchoTree, though those word not literally in the UO's plan.

The third dimension simply helps keep the conversation partners connected. All listeners can at least observe progress, and maybe anticipate a chance to complete a sentence soon.

The final dimension, finally, contributes by raising enjoyment in the interaction. This element is most obvious in the story telling scenario. All dimensions can contribute cumulatively, and are worthy of evaluation. An examination of the word prediction power is an important start.

	Enron Collection	**Google Bigrams**
Henry		
Enron EmpX		

Table 1. Two-by-two experiment matching utterance originators with word prediction sources.

Setup

As with all word prediction, EchoTree's predictive power depends on the match between the word source and the underlying frequency data. We tested and compared four such pairings in a two-by-two experiment, as shown in Table 1.

Results

DISCUSSION

RELATED WORK

FUTURE WORK

CONCLUSION