SUBLIME: A SPEECH- AND LANGUAGE-BASED INFORMATION MANAGEMENT ENVIRONMENT

Jahanzeb Sherwani, Stefanie Tomko, Roni Rosenfeld

Computer Science Department
Carnegie Mellon University
{jsherwan, stef, roni}+icassp@cs.cmu.edu

ABSTRACT

With ever-increasing amounts of information to be organized in people's daily lives, current mechanisms for personal information management (PIM) leave much room for improvement. We present Sublime, a distributed, multimodal, and mobile environment for speech-based personal information management. In addition to discussing the design philosophy and evolution of the current prototype, this paper describes the functionality and architecture of the system, and indicates future research directions for the project.

1. MOTIVATION

In the modern age, the amount and complexity of personal information acquired and used by an individual is much greater than at any other time in history: people must manage to-do lists, reminders, medical records, financial information, reservation numbers, bright ideas, etc. Current mechanisms for PIM are numerous, yet none are satisfactory: handwritten notes, voice memos, PDAs, desktop-based text editing, spreadsheets & databases all have their specific advantages and disadvantages, as summarized in Table 1.

The lack of a unified solution for PIM means that users are forced to compartmentalize their information, either by using different PIM methods at different times, and/or copying information from one PIM environment to another to overcome its limitations. The disadvantages of the compartmentalization of personal information are well-documented [1]. However, we are not aware of any solution that combines the advantages of different mechanisms to provide one unified environment for PIM. Additionally, speech technologies have matured to the point where they may be incorporated into PIM environments. Our research builds upon the VoiceNotes project [3, 4], which to our knowledge is the first example of a PIM application using speech recognition.

	Paper	PDA	Voice Memos	Desktop
Max data entry speed in WPM [2]	30	18	100	60
Quick data retrieval	✓	✓		✓
Easy to search				✓
Portable	✓	✓	✓	
Supports complex structures	✓	✓		✓
Can give reminders		✓		✓

Table 1: Summary of existing PIM mechanisms

2. RELATED WORK

VoiceNotes [3] is a handheld interface for recording and accessing audio recordings. Although there is no display, it is multimodal in the sense that all commands can be issued either through speech, or through buttons, or a combination of both. In addition to a fixed set of keywords, the system uses template matching to recognize user-defined "categories", and offers an element of random-access that is not found on standard tape-recorders. All commands are single keywords, without a higher-level grammar. Finally, there is no representation of the information in any format other than as an audio waveform.

Conversational VoiceNotes [4] builds upon its predecessor by specifying a more flexible grammar (e.g., **take a note** as well as **take an important note**) with greater functionality (e.g., **play all my important notes**). It also allows simple correction dialogs (**no I said…**). Additionally, support for reminders was added, whereby any note could be appended with a date and time value for when the reminder was to be given to the user.

SCANMail [6] is a multimodal interface for browsing transcripts of voicemails. Users can search all voicemail for

specific words, and can play back specific portions of voicemails using the transcript. Speech is not used to control the interface; the only speech component in the system is the voicemails themselves.

MiPad [7] is a multimodal system with a number of PIM applications (calendar, email, contact list & memos) leveraging specific language models for data entry into specific GUI text fields. Although it appears there are many parallels with MiPad and Sublime, the lack of detailed descriptions of the interface and user studies makes it difficult to compare the two systems.

2. CONTEXTUAL INQUIRY

To better understand PIM practices by people at large, we conducted a contextual inquiry involving eleven members of the university community: an undergraduate student, six graduate students, two staff members and two faculty members to elucidate their PIM practices, as well as gauge their satisfaction with their current methods. In addition to a qualitative understanding, we were able to collect statistics on their significance of usage for the following PIM mechanisms: the notes function in PDAs, desktop-based text editing, emails to themselves, text notes on their cell phones, and paper (notebooks, scraps of paper, and Post-It notes). The results are summarized in Figure 1.

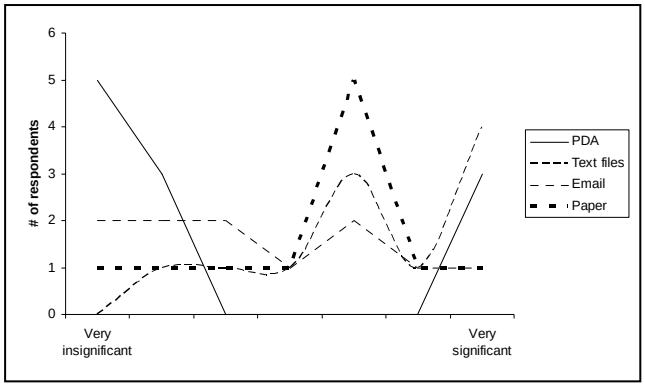

Figure 1: Usage of specific PIM mechanisms. Cell phone usage was almost nonexistent and is left out of the graph.

Additionally, while five of the respondents said they were not satisfied with their current practice, it was unanimously agreed that there was room for improvement, and all but one said they would try a speech based system on their PDA or phone.

The data that the users stored in their respective PIM systems had a wide distribution (e.g., "messages for roommates", "things to do with kids once I have kids", "books to read", "frequent flier numbers", etc.), although the most-used data were noting down appointments, reminders and to-do lists.

There are a couple of interesting things to note about the above data. PDAs are heavily used by a few people, and are almost entirely unused by the rest. Whether this is because of their cost, or because of usability issues, or both, is outside the scope of this study. Next, while paper is a popular choice, it is not used very significantly by many people; similarly with self-directed emails. Desktop-based PIM methods, however, are very significantly used.

3. INITIAL DESIGN

Based on the preliminary study, we decided to design our system with the simplest possible information model for users to learn, while still allowing some flexibility. Information would be stored in the form of lists, each with a title, and any number of items. For instance, a user could have a list titled **Grocery list**, with items **Eggs**, **Milk**, and **Cheese**. While we worked with the idea of more complicated structures, such as hierarchical lists, and user-definable types and structures, we decided to go with the simplest model to begin with, and grow the model space after we gained a basic understanding of how successful users are with simple models.

We began by developing a speech-only system for use over the phone, which would allow anyone to access their information without the need for specialized hardware. Additionally, we envisioned a web-interface where users could access their data through the GUI modality. One important design choice we made at this time was to interleave "interpretable speech" (meta language, e.g., **add a note**) with "non-interpretable speech" (actual information e.g., **grocery list**). In this regard, there were huge similarities with our system and the first VoiceNotes system.

The first prototype was developed on the RavenClaw architecture [5], which allows for flexible, conversational dialog systems. Thus, users would call in, and hear a "how may I help you?" prompt. Through an informal user study involving five users, however, we quickly discovered that such a prompt is confusing when users don't know what their options are and what the underlying assumptions of the system are (i.e., how is information structured, what functionality is supported). Based on this, we redesigned the initial prompt as a menu of options, to prime users lexically as well as functionally (in terms of knowing what the system can do).

Another hurdle in such an arrangement is that it isn't possible for users to search for specific words in their data, since by treating the "non-interpretable" segments of speech as a black box, we weren't able to allow users to reference subparts of their recorded speech. Additionally, our design choice had the consequence of making sharing information between users impossible, and also meant that the web-based GUI would not contain any useful information for users to see other than a graphical structure with audio icons

to represent their data. Finally, the lack of a GUI for users during interaction with the system meant that it was difficult for them to ground themselves in terms of the dialog state, as well as for them to know how their information was organized. It is also worth noting that we were able to replicate many of the observed interaction effects described in [3], such as users losing track of their bearings while navigating the audio-only information space, as well as feedback being perceived as too wordy or too terse at different times.

Based on the above experiences, we decided to move to an actual multimodal system incorporating speech and GUI elements simultaneously, and also diminishing the boundary between interpretable and non-interpretable speech.

4. THE CURRENT PROTOTYPE

The system we have developed since then is a multimodal interface that runs on a PDA, and uses grammar-based ASR to recognize the Sublime meta-language, and statistical language model (SLM)-based ASR to recognize actual user information. Audio is sent over a Wi-Fi connection, with actual ASR being performed on a desktop server. Recognition results are streamed back in real time to the PDA. This is accomplished using the Microsoft Speech Server architecture with the PocketPC speech plugin, and Dragon NaturallySpeaking for the SLM ASR. A screenshot is shown in Figure 2.

The functionality supported by the system includes adding, editing, and deleting lists through the stylus and through voice, searching for any words that are already in the system (even immediately after entering new words), and dynamically updating a user's central information immediately after edits are made on the PDA. The same information (audio and text) can be accessed on the web at any time.

By shifting to a multimodal environment, the system is now able to effectively ground the user in terms of knowing what the system heard: currently, it displays the exact hypothesis that it recognized from the user's utterance, so the user has a first-order grounded belief of what the system's current state should be. Further, we have anecdotal evidence that suggests that seeing a graphical representation of their personal information during navigation and manipulation allows users to better create a mental model that matches the system's model. Finally, since there are actual text representations of users' information, it now becomes possible to search for individual words in a specific entry, which anecdotal evidence suggests is preferred by users.

Initial results with this system are promising. Anecdotally, we have seen that users are able to learn to use the system much quicker than in the audio-only system, and there is a much stronger match between their expectations and the system's response. Sample dialogs are shown in Figure 3.

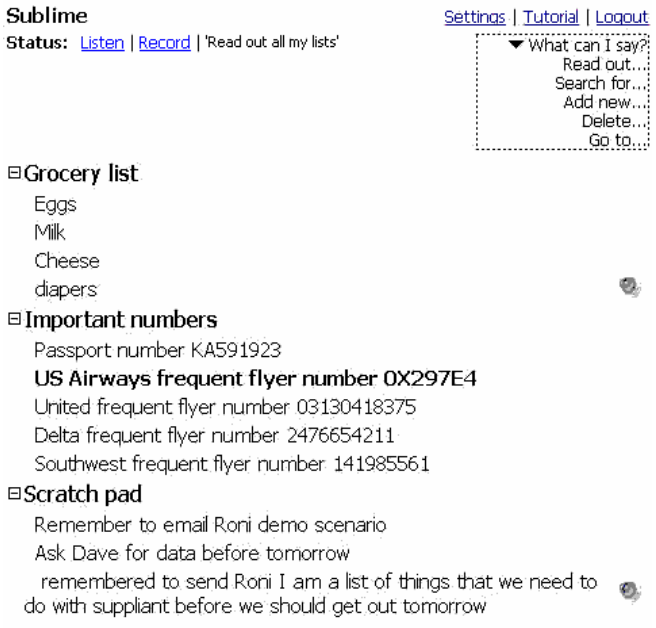

Figure 2: Screenshot of Sublime. The audio icons denote recorded speech that may be played back, along with first-pass transcriptions displayed in a few seconds after recordings are made. The bold line indicates a search result from a recent search (**Search for US Airways**).

Welcome to Sublime
Read out all my lists
You have 3 lists, Grocery list, Important numbers, and Scratch pad.
Read out Scratch pad
Scratch pad had 2 items: Remember to email Roni demo scenario, and Ask dave for data before tomorrow
Add new item to this list
Adding new item to Scratch pad, press record when ready
[user double-clicks the hardware talk button]
Remember to send Roni, umm, a list of things that we need to do with Sublime before we ship it out tomorrow
Entry recorded [hourglass icon appears for a few seconds, disappears, and is replaced by an audio icon, with the first-pass transcript appearing in place of the text]
Search for Roni
Two items found, first item: Remember to email Roni demo scenario
Next result
Next result: [in user's voice] Remember to send Roni, umm, a list of things that we need to do with Sublime before we ship it out tomorrow

Figure 3: Sample dialog between user and Sublime

Using this prototype, we hope to investigate a large array of questions, the more significant of which are described below.

5. FUTURE DIRECTIONS

5.1. Turn-taking

There are a variety of interaction options: push-to-talk, tap-and-talk, hands-free. Within these there are further combinations of initiative options: system-initiative, user-initiative, and mixed-initiative. Additionally, within specific states, it may be preferable for the system to prompt the user itself, even though the rest of the interaction is user-initiative, such as while recording user information.

5.3. Information structures

While the system currently works only with one-level deep lists, we are interested in looking into different information structures to allow the user more flexibility in the kind of data they wish to store. Examples include hierarchical lists, commonly-used templates such as to-do lists and shopping lists, free-form notes, association lists (hash tables), and many more. Investigating each structure involves evaluating different speech interfaces to such structures, and examining user-centric indicators as to whether the benefits of such structures and interfaces outweigh the costs.

5.4. Mobile interfaces for dictation correction

While dictation interfaces for desktop systems have been researched and productized, we aren't aware of similar interfaces for mobile devices. We plan to design and evaluate interfaces for multimodal (stylus + voice) dictation correction on Sublime, and have anecdotal evidence that suggests that these mechanisms will easily surpass traditional text entry mechanisms on PDAs

5.5. Meta-language design issues

We are interested in looking at the costs and benefits of designing a language for users with which to speak about their personal information, and test it to see how learnable and habitable it is. We are investigating what design choices give the maximum returns on learning investment to users.

5.6. Ubiquitous access

While the information stored in Sublime is already accessible via any web-browser, we wish to investigate mobile use of Sublime "in the wild": namely, outside the lab and without the constraint of being in a WiFi hotspot. With the advent of 3G cellular networks, this is now a real possibility, and one that we are actively pursuing. By testing the system with real users in real situations, we hope to gain access to user interaction data that is more reflective of actual user behavior than laboratory-based user studies.

6. CONCLUSION

In this paper, we have presented Sublime, a speech-based, distributed, multimodal, mobile environment for personal information management. We described the initial contextual inquiry, which was the basis for initial design choices in the system. The first prototype was presented, as well as its shortcomings. Finally, we presented details of the current system, and indicated the multiple lines of research in multimodal interface design we are undertaking.

According to the criteria described in Table 1, we feel Sublime has the potential to be a winning PIM mechanism: it supports high speed data entry (at the speed of speech), quick data retrieval, easy and robust searching, is portable, can be expanded to support more complex structures, and can be extended to give reminders.

We would like to thank Nuance (specifically Francis Ganon and Francoise Renaud) for contributing the Dragon NaturallySpeaking SDK, which was used in the current prototype.

7. REFERENCES

[1] "Workspaces that work: towards unified personal information management", Boardman, Proceedings of HCI2002, People and Computers XVI - Memorable yet Invisible, Volume 2, 216-7, London, 2002.

[2] "Developing a voicespelling alphabet for PDAs", J. R. Lewis and P. M. Commarford - IBM Systems Journal, Vol. 42, No. 4, pp 624-638, 2003.

[3] "VoiceNotes: A speech interface for a hand-held voice notetaker", L. J. Stifelman, et al., Proceedings INTERCHI, Amsterdam, pp. 179-186, 1993.

[4] "A Tool to Support Speech and Non-Speech Audio Feedback Generation in Audio Interfaces", L. J. Stifelman, Proceedings UIST '95: Eighth Annual Symposium on User Interface Software and Technology, pp. 171-179, 1995.

[5] "RavenClaw: Dialog Management Using Hierarchical Task Decomposition and an Expectation Agenda", Dan Bohus, Alex Rudnicky, Eurospeech, Geneva, Switzerland, 2003

[6] "SCANMail: Audio Navigation in the Voicemail Domain", S. Whittaker, et al., Proceedings HLT, San Diego, 2001.

[7] "Distributed Speech Processing in MiPad's Multimodal User Interface" , L. Deng, et al., IEEE Transactions on Speech and Audio Processing, vol. 10, no. 8, pp. 605-619, November 2002.