Proceedings of the International Workshop on Information Presentation and Natural Multimodal Dialogue (IPNMD 2001), Verona, Italy, 2001.

FIRE: An Information Retrieval Interface For Intelligent Environments

Krzysztof Gajos and Ajay Kulkrani*

MIT Artificial Intelligence Laboratory
{kgajos, kulkarni}@ai.mit.edu

Abstract. Searching for relevant information on the world-wide web is often a difficult and frustrating task. The information one is looking for, is hidden among thousands of documents returned by a search engine. One way of making search for relevant information easier, is to create better interfaces to the search engines; interfaces that facilitate quick and efficient browsing through the multitude of returned documents. In this paper, we present FIRE - a multimodal interface for information retrieval deployed in the Intelligent Room at the MIT AI Lab. FIRE differs from most other interfaces for information retrieval in that it combines a couple of interaction modalities to improve the search process.

1 Introduction

This paper presents the current state of our work on a new multimodal interface-situated in an Intelligent Environment-for retrieving information from the web. The work brings together progress made in three research areas: multi-modal interfaces, interfaces for information retrieval, and intelligent environments.

The motivation for building FIRE (the Friendly Information Retrieval Engine) was three fold: first, we wanted to build a very natural and effective information retrieval interface. Second, we wanted to demonstrate new capabilities that become possible when building

* This work was funded by Acer Inc., Delta Electronics Inc., HP Corp., NTT Inc., Nokia Research Center, and Philips Research under the MIT Project Oxygen partnership, and by DARPA through the Office of Naval Research under contract number N66001-99-2-891702.

applications situated in an Intelligent Environment (IE). Finally, we wanted to test the limitations of the technology we have developed for our IE.

FIRE takes advantage of the numerous display devices that many IEs offer and of the ubiquity of speech input and output in such spaces. Our current implementation of FIRE was build within the Intelligent Room Project [2] at the MIT AI Lab.

1.1 Problems with search engines

There are numerous problems with how the current search engines and Web directories organize the information [6]. Many of them stem from the current trend to assign each document to exactly one category. That makes it difficult to look for information that relates to several categories at once. Also, if one wants to browse a number of documents relating to a particular topic, one often needs to traverse a large number of sub trees in order to find all of the relevant information. This has to do with which nodes were chosen as top nodes in the category tree, and which were placed further down. For example, if we were to look for documents on the economy of European countries, it would really matter if the tree was organized like this:

$Economy \rightarrow Regional \rightarrow Europe \rightarrow Poland$

or like this:

$Regional \rightarrow Europe \rightarrow Poland \rightarrow Economy$

In the first case, we can just browse all documents under Europe and all of them will be somewhat relevant to our search. In the second case, if we look at all documents under Europe, we will get information about countries' geography, culture, customs, etc., as well as the economy.

We have designed FIRE in a way that allows browsing of relevant information returned by a search engine, even if the category tree had not been constructed in our favor, or if the topic of interest spans several dinstinct categories.

2 Related work

A number of approaches have been suggested to make searching large centralized corpora for relevant information easier. Three of the main trends are summarized here.

Preprocessing and annotating information. The START natural language query system employs this strategy to return the most relevant information in response to a query [7]. The strength of this approach is that it provides just the right information in response to a query. Its main weakness is that it requires a lot of human effort to set up and maintain.

Processing retrieved documents based on content. Documents returned by keyword-based search engines are analyzed based on their content and grouped according to some measure of similarity. The Scatter/Gather [4] algorithm is a prominent example of this strategy. The strength of this approach is that it allows browsing through collections of uncategorized documents. Unfortunatelly, the entire body of the documents needs to be analyzed thus drastically impacting the speed of the retrieval process.

Advanced visual interfaces, such as Cat-a-Cone [5], organize categorized collections of documents visually in a way that makes browsing and selecting the most relevant information easier. Many of such interfaces make it easy to explore several categories simultaneously and to provide instant access to a large portion of the information base at once, without cluttering the screen. Their shortcoming is that they provide no direct access to the information not presented on the screen. Also, they rely on documents being already categorized.

3 FIRE

The key goal of the work on FIRE is to create an interface that will provide a natural and efficient way of searching and browsing documents on the World Wide Web. FIRE is meant to use one or more of the existing search engines. It provides tools for easily identifying and selecting the most relevant search results from the hundreds or thousands returned by the search engine.

FIRE takes the visual interface approach (described in the previous section) one step further: although it still relies on search engines that categorize their search results, it provides a way to easily reach both visible and invisible search results. It also attempts to make browsing through the returned information easier by incorporating several modalities. Instead of using just a purely visual interface, FIRE combines several modalities: a multi-display graphical component, a pointing device, as well as speech input and output. What is more, FIRE is deployed in an IE, an immersive multimodal environment, where users interact multimodally not only with FIRE but also with the environ-

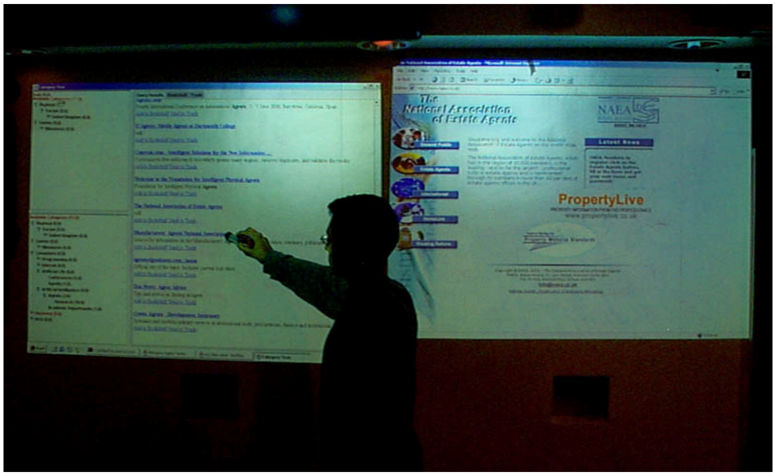

Fig. 1. FIRE in action: the left display shows the FIRE interface; the browser with the most recently selected document is displayed on the right.

ment itself, including devices (such as lights and projectors) and other software (e.g. the browser or the display manager).

FIRE makes search for information particularly effective when speech and gesture are used together to complement one another. It has been observed, however, that users rarely use all of the available modalities simultaneously but tend to pick one mode or switch between modes [8]. For that reason, FIRE is also perfectly usable if used just with a pointing device, or if driven solely by speech with visual feedback. Our initial tests indicate, however, that users familiar with the Intelligent Room find it easy and convenien to use both modalities at once, at least part of the time.

3.1 The interface

FIRE's interface has a number of graphical components interacting with one another. After the user makes a query, and information is retrieved from the Web, a *tree of all potentially relevant categories* is displayed. This tree is constructed based on what categories the documents returned by a search engine belonged to. Simple heuristics are applied to rank the categories in order of most likely relevance. The categories deemed as more relevant are displayed towards the top of the tree and the font size is proportional to the predicted relevance.

Another view shows the tree of *categories selected by the user*. Here the categories are ranked based on user's feedback ("it surely has to do with economy" vs. "it may have something to do with politics"). Our intention was to provide a space where user could see at a single glance all of the categories that he considers worth browsing though.

Another large component shows the *currently analyzed documents*. Whenever the user focuses on a set of categories, relevant documents are presented there. Each document is shown as a title and a short summary.

There is also the *local bookshelf* where the user can place relevant documents for short-term storage. Depending on the availability of resources and on user's preferences, the bookshelf can be placed on a separate display or together with the main part of the interface. The main display always has an icon where the user can place newly found documents to be moved onto the bookshelf.

The trashcan is the last component of the main part of the interface. As the name implies, the trashcan is a container for all discarded elements such as documents and categories. It was added relatively late in the development process. We have realized that sometimes users wanted to undo some of their operation after a relatively long time. Simple undo mechanism was not adequate in such situations. It became clear that we needed to give the users a way of verifying what items have been discarded.

Finally, there is a *browser*, used to show full text of the documents. The browser is almost always placed on its own display unless none is available.

We use two input modalities: speech and gesture. Either of them can be used alone to accomplish the task. However, each of them is better suited for some parts of the process than for the others.

Role of speech Speech in FIRE is used for four main tasks:

Taking shortcuts and probing invisible parts of the category tree and document base. For example, if a user asks about "agents", the system will display main categories such as Travel, Business, or Computers. The user, can quickly probe the system by asking "Do you have anything related to Artificial Intelligence?" If Artificial Intelligence is among the categories associated with any of the returned documents, the system will show the subset of documents about agents that are in that category and all of its subcategories.

Accessing multiple parts of the tree at once. As described in the example in Section 1.1, to view documents about the economy of European countries, the user may need to visit many separate branches in the category tree. Using FIRE, the user may very conveniently use speech to say “My query has to do only with Economy,” and *all* of the branches about economy will be presented and all other branches will be discarded.

Speech is also very useful for *command and control* part of the interaction. It can be used to undo actions or to manipulate the interface itself.

Finally, speech can often be used to make the *initial query*. In cases where the query includes uncommon terms, the user can easily fall back on a keyboard.

Role of gesture FIRE uses standard gestures such as selection or drag-and-drop. Its strength comes from incorporating novel input devices such as a laser pointer (whose position is tracked in real time with a camera), or an on-wall display with a specially instrumented electronic marker. Gestures are used to interact with the interface in a traditional GUI style, and to set context for spoken commands.

In particular, by using hand gestures the user can browse through the available categories and documents, and select documents for viewing and moving onto the bookshelf.

We are currently in the process of adding two new gestures: strike-through to delete (i.e. move to trash), and circling to select one or multiple objects.

4 Modalities: recognition and integration

FIRE uses relatively unsophisticated—yet effective—recognition methods for speech and gesture recognition. For gesture recognition we use primarily a pen-like pointing device, which can be used to interact accurately even with small objects on the screen. The two gestures we currently recognize (pointing and drag-and-drop) are unambiguious and easy to recognize. The two other that we are in the process of adding (strike-through and circling) are not as trivial but still easy to recognize correctly.

Our speech recognition system [3] is entirely grammar-driven. This ensures very good recognition rate and makes the processing of the spoken utterances straight-forward. Our choice of tools has made the implementation process easy at the expense of the “naturalness” of our interface. Hand and finger gestures would be preferred to pen strokes for pointing, and unrestricted speech recognition would eliminate the problem of user occasionally trying to use a phrase that is not in the grammars.

The benefits of our approach are very low recognition error rate and ease of development. Grammar-driven speech recognition engines make it easy to recognize and parse complex utterances. Thanks to this, our users can make statements like “My query has to do with Economy, Politics and Government but not with Culture or Travel” or “It has nothing to do with Artificial Intelligence but it might be relevant to Programming.” There are very few such hybrid constructs that we observed people using during their interactions with FIRE, and they are easy to describe within a grammar.

Because of the good recognition accuracy of our speech and gesture recognition systems, the integration of modalities is done at the post-recognition stage in FIRE, and the modalities do not cross-influence one another. In practice, therefore, multi modal integration in FIRE is restricted to the resolution of diactic references in utterances like “this category is not relevant,” “move this to the bookshelf,” or “put this there.” In the case of the last utterance, we need to resolve two references.

The context for resolving these references may be set by either speech (e.g. “What do you have under HCI” sets context to the HCI category) or gesture.

When we do the integration, we use temporal co-occurance and semantic compatibility to verify that the current context is relevant to the spoken command. If we cannot resolve what the user is referring to, we request clarification. For example, in case of the “put this there” command, if we cannot detect a valid destination for an object, we will ask the user “Where do you want me to put it?”

It is not to say, however, that the integration task has been made trivial. There are still cases that require some semantic analysis of recent events in order to establish how to resolve references best. For example, let us assume that the user drags a document to a trashcan. If the then says “put this there as well” while pointing at another document, “this” will be resolved to mean the new document and “there” to mean the trashcan. If instead she were to say “I actually meant to put it there” while pointing at the bookshelf, this time “this would refer to the document that was previously placed in the trashcan and “there” would mean the bookshelf.

5 Sample interaction

User says "I need information about agents." FIRE contacts a search engine and retrieves the results. It then displays a tree of all potential categories and a list of a few documents that appear most relevant. The top categories are Computers, Business and Travel. The user drags Computers onto the area with chosen categories. This sub-tree is expanded as deep as possible given available screen space (giving preference to those branches that are predicted to be more relevant). Again, most relevant documents are shown, this time only from the branch relevant to Computers. The user now asks "Do you have anything under HCI?" HCI is not visible on the screen but, indeed, under $Computers \rightarrow ArtificialIntelligence$ there is HCI. FIRE expands the right part of the tree and shows documents under HCI. The user selects some of them with a pointing device and they appear in the browser. Those that are particularily interesting, the user moves onto the bookshelf icon.

The user can now ask "Is there anything under Programming Languages?" FIRE replies that there is nothing but then the user notices that there is a branch called Programming under Computers. Selecting this branch with a pointer, reveals a number of documents about current agent programming tools. The user moves some of them onto the bookshelf. Saying "I am done" clears the main interface and brings up the bookshelf with all the documents placed there during the search. Now the user can evaluate the quality of the collected material and, potentially, save it for future reference.

6 Evaluation

Our initial informal experiments have shown that the itnerface is comfortable to use after a short initial training. Users were given a short explanation of the individual elements of the interface, and the extent of things they could use speech for. Our test users were members of the Intelligent Room project, already familiar with other multi-modal applications running in the Room. The users were particularily happy with the bookshelf, and with the ability to quickly browse the search results by category. Users have also commented favourably on having separate windows for browsing the returned results and for viewing the full documents.

On the negative side, users commented on our Spartan interface, and they found having two separate category trees unnatural, though were not able to suggest a different method of keeping track of selected categories while having access to all the rest of them. They also found the speech interface somewhat brittle. This particular concer we will address in the next setcion.

7 Futher work

We are currently working on a number of improvements to the system. Most significantly, we are in the process of incorporating new speech recognition software based on the Galaxy [9]. This engine is speaker independant and while it also works in a grammar-driven mode, it is much more flexible in that the grammars specify only the keywords and the general structure of the allowed utterances. In Galaxy, the grammar descriptions can contain wild-cards and thus allow for wider linguistic flexibility.

We are also in the process of integrating Haystack [1] with FIRE. Haystack is a personal information management tool. After integrating with FIRE, it will be able to answer questions like "When I was looking for information on agents yesterday, did I see anything about 007?"

Finally, we are developing an algorithm that will allow us to rebuild the category tree returned to us by the search engine in a way that best suits a particular search.

8 Contributions

FIRE demonstrates how the new potentials for human-computer interaction—that become available with the emergence of Intelligent Environments—can be used to build an effective and natural interface for information retrieval. IEs usually have more resources than a single desktop computer. If those resources become available, FIRE makes effective use of them. It uses up to three displays to separate navigation through information space from previewing retrieved documents. FIRE also benefits from the ubiquity of speech input and output in a smart space: while in such a space, the user does not have to make any special effort to start interacting with FIRE by means of speech, because all other compontents of the space use speech already. In comparison to purely visual interfaces, through the use of speech FIRE allows easy access to multiple parts of the category tree at once and makes it possible to take shortcuts to invisible parts of the information space.

References

1. E. Adar, D. Karger, and L. Stein. Haystack: Per-user information environments. In *Proceedings of the 1999 Conference on Information and Knowledge Management, CIKM*, 1999.
2. Michael Coen. Design principles for intelligent environments. In *Fifteenth National Conference on Artificial Intelligence (AAAI98)*, Madison, WI, 1998.
3. Michael Coen, Luke Weisman, Kavita Thomas, and Marion Groh. A context sensitive natural language modality for the Intelligent Room. In *Proceedings of MANSE'99*, Dublin, Ireland, 1999.
4. D. R. Cutting, J. O. Pedersen, D. Karger, and J. W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proc. of the 15th Int. ACM/SIGIR Conference*, 1992.
5. M. Hearst and C. Karadi. Cat-a-Cone: An interactive interface for specifying searches and viewing retrieval results using a large category hierarchy. In *Proc. of the 20th Int. ACM/SIGIR Conference*, Philadelphia, PA, 1997.
6. Marti A. Hearst. Interfaces for searching the web. *Scientific American*, March 1997.
7. Boris Katz. From sentence processing to information access on the world wide web. In *Proceedings of the Dans AAAI Spring Symposium on Natural Language Processing for the World Wide Web*, 1997.
8. Sharon L. Oviatt. Ten myths of multimodal interaction. *Communications of the ACM*, 42(11):74–81, November 1999.
9. S. Seneff, E. Hurley, R. Lau, C. Pao, P. Schmid, , and V. Zue. Galaxy-II: A reference architecture for conversational system development. In *Proceedings of ICSLP 98*, Sydney, Australia, November 1998.