ABSTRACT

We report on the Application Generator (AG), a system that automatically creates, and then manages, user-customized applications requiring a speech interface. The AG is composed of four modular components: the Automatic Dialogue Generator (ADG), the Profile Manager (PM), the Information and Services Manager (ISM), and the Dialogue Manager (DM). The PM module encodes the user’s intent with a representation of the services and information topics the user is interested in. Thus, the user defines a customized application; both in topical content and presentation format. The ISM module determines the available system resources by searching databases for the requested information and then delivers the information in a format defined by the user. A key feature of the ADG module is that it generates, in a uniform and consistent manner, a finite state dialogue for any task described by a set of tables. Finally, the DM module uses a set of Voice Interface Language (VIL) commands to carry out the actual dialogue session with the user.

1. INTRODUCTION

Our primary goal is to build a platform that generates a dialogue model by matching a user’s expectations with the system’s capability and available resources (see Figure 1). The system determines its available resources and the user encodes their intent in a profile. Both parties observe each other’s behavior during subsequent dialogue sessions. A dialogue session is carried out by means of a voice user interface (VUI), an effective means of human-machine communication for many applications. Typically, applications with a speech interface are “domain-specific”, where design engineers train the computer’s Dialogue Manager for narrowly defined tasks [1-3]. Examples have included weather reports, airline travel information systems, and banking services [4-6]. These predefined tasks are limited in that users have no control over the content and presentation format of the service.

Recently, users have been able to define personalized information services on the Internet [7], accessible using a graphical user interface (GUI). Our research has led us to investigate a similar platform that combines a speech interface with an Application Generator. The AG defines the dialogue flow between human and machine with a primary goal of providing the user with an interface to a variety of services that the user requests and defines. Therefore, the services offered by the agent will be created, organized and presented in a form that matches the user expectations and the system capabilities to deliver the content of database/service providers. This includes (but is not limited to) transaction services, messaging and information retrieval services. The user should be able to communicate using any of several methods including voice, keyboard, or mouse. We have concentrated on the keyboard and voice interface in the research discussed here.

The Application Generator platform consists of the following four software components:

- Profile Manager (PM)
- Information and Services Manager (ISM)
- Auto Dialogue Generator (ADG)
- Dialogue Manager (DM).

First, the PM uses a Q&A session to obtain knowledge about the user’s preferences, storing this in a user profile. Second, the ISM [8] accesses the available databases and services, including distributed sources such as the World Wide Web or corporate file systems. Then, the ADG [9] combines these data with the profile and builds a speech interface by generating a series of dialogue states, with associated grammars and system prompts. Finally, the DM [10] interfaces between the user and the dialogue specification generated by the ADG.
2. APPLICATION GENERATOR COMPONENTS

The four modular components that make up the AG platform are shown pictorially in Figure 2 and described more fully below. Each component plays a critical role in enabling the AG to match system capabilities and resources with user preferences and intents.

![Figure 2: The Application Generator platform that builds user defined applications.](image)

2.1. PM – Profile Manager

The PM generates and maintains a user profile, which encodes a model of the user, making it possible to build a personalized application for each user. The profile contains knowledge about the user’s intent, both in types of services desired, as well as in preferences for service organization and presentation. A question and answer session creates an initial user profile that includes such preferences as services, topics and attributes of interest, and presentation formats. The PM monitors subsequent dialogue sessions, tracking the user behavior and modifying the profile as needed.

The PM collects three main types of information about the user, a set of topics, subtopics, and keywords, as well as the desired presentation format. Examples of information services the user may be interested in include Email, Voicemail, News, and Sports. The ISM provides the PM with information about what system capabilities/services and information are available. The PM may request the ISM to perform some action on the Web on the user’s behalf, such as search for a new information service or register on a particular web site. The PM informs the user of the choices and allows the user to specify what services are desired. The user's application can then be modified in order to be more efficient, delivering the requested services and information more readily in an updated form, with a new dialogue flow.

2.2. ISM – Information Services Manager

In a networked environment, there is a vast number of services and amount of information available. The ISM uses its knowledge of the user, encoded in the user profile, to collect, prune, and organize the available information and services, creating a subset of services that the user needs [8]. The user profile specifies a list of desired services such as weather reports, email messages, or banking transactions. For those services that are an information service, a list of topics of interest is also included in the user profile. The ISM then extracts, from the Web, information potentially of interest to the user and stores it in a local user cache (or mailbox). This information extraction can occur continually in the background, overnight, periodically, on notification by another computer agent, or on the request of the user for updated information.

The ISM is responsible for organizing the information for presentation to the user. The ISM organizes the requested services and information in a hierarchical format for easy access and retrieval, sending this information to the ADG. This includes: summarization, indexing of paragraphs and sentences for browsing, and ranking each document according to the user’s set of keywords.

The ISM also provides the ADG module with lists of words to be used for creating language models for the Automatic Speech Recognizer (ASR) engine and prompts for the Text-To-Speech (TTS) engine.

2.3. ADG – Auto Dialog Generator

A unique feature of the ADG is that it automatically generates a finite-state dialogue for any task that is specified in tables [9]. The ADG combines information from the user’s profile with information from external sources to generate a dynamic dialogue flow, customized to the user. The ADG combines three sources of information: the user profile, information retrieved by the ISM, and a library that contains tables of information such as sub-grammars, search word clusters, and URL addresses. These three sources are then used to generate the user specific dialogue flow.

The dialogue flow specification is generated automatically and is encoded in a series of dialogue states that form a connected graph structure, organized in a set of subdirectories on the computer. Each dialogue state contains several specification files. These include: actions (actions to be performed), inputs (possible commands from the user), grammar (semantically acceptable phrases, vocabulary, and phonetic transcriptions), nextstates (set of next dialogue states), retarray (input synonyms), thes (text string for speech synthesizer), and variables (domain specific).

A major part of the ADG strategy is to define each dialogue state in a uniform and consistent manner. Therefore, library templates and keywords (supplied by the ISM) specify the various dialogue components. This includes Text-to-Speech
example, the user chose two subtopics, Baseball and Extreme Sports. A portion of the user profile is shown below. Each subtopic also includes a set of associated keywords, as shown.

**SAMPLE USER PROFILE**

<table>
<thead>
<tr>
<th>NAME_USER_FIRST</th>
<th>Andy</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME_USER_LAST</td>
<td>Pargellis</td>
</tr>
<tr>
<td>TOPIC</td>
<td>sports</td>
</tr>
<tr>
<td>SUBTOPIC</td>
<td>Babylon-five</td>
</tr>
<tr>
<td>KEYWORDS</td>
<td>McGwire, Dodgers, Yankees</td>
</tr>
<tr>
<td>SUBTOPIC</td>
<td>Extreme Sports</td>
</tr>
<tr>
<td>KEYWORDS</td>
<td>Daytona, Busch, World Cup</td>
</tr>
</tbody>
</table>

The ISM periodically searches external databases, such as websites, for documents containing one or more keywords from the user profile. The selected documents are then ranked and processed according to the user-defined presentation format.

The ADG then generates the connected graph structure along with associated dialogue flow. This is done using relevant library files and information (documents) supplied by the ISM. Next, the ADG generates the grammars associated with each dialogue state (such as Sports). In the case of information states that include downloaded documents, the grammar also contains important words found in the documents selected that day, for example, “Nikki Stone” or the “Boston Red Sox”. The user could then ask for a subset of documents containing any of those special words.

The DM is responsible for carrying out the dialogue session with the user. The application proceeds by first asking for the user’s name (“Entry” state). The user’s customized application is then retrieved, which includes the dialogue graph structure, associated dialogue specification components for each state, and documents that have been recently downloaded. The user may then ask for a particular service such as News or Sports. In the Sports service, the user queries the system by browsing the documents, shown in the following sample dialogue session:

**SAMPLE DIALOGUE SESSION**

SYS: Main menu. How may I help you? You may ask for Email, News, Sports, or Weather services. We describe the Sports service in more detail, as it is based on the dynamically created dialogue described above, with libraries designed specifically for this domain. We are currently expanding this into the Email and News domains.

The AG creates a dialogue session as follows. First, the PM generates a user profile, listing the user’s preferences in topic content as well as presentation format. In the following example, the user chose two subtopics, Baseball and Extreme Sports. A portion of the user profile is shown below. Each subtopic also includes a set of associated keywords, as shown.
4. RESULTS AND DISCUSSION

The main goal of this project is to match user expectations with system capabilities and then automatically generate applications with associated dialogue flows. This enables each user to define their own dialogue session, resulting in a dialogue interaction that is more accurate, faster, and rewarding. This task is accomplished by means of the four modules that make up the AG.

4.1. Major Advantages of the AG

The Application Generator discussed in this report has several key features lacking in systems that are domain dependent and/or generated by hand. They are:

- The DM uses VIL commands to form a domain independent platform with domain specific variables passed as arguments
- The PM allows a user to define personalized applications
- The ISM keeps track of system resources and downloads useful information to the user’s mailbox
- The ADG automatically generates a finite-state dialogue for any task description specified in tables
- Prompts and grammars are generated dynamically and in a consistent manner
- The application is organized visually as a directory tree
- New, user-specified applications, can be quickly generated

4.2. Future Research Topics

There are a number of very interesting areas of research that would enable the AG to create more effective dialogue sessions. First, some dialogue states might perform better if they were not predefined by an ADG but were dynamically generated during runtime. Context sensitive states such as Help states fall into this category. Second, a means of automatically determining characteristics of the user’s behavior, and then modifying the profile, would create a more dynamic service. Third, a document summarization module would be very useful where a long document is condensed into a concise summary, consisting of two or three short sentences. Fourth, it would be interesting to know what types of things the user might want to know more about, and then suggest these topics to the user.

5. SUMMARY

The Application Generator platform automatically builds dialogues that are customized for each user. A key feature is that a finite state dialogue can be defined, in a uniform and consistent fashion, for any task described by a set of tables. A profile manager encodes the user’s desires, enabling the user to specify a dialogue, both in topic content, as well as presentation. An information and services manager determines which available system resources are of interest to the user and then organizes the information in a manner defined by the user. A dialogue generator combines this information with the user’s profile to automatically generate the personalized application. Finally, a dialogue manager carries out the actual dialogue sessions using a set of domain independent VIL commands.

6. REFERENCES

7. Two popular websites allowing users to personalize their own webpages are www.mynetscape.com and www.pointcast.com.