Abstract

In this paper, A Multi-source intelligent DJ (Mi-DJ) service is introduced. It is an audio program platform that integrates different media types, including audio and text format content. It acts like a DJ who plays personalized audio program to user whenever and wherever users need. The audio program is automatically generated, comprising several audio clips; all of them are from either existing audio files or text information, such as e-mail, calendar, news or user-preferred article. Our unique program generation technology makes user feel like listening to a well-organized program, instead of several separated audio files. The program can be organized dynamically, which realizes context-aware service based on location, user's schedule, or other user preference. With appropriate data management, text processing and speech synthesis technologies, Mi-DJ can be applied to many application scenarios. For example, it can be applied in language learning and tour guide.

1. Introduction

In our daily life, “listening” is an important way to retrieve information. And in some circumstances it is the only way, or the most convenient way. For example, while driving, many people are used to listen to audio broadcast programs, to get the traffic information, daily news, or simply listening to some music. Another example is MP3 players which are widely used in many kinds of situations. People are getting more and more used to retrieve information from their MP3 players or mobile devices. And often they are “listening” instead of “watching”, when they jog, ride bicycle, or walk.

Based on such needs, if we can integrate the speech and language technology to organize the content of audio program, and use TTS to transform the text content into audio content, along with context aware technology on the mobile device that provides contents that meet users needs, we can provide users an intelligent personalized audio program service, which greatly facilitates users for listening audio information.

In this paper, we provide a Mi-DJ Service on mobile phone. Mi-DJ means “Mobile” “intelligent” DJ. It turns a mobile device into a personal virtual DJ by providing personalized audio program to user on a daily basis. The program comprises several audio clips; all of them are from either existing MP3 music files, user’s personal information content such as email, calendar events, or news article. If the content belongs to text-format, we use the text-to-speech to convert to speech audios.

Based on this service, users can either pre-arrange the content in the order they like by using the program definition language, or select pre-defined template for specific scenario. Mi-DJ Service provides intelligent service, which utilizes different contexts about the user, including location, schedule, user’s state (whether user is busy, away, or available), or user’s action (whether user is moving or stay in a fixed position) to provide various context aware services.

The remainder of the paper is organized as following: Section 2 introduces some related work. Section 3 describes system framework and how each component works. Section 4 provides two scenarios that we current deploy and in Section 5 provides a conclusion and future work.

2. Related Work

In this paper, we propose the system architecture for constructing personalized audio broadcast program.

2.1. Automatic broadcast program construction

The concept of automatic broadcast program construction was proposed before. Hayashi, M., et al [1][2] used APE (Automatic Production Engine) to generate TVML (TV program Making Language) scripts, and the TVML Player interpreted the TVML script and created the TV program in real time. With APE and the key information, making TV program is simplified. But so far the APE lack of capacity to deal with natural language, it requires labor work to make a TV program.

2.2. Speech synthesis

Mi-DJ uses a text-to-speech module to synthesize the textual content of clips. In Mi-DJ, we adopt HMM-based speech synthesis system (HTS) to synthesize the textual content. Recently, HTS has gain a lot of attention [7][8]. HTS has a good quality of synthesized voice, and due to compact language dependent module, memory size of HTS is relatively small, it is suitable for mobile device. Through speaker adaptation technology, HTS can easily change synthesized voice characteristics. Such feature allows us to get different voices more easily.

2.3. Context aware framework

The development of mobile devices hardware and software, and the widely deployed wireless network have enabled service providers to provide sophisticated and convenient mobile services. Many of the mobile services are related with multimedia service.

Those services not only need to support a wide variety of formats due to the diversity of mobile devices and their parameters, but also should be able to provide various type of content that satisfies users in different situation. Therefore a flexible and extensible framework is required. Wai Yip Lum and Francis C.M. Lau [3] designed a quality-of-service-aware
decision engine for the appropriate adaptation decision to generate an optimal content version. They used the quantitative methods to measure the QoS of the content versions and proposed a negotiation algorithm automatically to integrate the user preference, and context information, for example, device capabilities or network parameters, to determine a good scored content version. In [4], the MobileMM4U framework enabled personalized mobile multimedia presentations. It used static device parameters, user preferences and the current context of the user. This framework did not perform multimedia transcoding process, but used content selection approach to select best matching contents. Jannach, et al [5] proposed a knowledge-based approach. It used semantic web services technology as a tool for describing the semantics of adaptation services and constructing multi-step adaptation sequences. They showed results of the ISO/IEC MPEG core experiment on this framework. And their framework is now already contained in an (non-normative) amendment of the MPEG-21 DIA standard. In [6], Diana Weiß, et al introduced the middleware architecture to perform an automatic and distributed adaptation process, while the server provided a pre-processing of the content, the client completed the overall adaptation. Furthermore, the adaptation rule language was developed to support this task.

Those content adaptation frameworks used different context information as constraints to select the best matching content or perform transformation to achieve the generation of personalized multimedia content.

Our framework differs from the existing technologies in that TTS technology is used, to transform text content into audio content. By using high quality HMM-based TTS, we can make the program content more flexible. Thus users will feel like listening to a complete, well organized program, instead of a set of separates program clips.

### 3. System Framework

Mi-DJ converts textual content to audio clips and combines those with other existing audio files like music mp3 to provide user a personal audio program. Mi-DJ comprises several modules, including Data Integration Module, Content Extraction and Classification Module, Program Generator, Text-to-speech (TTS), Text Analysis Module, Context-aware Manager module. System framework is shown as Figure 1.

Data Integration Module collects all kind of content from different data sources, including unread mails, news, weather information, user preferred web content and domain specific content, like language learning materials or sight-seeing information for tour guide application. In addition to the text content, this module can also collect audio files from specified data sources.

The Content Extractor & Classifier extracts information from Data Integration Module, and gets appropriate material for program clips, therefore generating a set of program clips (the format of program clips can be either audio or text). And then the Program Generator (PG) organizes the playing sequence of the program clips, and adds appropriate pre-description and post-description for each program clips. The generation of pre-description and post-description requires Clip Content Processor (CCP) module to analyze existing program clips, to find the keyword or key feature about the program clips, in order to create appropriate description for each clip. The program generation process is shown in Figure 2. After the content is generated, TTS Module is used to transform the text into audio format. The Context aware manager (CAM) module is a rule base representation, it gathers the user’s context information including location, schedule, user’s state (whether user is busy, away, or available), user’s action (whether user is moving, or staying in a fixed position), and user’s preference. CAM will trigger related actions once the user context meets some conditions in of the rules in CAM. For example, when the CAM detects that the user has switched into other mode, the program rendering styles will change accordingly.

![MiDj Framework](image1)

The CCP module will have different process according to three clip types. The Music Commentor will deal with audio content, like mp3 file. The NoteMaker will handle general web content and the Reminder will copy with personal content. The CCP process is shown as Figure 3.

![Clip Content Processor](image2)

Figure 1: MiDj Framework

Figure 2: Program Generator

Figure 3: Clip Content Processor
The Music Commentor module first extracts the meta data of mp3 files from ID3 tags and uses that information, for example, singer’s name, title of the song or name of the album, to search for music comments from popular music web sites, and then get the most popular review post or summarize the review from multiple related posts. The sentence generation module uses the meta data to generate the introductory description text before the song is played. Finally, TTS is used to synthesize those generated text data in real time. With the introductory saying before the MP3 files are played, and summary of comments after played, the users will experience as if they are listening to a broadcast program with a virtual DJ, instead of merely listening to a MP3 file. Regarding to personal content, our Reminder module deals with unread emails and calendar. If a new mail has arrived, or there is an event in the calendar, this module will extract datetime, location data to generate appropriate alert to remind users about the new mail or the event in the calendar. As for web content, if the content belongs to domain specifics, the NoteMaker module will make various types of notes, including keywords, key phrases and significant sentences, even short summarization.

The program definition language module use to control audio program including clip sequence, types, description generation and rendering styles. Currently we build the PDL repository; there are different categories for example, the tour guide, language learning, etc. Each Category can have a number of PDL files and the user can update or create his own PDL files.

4. Service deployment

This framework can be applied in many situations which "listening to" information is needed. In this paper we describe two application scenarios we current deploy.

4.1. Tour Guide Application

The first scenario is the tour guide application. We have developed a mobile application called Mi-Guide. Mi-Guide provides real time audio information based on user’s location and preference. Mi-Guide is different from traditional tour-guiding application in that the content is organized dynamically, from different data sources.

![Figure 4: Mi-Guide Application. Different spots are shown in different icons. There are more than 20 spots in this campus.](image)

Mi-Guide was successfully deployed in a campus, to provide dynamic and personalized tour-guide service. Mi-Guide retrieves information from multiple data sources, including build-in information about the campus, MP3 files in user’s mobile device, news article from news web site, or user’s discussion from the forum web site. Text content is synthesized into speech in real time, using HMM based TTS which the space required is less than corpus based TTS, and the speech quality is promised. Figure 4 shows the example graph of the system.

4.1.1. Tour Guide Content

We have successfully deployed this application in a campus. There are about forty buildings in this campus, including research laboratories, restaurants, convenient stores, and some site seeing spots. In Mi-Guide, information of twenty major spots is included. For each spot, there is information that is build-in when the application is installed. Besides the build-in information, each spots may get information that will change by time from web site. For example, there may be exhibition in some building, or some promotions in the nearby convenient stores. Even the build-in information will be processed to be presented in appropriate length. Thus based on different user preference and the different paths that traveled by different users, each user will receive the tour guide content that is specifically designed for him/her.

4.1.2. Contexts and Intelligent Service

In the Mi-Guide, we provide three services, as shown in Figure 5. The service of Dynamically Adjust Content Length is will adapt the length of program clips according to the planed duration on that spot. The planed duration time is retrieved from users’ schedule. Or a default duration time is defined if user doesn’t specify the duration time for that spot.

Travel Information Reminder service: this service will check user’s schedule, to see if there is any special event happening in the spot that user is going to visit. It will also remind user to go to next spot based on user’s schedule about when to arrive and when to leave a specific spot.

Figure 5. Three Services With Different Contexts
unexpectedly good". And most of them mentioned the negative feedback about the stability of wireless network. Based their feedback, we are encouraged to apply Mi-Guide to another campus or exhibitions.

4.2. Language Learning Application

The second scenario is the language learning application. We developed a mobile application called Mi-Tutor. Mi-Tutor will analyze the learning material that user plans to learn, and give brief introduction about this material before the system start to “speak out”. After the material is read, Mi-Tutor will provide review of important sentence, sentence pattern, or vocabularies.

Besides analyzing official learning materials from text book or teachers, another key feature of My-Tutor is that it can recommend some interesting articles whose sentence patterns or grammars are similar to official learning materials. Those recommended articles may come from web sites that user is interested in. For example, from new web sites or movie, sports web sites. Mi-Tutor motivates user to learn by providing information that user is interested in.

Mi-Tutor can also detect if there is appropriate articles or information that related to user’s location or collect he information regarding the events happening in certain date in the history. The information is processed to become the learning material in real time, thus making it possible for learning language from the environment.

4.2.1. Application Scenario

Here is an example about how Mi-Tutor works.

It is 2008/10/31 6:00 A.M, now, and Anderson is sleeping. According to the Anderson’s PDL, Mi-Tutor downloads Anderson’s unread emails, news article about finance and sport, and gets the some articles about IT innovation, which is downloaded from popular web site. Finally Mi-Tutor completes the today’s program. The program is shown as Figure 6.

<table>
<thead>
<tr>
<th>Time</th>
<th>Program</th>
<th>Rendering</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>08:00-08:05</td>
<td>Today headline news</td>
<td>Audio</td>
<td>Driving</td>
</tr>
<tr>
<td>08:06-08:08</td>
<td>Weather report</td>
<td>Audio</td>
<td>Driving</td>
</tr>
<tr>
<td>08:12-08:15</td>
<td>Music</td>
<td>Audio</td>
<td>Driving</td>
</tr>
<tr>
<td>08:16-08:20</td>
<td>Today unread mail and events in schedule</td>
<td>Audio</td>
<td>Driving</td>
</tr>
<tr>
<td>08:29-09:00</td>
<td>Learning materials of IT innovation</td>
<td>Audio</td>
<td>Walking</td>
</tr>
<tr>
<td>08:36-09:00</td>
<td>Learning materials of IT innovation</td>
<td>Text</td>
<td>Read in the office</td>
</tr>
<tr>
<td>12:30</td>
<td>Reminder about visiting museum in Taipei at 14:00</td>
<td>Text, Vibrate</td>
<td>Lunch</td>
</tr>
<tr>
<td>12:40-13:00</td>
<td>Learning materials about TTS and Dialog systems</td>
<td>Audio</td>
<td>Driving</td>
</tr>
<tr>
<td>13:30</td>
<td>Learning materials about National Palace Museum</td>
<td>Audio</td>
<td>Driving</td>
</tr>
<tr>
<td>17:30</td>
<td>Learning materials about Halloween</td>
<td>Audio</td>
<td>Driving</td>
</tr>
<tr>
<td>19:00</td>
<td>Learning content review</td>
<td>Audio</td>
<td>Jogging</td>
</tr>
</tbody>
</table>

Figure 6: Anderson’s Program In 2008/10/31

On the way to office, Mi-Tutor starts to play the program. After playing headline news, weather report, emails and schedule event, Mi-Tutor starts to play articles about IT innovation. At 8:30, After Anderson gets into office, Mi-Tutor changes the rendering style as text reading mode. At 12:30 Mi-Tutor reminds him of a meeting at 14:00 in Taipei. On the way to Taipei, Mi-Tutor starts playing the remaining articles. When he passes the National Palace Museum, Mi-Tutor plays the introduction about National Palace Museum. On his way home, Mi-Tutor plays the story about origin of Halloween. Anderson suddenly finds that 10/31 is Halloween. At 19:00, when he is jogging in the park, Mi-Tutor plays the review of today’s learning content.

5. Conclusion and future work

In this paper, we described the Mi-DJ service framework. By combining TTS, language technology and context aware functionality, Mi-DJ integrates different media source and presents a complete program in audio form. We applied this framework to two applications. In the future, we will try to apply this framework to more audio related scenarios, and try to analyze patterns from users’ PDL files, in order to find program organizations that are more variable and suitable for users. To make voice of Mi-DJ more vivid, emotional TTS is considered. Furthermore we will try to utilize more human-computer interaction technologies, such as ASR and dialog system, to make Mi-DJ more anthropomorphic.

6. Acknowledgements

This research is a partial result of the Project 8352B24100 conducted by ITRI under sponsorship of the Ministry of Economic Affairs, Taiwan, R.O.C.

7. References

[1] Hayashi, “TVML(TV program Making Language) Make Your Own TV Programs on a PC!”, Virtual Studios And Virtual Production, 2000