A Spoken Dialogue System for Electronic Program Guide Information Access

Seokhwan Kim, Cheongjae Lee, Sangkeun Jung, and Gary Geunbae Lee

Department of Computer Science and Engineering
Pohang University of Science and Technology, South Korea
e-mail:(megaup, lcj80, hugman, gblee)@postech.ac.kr

Abstract—**In this paper, we present POSTECH Spoken Dialogue System for Electronic Program Guide Information Access (POSSDS-EPG). POSSDS-EPG consists of automatic speech recognizer, spoken language understanding, dialogue manager, system utterance generator, text-to-speech synthesizer, and EPG database manager. Each module is designed and implemented to make an effective and practical spoken dialogue system. In particular, in order to reflect the up-to-date EPG information which is updated frequently and periodically, we applied a web-mining technology to the EPG database manager, which builds the content database based on automatically extracted information from popular EPG websites. The automatically generated content database is used by other modules in the system for building their own resources. Evaluations show that our system performs EPG access task in high performance and can be managed with low cost.**

I. Introduction

During the past few years, digital broadcasting services have been widely used. One of the main advantages of digital broadcasting services is that the users can get more advanced information on broadcasting contents. The Electronic Program Guide (EPG) is the most widely used application of digital broadcasting services, and it provides detailed information on TV programs, such as program titles, channel names, genres, actors, program start times and so on. However the current EPG systems are still difficult to use for obtaining some information, even though they aim to provide user-friendly services. It is mainly due to the limitation of former interfaces, such as the remote controller which is used as the main input device in most of EPG systems.

In order to make an EPG system user friendly, we developed POSSDS-EPG, a spoken dialogue system for EPG information access. In the past few years, spoken dialogue systems have been used in various applications for natural and convenient interface with users [1], and recently, the interest in spoken dialogue systems has been sharply increasing. There have also been some studies on spoken language dialogue interfaces for operating a TV and retrieving information on TV programs [2].

POSSDS-EPG consists of several sub-modules which are general in most of spoken dialogue systems for other domains, such as automatic speech recognizing module (ASR), spoken language understanding module (SLU), dialogue managing module (DM), system utterance generating module (SUG) and text-to-speech synthesizing module (TTS). Besides these domain independent modules, we incorporated the EPG database managing module by using the web mining methodology. This is a necessary part of the system due to the characteristic of EPG data of being updated frequently and periodically. In order to constantly maintain up-to-date data, we developed a web mining module which automatically extracts data for building EPG database from websites dealing with information on TV programs. By using the automatic web mining module, we can reduce the cost and effort for consistently maintaining EPG database. In this paper, we describe the major methodologies behind POSSDS-EPG.

II. POSSDS-EPG: POSTECH Spoken Dialogue System For EPG Domain

POSSDS-EPG consists of a set of appropriate modules that are designed to be connected to each other according to the order. An overview of the system is shown in Fig 1. The overall system aims to output the synthesized spoken response corresponding to an input utterance spoken by the user.

In order to handle the spoken input utterance, the ASR module is operated first, which recognizes the user utterance. The recognized result is used for the input of the spoken language understanding module. The SLU module extracts semantic concepts from the user utterance and constructs the pre-defined semantic frame by using extracted results. The dialogue manager generates system responses with the semantic frame and the discourse history. The discourse history is a set of semantic frames in one dialogue session. The result of the dialogue manager is represented by corresponding system action tags selected from the pre-defined system ac-

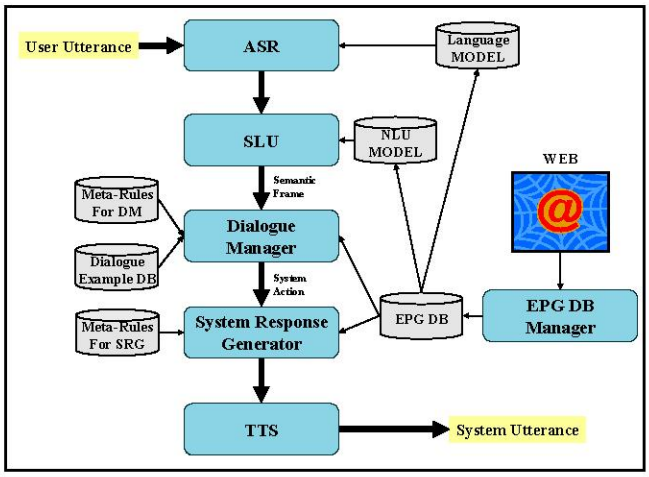

Fig.1 Overview of POSSDS-EPG System Architecture

tion tag set. Each system action tag is defined by considering the expected response of the system at each situation of the dialogue. The manner of the system response is determined by considering the system action tag in the system utterance generating module. The system utterance generating module produces literal system utterances by assembling retrieved records from the EPG database with system utterance templates which are determined by given system action tags. The final result of the overall system is produced by the TTS module, which synthesizes the spoken utterance from the literal utterance.

Each module in POSSDS-EPG is based on domain-independent methodologies, with the consequence that it can be easily applied to other domains. To build a spoken dialogue system for some domain, it is essential to construct the contents database for the domain, since all the modules need to take respective models and rules for the given domain, and they can be built from the contents database on the domain. In case of the EPG domain, we not only have to consider constructing the contents database initially, but also maintaining it continuously, due to the characteristic of the EPG contents of being updated frequently and periodically. Maintaining the contents database manually is time consuming, labor intensive, and expensive. To reduce the cost of the task, we developed the EPG database managing module based on the web mining technologies. It automatically extracts the up-to-date information from websites which provide information on TV programs. Extracted data are stored and managed in the form of the schema of the content database which can be directly used in other modules.

In following subsections, we describe detailed properties and methodologies of each module.

A. Automatic Speech Recognizer

The speech recognizer in POSSDS-EPG was developed based on HTK (Hidden Markov Model Toolkit). The recognizer uses a pre-trained dialogue acoustic model and adopts the EPG domain specific language model. To build the language model, the candidate utterances that have high probability of being spoken by users are required. We generate the candidate utterances automatically by using the dialogue examples in the existing example database and the retrieved result from the up-to-date EPG database. Table 1 shows an example of the automatically generated candidate utterances. Each dialogue example record in the example database not only has the utterance itself, but also the tagged information on the named entities contained in the utterance. The candidate utterances are generated by replacing each named entity in the existing utterance with the corresponding entity in the retrieved results. The candidate utterances are used for building the up-to-date language model for the speech recognizer.

Table.1 Examples of the automatically generated candidate utterances

An Existing Utterance
I want to watch **drama Hae-Sin** around **9 pm.** [genre = drama], [program_name = Hae-Sin], [time = 9 pm]
Retrieved Results
[genre = **movie**], [program_name = **Monster**], [time = **11 pm**] [genre = **sports**], [program_name = **Basketball**], [time = **7 pm**]
Candidate Utterances
I want to watch **movie Monster** around **11 pm.** I want to watch **sports Basketball** around **7 pm.**

B. Spoken Language Understanding

The SLU module of POSSDS-EPG was constructed by a concept spotting approach which aims to extract only the essential information for predefined meaning representation slots [3]. The semantic frame is made up of these slots including dialogue act, main action, and component slots for the EPG domain. An example of the semantic frame for the EPG domain is shown in Table 2.

Each slot value is selected from the corresponding predefined tag set shown in Table 3. While the dialog act tag represents the grammatical function of the utterance, the main action tag mainly comprises the semantic function of it. Due to the characteristics of the dialog act and the main action tag, each dialog act tag deals with domain-independent concepts, while each of the main action tag is defined for the domain-specific task of the dialog manager for the EPG domain. The component slots are used for representing named entities in the utterance.

We regarded the SLU problem as a classification problem,

Table.2 An example of the semantic frame

User Utterance	그럼 SBS 드라마는 언제 하지? (Geu-leom SBS deu-la-ma-neun eon-je ha-ji?) Then, when do the SBS dramas start?
Dialog Act	When-question
Main Action	Search_start_time
Component Slots	[channel = SBS, genre = 드라마]

Table.3 The list of the predefined slots in the semantic frame for the EPG domain

Dialog act		
Wh_Question	Yes_No_Question	Accept
Reject	Statement	Request
Express	Conventional_Open	Conventional_Close
Main Action		
Search_program	Search_channel	Search_day
Search_starttime	Search_endtime	Search_currenttime
Search_currentdate	Move_channel	Alarm
Record	TV_on	TV_off
Component slot		
Genre	Channel	Date
Start_time	End_time	Cast
Day	Program	

which can be solved by statistical machine learning frameworks. To build a statistical model for the SLU problem, we should prepare the training corpus containing utterances that have high probability of being spoken by users. We can easily create a training corpus by reusing the candidate utterances that are used for building the language model in the speech recognizer, which is referenced in the previous subsection.

C. Dialogue Manager

To develop an effective and practical spoken dialogue system, we proposed the situation-based dialogue management method using dialogue examples in our previous work [4]. Fig 2 illustrates an overall strategy of the example-based dialogue modeling. For the system utterance generation, we automatically construct and index a dialogue example database from the dialogue corpus. The dialogue manager retrieves the best dialogue example for the current dialogue situation, which includes a current user utterance, semantic frame and discourse history. From the retrieved result, the dialogue manager determines the system action tag from the pre-defined tag set shown in the Table 4.

D. System Utterance Generator

The system utterance generator generates the literal system utterances based on the system action tag and the utterance generating template. Each system action tag has at least one utterance generating template which is constructed manually. The system utterance generating task is advanced by filling slots in the template with proper values, such as retrieving results from the EPG database, slot values in the semantic frame, and constituents in the discourse history. Table 5 shows an example of this task.

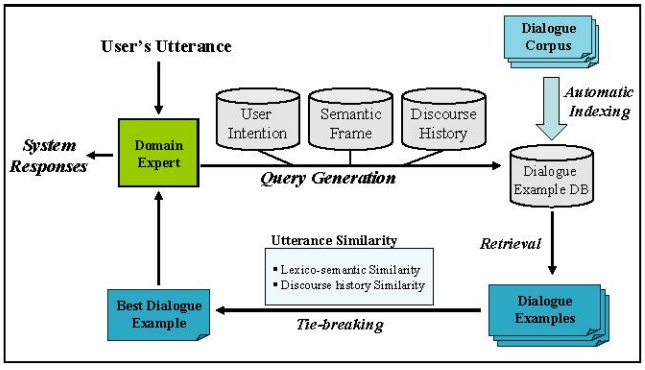

Fig.2 Example-based Dialogue Modeling Strategy

Table.4 The list of the predefined system action tags

System Action		
Salutation	Say	Select
Inform_channel	Inform_program	Inform_cast
Inform_alarm	Inform_number	Inform_record
Inform_time	Inform_positive	Inform_negative
Specify	Confirm	Finish

Table.5 An example of the system utterance generation

System Action Tag	Inform_Channel
Utterance Template	[program_name]은 [channel]에서 합니다. ([program_name] eun [channel] e-seo hap-ni-da) [program_name] is broadcasted on [channel].
Slot Values	[program_name = 해신, channel = KBS]
System Utterance	**해신은** KBS에서 합니다. (Hae-Sin eun KBS e-seo hap-ni-da) Hae-Sin is broadcasted on KBS.

E. EPG Database Manager

The main purpose of the EPG database manager is to build a content database for the other modules in POSSDS-EPG with minimal human effort. Fortunately, there are several websites which provide the information on TV programs, and we can obtain well-formed data for EPG from these websites. We chose an EPG website (http://www.epg.co.kr) dealing with the information on Korean TV programs. The EPG database manager builds a contents database from the information on the website following the procedure shown in Fig 3. Firstly, unnecessary parts of the web page are filtered out on the content filtering step. Since we are mainly concerned on the information in the form of the table, several tables are distilled through the first step. Then, the EPG database manager extracts the needed information from the tables referring to the structure and the header information of the given table. The extracted data are stored in content database for the POSSDS-EPG on the last step. All the steps in the EPG database manager are currently implemented by using the website specific wrapper rules.

III. EVALUATIONS AND IMPLEMENTATION

A. Automatic Speech Recognizer

In order to evaluate the effectiveness of the automatically managed spoken dialogue system, we measured the user satisfaction of the system and compared it with our previous system which is manually tuned for the fixed EPG database. The user satisfaction was defined with the linear interpolation of three different measures: user perception of Task Completion Rate (TCR), Mean Recognition Accuracy (MRA), and Success Turn Rate (STR) instead of the elapsed time.

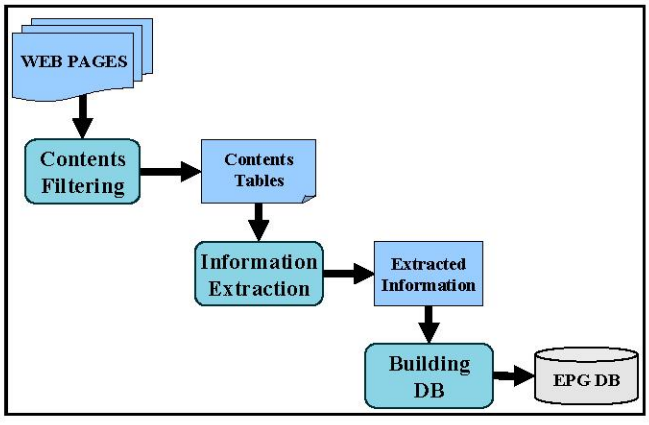

Fig.3 A procedure of web-mining-based EPG content database building

Each was weighted by a factor of 1/3, so that the maximum value of the user satisfaction is one. We asked each of 5 test volunteers to perform 5 different EPG tasks. The volunteers evaluated every system's response in each dialogue turn.

As we can see in Table 6, the user satisfaction of the new system which is automatically managed was 0.73 for spoken input, while the previous system achieved a user satisfaction of 0.75 for spoken input. It means that we can automatically manage the POSSDS-EPG, which has a close performance to the manually built system with much less human effort.

B. Implementation

We implemented POSSDS-EPG to be separated into three units of programs including the engine part, the graphical user interface (GUI) part, and the information feeding part. The engine part contains the above-mentioned modules except to the EPG database manager. The engine part was developed by using standard C++ library and can run under both Linux and Windows platform, but the GUI part of the system can only be operated under the Windows platform, because it is implemented by using Microsoft Visual Studio's MFC library. Fig 4 shows a screen shot of the POSSDM-EPG, which is displayed through the GUI part of the system on the Windows platform. While the engine part and the GUI part are operated tightly coupled, the information feeding part is independent from the other parts of the system. The information feeding part which incorporates the EPG database manager module runs at offline to automatically build a content database.

IV. CONCLUSION

This paper describes the POSSDS-EPG, a spoken dialogue system for EPG information access. To make an effective and practical spoken dialogue system, we incorporated suitable methodologies for each module, such as ASR, SLU, DM, and SUG. In addition to these modules, we implemented the EPG database manager which automatically builds the content database for the other modules in the system by using the web-mining methodologies. The automatically generated content database is used by each module to build its own resources. The evaluation result demonstrates that the EPG database manager can dramatically reduce the cost of constructing and maintaining our system. Currently the EPG database manager was implemented using site-specific wrapper rules, but we will reform the system to be operated without regarding the type of the target website in order to collect wider information from more diverse websites.

Table 6 The comparison of the dialogue performances between the manually built EPG database and the automatically built one

Evaluation	Manually Managed System	Automatically Managed System
TCR	0.76	0.72
STR	0.65	0.62
MRA	0.85	0.85
User Satisfaction	**0.75**	**0.73**

TCR: User Perception of Task Completion Rate
STR: Success Turn Rate
MRA: Mean Recognition Accuracy
User Satisfaction = aTCR + bSTR + rMRA

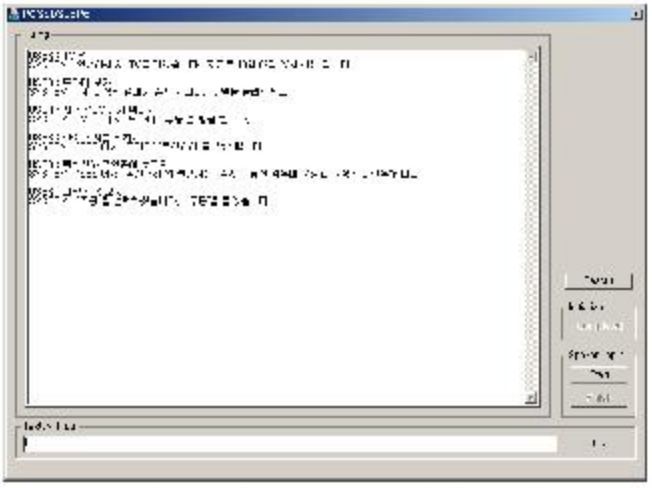

Fig.4 A screen shot of the POSSDM-EPG operated on the Windows platform

V. ACKNOWLEDGEMENT

This research was supported by the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Commerce, Industry and Energy of Korea, and also supported by the Invitation for Proposals on Internet Services Theme funded by Microsoft Research Asia.

VI. REFERENCES

[1] J. Allen, D. Byron, M. Dzikovska, G. Ferguson, L. Galescu, and A. Stent, "Toward conversational human-computer interaction," AI Magazine, vol. 22, no. 4, 2001, pp 27-37,.

[2] J. Goto, M. Miyazaki, and T. Kobayakawa, "A TV Agent System that Integrates Knowledge and Answers Users' Questions" Proceedings of the 2006 International Conference on Intelligent User Interfaces, pp 300-302, 2006.

[3] J. Eun, C. Lee, and G. Lee, "An Information extraction approach for spoken language understanding" in Proceedings of the 8th International Conference on Spoken Language Processing, 2004, pp. 2145-2148.

[4] C. Lee, S. Jung, J. Eun, M. Jeong and G. Lee, "A Situation-based Dialogue Management Using Dialogue Examples" in Proceedings of the international conference on acoustics, speech and signal processing 2006, vol. 1, 2006, pp. 69-72.