Development of Slovak GALAXY/VoiceXML Based Spoken Language Dialogue System to Retrieve Information from the Internet

Jozef Juhar[1], Stanislav Ondas[1], Anton Cizmar[1], Milan Rusko[2], Gregor Rozinaj[3], Roman Jarina[4]

[1]Department of Electronics and Multimedia Communications
Technical University of Košice, Košice, Slovakia
Jozef.Juhar@tuke.sk, Anton.Cizmar@tuke.sk, Stanislav.Ondas@tuke.sk

[2]Institute of Informatics, Slovak Academy of Science, Bratislava, Slovakia
Milan.Rusko@savba.sk

[3]Department of Telecommunications, Slovak University of Technology, Bratislava, Slovakia
Gregor.Rozinaj@stuba.sk

[4]Department of Telecommunications, University of Žilina, Žilina, Slovakia
jarina@fel.utc.sk

Abstract

In this paper we describe the research and development of the first Slovak spoken language dialogue system. The dialogue system is based on the DARPA Communicator architecture. The proposed system consists of the Galaxy hub and telephony, automatic speech recognition, text-to-speech, backend, transport and VoiceXML dialogue management modules. The SLDS enables multi-user interaction in the Slovak language. The funcionality of the SLDS is demonstrated and tested via two pilot applications, „Weather forecast for Slovakia" and „Timetable of Slovak Railways". The required information is retrieved from Internet resources in multi-user mode through PSTN, ISDN, GSM and/or VoIP network.

Index Terms: dialogue system, Galaxy, VoiceXML, MobilDat

1. Introduction

Due to the progress in the technology of speech recognition and understanding, the spoken language dialogue systems (SLDS) has begun to emerge as a practical alternative for a conversational computer interface. They are more effective than an IVR systems since they allow a more free and natural interaction and can be combined with the input modalities and visual output.

The above statement is true for many languages around the world, not just Slovak. In this paper we describe the development of the first SLDS which is able to interact in the Slovak language. The system has been developed in the period from July 2003 to June 2006 and is supported by the National programme for R&D "Building of the information society". The main goal of the project is in the research and development of a SLDS for information retrieval using voice interaction between humans and computers. The SLDS has to enable multi-user interaction in the Slovak language through telecommunication networks to find information distributed in computer data networks such as the Internet. The SLDS will also be a tool for starting research in the area of native language teachnologies in Slovakia.

Contractors of the project are the Ministry of Education of the Slovak Republic and the Technical University of Košice. Collaborative organizations are the Institute of Informatics, the Slovak Academy of Sciences Bratislava, the Slovak University of Technology in Bratislava and the University of Žilina.

The choice of a solution has come from contemporary free resources, state-of-the-art in the topic and the experiences of the partners involved in the project. As described further the solution is based on the DARPA Communicator architecture based on the Galaxy hub, a software router developed by the Spoken Language Systems group at MIT [12], subsequently released as an open source package in collaboration with the MITRE Corporation, and now available on SourceForge [13]. The proposed system consists of the Galaxy hub and six modules (servers). Funcionality of the SDS is demonstrated and tested via two pilot applications – „Weather forecast for Slovakia" and „Timetable of Slovak Railways" retrieving the required information from internet resources in multi-user mode through telephone.

2. System architecture

The architecture of the developped system is based on the DARPA Communicator [12], [13]. The DARPA Communicator systems use a 'hub-and-spoke' architecture: each module seeks services from and provides services to the other modules by communicating with them through a central software router, the Galaxy hub. Java, along with C and C++ supported in the API to the Galaxy hub. The substantial development based on the Communicator architecture has been already undertaken at Carnegie Mellon University and the University of Colorado [14],[15].

Our system consists of a hub and six system modules: telephony module, automatic speech recognition (ASR) module, text-to-speech (TTS) module, transport module, back/end module and module of dialogue management. The relationship between the dialogue manager, the Galaxy hub, and the other system modules is represented schematically in Fig. 1.

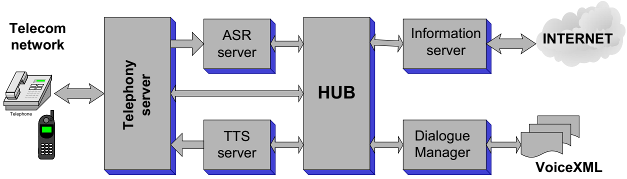

Fig. 1. *Architecture of the Galaxy/VoiceXML based spoken Slovak dialogue system*

The telephony module connects the whole system to a telecommunication network. It opens and closes telephone calls and through the/a Broker Channel transmits speech data to/from the ASR/TTS modules. The server of automatic speech recognition (ASR) performs the conversion of incoming speech to a corresponding text.A context dependent HMM acoustic models trained on SpeechDat-Sk and MobilDat-Sk speech databases and ATK/HTK and Sphinx IV based speech recognition engines were used in this task. The dialogue manager controls the dialogue of the system with the user and performs other specified tasks. The heart of the dialogue manger is as the interpreter of VoiceXML mark-up language. The information server connects the system to information sources and retrieves information requiered by the user. The server of text-to-speech (TTS) synthesis converts outgoing information in text form to speech, which is more user friendly.

The communicator supports „Windows-only" as well as a mixed Windows/Linux platform solution. In this case a Transport Server, managing file transmissions between platforms is active.

3. MobilDat-SK speech database

The MobilDat-SK is a speech database containing recordings of 1100 speakers recorded over a mobile (GSM) telephone network. It serves as an extension to the SpeechDat-E Slovak database [6] and so was designed to follow the SpeechDat specification [5] as closely as possible. It is balanced according to the age, regional accent, and sex of the speakers. Every speaker pronounced 50 files (either prompted or spontaneous) containing numbers, names, dates, money amounts, embedded command words, geographical names, phonetically balanced words, phonetically balanced sentences, Yes/No answers and one longer non-mandatory spontaneous utterance.

Every speaker called only once; from one acoustic environment. The required number of the calls from different environments were specified as a minimum 10 % of the database for each environment. The home, office, public building, street and vehicle acoustic environments were chosen.

We decided to use the database content adopted from SpeechDat-E database, however according to assumed practical applications some new items were added:

- O4 - sentence expressing a query on departure or arrival of a train, including names of two train stations from a set of 500.
- O6 - name of the town or tourist area from a set of 500 names.
- O9 – web domain or e-mail address from a set of 150 web domains and 150 e-mail addresses
- R1 - One non-mandatory item – a longer spontaneous utterance was added at the end of the recording. The caller had to answer a simple question from a set of 25 such as: "How do you get from your house to the post-office?". This item should considerably extend the spontaneous content of the database.

4. Automatic speech recognition

The development of reliable and fast speech recognizer is not an easy task. Fortunately there are several speech recognizers available for nonprofit research. We have adapted two well-known speech recognizers as the ASR module for our system. The first is ATK, on-line version of HTK [3]. The ATK based ASR module was adapted for our SDS running on a Windows-only platform and on a mixed Windows/Linux platform as well. In the second case the ASR module runs on a separate PC with Linux OS. The second speech recognizer we adapted for our system is Sphinx-4 written in Java [2], [10]. Both ASR modules provide similar results.

SpeechDat-SK [6] and MobildDat-SK [7] databases were used for traning HMM's. Context dependent (triphone) acoustic models were trained in a training procedure compatibile with "refrec" [1], [4]. Dynamic speech recognition grammars and lexicons are used in the speech recognizers.

5. Text-to-speech synthesis

Two TTS modules has been designed using two different approaches – diphone and corpus based synthesis.

5.1. The diphone concatenative synthesizer

This speech synthesizer is based on concatenation of small elements of a pre-recorded speech signal, mainly diphones. An original algorithm similar to the Time Domain Pitch Synchronous Overlap and Add (TD-PSOLA) was used for concatenation.

The pronunciation is controlled by a block of orthographical-to-orthoepical (grapheme to phoneme) conversion based on a sophisticated set of rules supplemented by a pronunciation vocabulary and a list of exceptions. This elaborated unit has proven to be more reliable than our similar data driven system based on CART trees [8].

5.2. Corpus synthesizer

The second method is based on corpus synthesis or concatenation of prepared acoustic units. Concatenated units may be of uniform or non-uniform length. The advantage of corpus-based method is in minimizing the number of concatenations in synthesized speech and thus reducing the need for speech processing causing artificiality.

The most critical phase of corpus synthesis is the selection of appropriate units. Two cost functions are used for evaluating the optimal unit sequence: target cost defined between the desired and each candidate unit in the database and concatenation cost defined between each pair of candidate units in the database. The desired unit – target may be defined by various parameters characterizing its prosodic and phonetic features. Since these features also define units in the database, cost functions are computed as certain distances between these parameters. Apparently, the final selected unit sequence is determined by its low values of the costs. However, there are a couple of ways how to take the costs into account. The optimal one is to select those units that minimize this equation:

$$C\left(t_i^n, u_i^n\right) = \sum_{i=1}^{n} C^t\left(t_i, u_i\right) + \sum_{i=2}^{n} C^c\left(u_{i-1}, u_i\right) \qquad (1)$$

where n stands for number of units in the sequence, u_i for i-th unit in the sequence, t_i for i-th target, and C^t and C^C for target and concatenation cost respectively. Each cost is represented by several subcosts that contribute to the final cost by different measures and so these subcosts must be weighted. The minimum sums in (1) are then effectively computed by a Viterbi search.

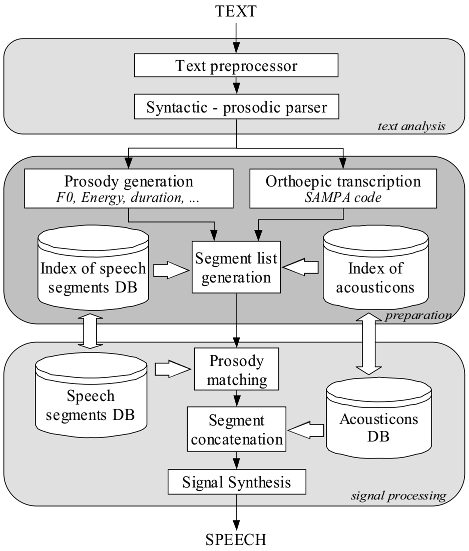

Fig.2 Schematic diagram of the diphone concatenative synthesizer

6. Dialogue manager

There are many approaches of how to solve dialogue manager unit and and also many languages for writing a code for it. Voice Extensible Markup Language (VoiceXML) is a markup language for creating voice user interfaces that use automatic speech recognition (ASR) and text-to-speech synthesis (TTS). Many commercial VoiceXML-based speech applications have been deployed across a diverse set of industries, including financial services, government, insurance, retail, telecommunications, transportation, travel and hospitality. VoiceXML simplifies speech application development, enables distributed application design and accelerates the development of interactive voice response (IVR) environments. For these reasons, VoiceXML has been widely adopted within the speech industry, and for these reasons we decided that the dialogue manager unit be based on VoiceXML interpretation.

Fig. 3 shows the main components of our dialogue manager based on the VoiceXML interpreter [11]. Its fundamental components are VoiceXML interpreter, XML Parser and ECMAScript unit.

The dialogue manager is written in C++. We have started from scratch and in its actual state the interpreter performs all fundamental algorithms of VoiceXML language and service functions for all VoiceXML commands, i.e. Form Interpretation Algorithm, Event Handling, Grammar Activation and Resource Fetching Algorithms. It supports the full range of VoiceXML 1.0 functions. The goal is full support of VoiceXML v2.0.

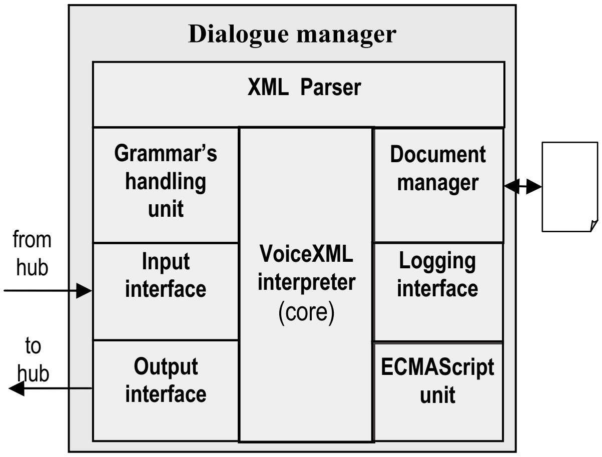

Fig.3 Basic components of VoiceXML based dialogue management unit

7. Telephony module

The telephony (audio) server connects the whole system to a telecommunication network. A direct (broker) connection between audio server and ASR server or TTS server is established to transmit speech data and this way reduces the network load. The telephony server is written in C++ and supports telephone hardware (Dialogic D120/41JCT-LSEuro). The ongoing work will be concentrated on the support of barge-in, DTMF and call-routing.

8. BackEnd module

After an analysis of various existing approaches of information retrieval from the web, and of the task to be carried out by the information server in our pilot applications, we came to the decision that no complicated data retrieval system is needed. On the contrary – a rule based ad-hoc application searching only several predefined web-servers with a relatively well known structure of pages will do a much better job. As the number of web-servers giving detailed weather forecast for Slovakia, as well as the number of web-servers providing information on train and bus connections in Slovakia is very limited, we had been checking several selected servers for stability and information reliability for about a month, and then we chose several candidate servers, from which the information is to be retrieved.

The information server (backend server) is capable of retrieving the information contained on the suitable web-pages according to the Dialogue Manager (DM) requests, extracting the needed data, analyzing it and if they are taken for valid, returning the data in the XML format to the DM. If the backend server fails to get valid data from one web source, it switches to a second wrapper retrieving the information from a different web-server.

The information server communicates with the HUB via the GALAXY interface. This module accomplishes its own communication with HUB, receives input requests, processes them and makes decisions to which web-wrapper (WW) the request should be sent, receives the answer and sends it back to the HUB.

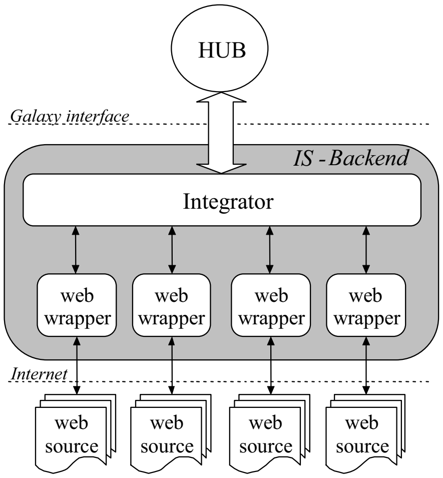

Fig.4 Progress of performance for consequently trained models

The web-wrapper is responsible for the navigation through the web-server, data extraction from the web-pages and their mapping on to a structured format (XML), convenient for further processing. The wrapper is specially designed for one source of data; thus to combine data from different sources, several wrappers must be designed.

Wrappers are designed to be as robust as possible against changes in the web-pages structure . Nevertheless, in the case of substantial changes in the web-page design, the adaptation of the wrapper would probably be inevitable.

To speed up the system (to eliminate the influence of long reaction times of the www-pages) and to assure drop-out resistance while simultaneously keeping the information as current as possible, automatic periodic download and cashing of the web-pages content were introduced.

The server is open for future applications by the possibility of creating web-wrappers for new services and adding them to the existing wrappers. The information on the currently accessible wrappers is stored in the system's configuration file.

9. Conclusions

In this paper we have described the development of the first Slovak spoken language dialogue system. Our main goal was to develop a dialogue system that will serve as a starting platform for further research in the area of spoken Slovak engineering. We successfully combined up to date free resources with our own research into functional system, that enables in multi-user mode an interaction through telephone in the Slovak language to retrieve required information from Internet resources. The funcionality of the SLDS is demonstrated and tested via two pilot applications, „Weather forecast for Slovakia“ and „Timetable of Slovak Railways“ [9]. Applying new findings we are continuing in further developments and improvements to the system.

10. Acknowledgements

This work was supported by the Slovak Grant Agency VEGA under grants No. 1/1057/04, 1/3110/06 and the Ministry of Education of the Slovak Republic under research project No. 2003 SP 20 028 01 03.

11. References

[1] Lindberg, B., Johansen, F. T., Warakagoda, N., Lehtinen, G., Kačič, Z., Žgank, A., Elenius, K., Salvi, G., *“A noise robust multilingual reference recognizer based on SpeechDat (II)”*, in Proc. ICSLP 2000, Beijing, China, 2000.

[2] Lamere, P., Kwok, P., Walker, W., Gouvea, E., Singh, R., Raj, B., Wolf, P., “*Design of the CMU Sphinx-4 decoder*”, in Proc. Eurospeech 2003, Geneve, Switzerland, September 2003, pp. 1181–1184

[3] Young, S., “*ATK: An application Toolkit for HTK, version 1.3*”, Cambridge University, January 2004.

[4] Lihan, S., Juhár, J., Čižmár, A., “*Crosslingual and Bilingual Speech Recognition with Slovak and Czech SpeechDat-E Databases”,* In Proc. Interspeech 2005, Lisabon, Portugal, September 2005, pp. 225 – 228.

[5] Winski R., “*Definition of corpus, scripts and standards for fixed networks*”, Technical report, SpeechDat-II, Deliverable SD1.1.1., workpackage WP1, January 1997.

[6] Pollak, P., Cernocky, J., Boudy, J., Choukri, K., Rusko, M., Trnka, M. et al. “*SpeechDat(E) „Eastern European Telephone Speech Databases*“, in Proc. LREC 2000 Satellite workshop XLDB - Very large Telephone Speech Databases, Athens, Greece, May 2000, pp. 20-25.

[7] Rusko, M., Trnka, M., Darjaa S., “MobilDat-SK - A Mobile Telephone Extension to the SpeechDat-E SK Telephone Speech Database in Slovak”, accepted for SPEECOM 2006, Sankt Peterburg, Russia, July 2006.

[8] Darjaa, S., Rusko, M., Trnka M., “*Three Generations of Speech Synthesis Systems in Slovakia*”, accepted for SPEECOM 2006, Sankt Peterburg, Russia, July 2006.

[9] Gladišová, I., Doboš, Ľ., Juhár, J., Ondáš, S., “*Dialog Design for Telephone Based Meteorological Information System*”, in Proc. DSP-MCOM 2005, Košice, Slovakia, Sept., 2005, pp. 151-154.

[10] Mirilovič, M., Lihan, S., Juhár, J., Čižmár,A., “*Slovak speech recognition based on Sphinx-4 and SpeechDat-SK”,* in Proc. DSP-MCOM 2005, Košice, Slovakia, Sept. 2005, pp. 76-79.

[11] Ondáš, S., Juhár, J., “*Dialogue manager based on the VoiceXML interpreter*”, in Proc. DSP-MCOM 2005, Košice, Slovakia, Sept. 2005, pp.80-83.

[12] http://www.sls.csail.mit.edu/sls/technologies/galaxy.shtml.

[13] http://communicator.sourceforge.net/

[14] http://fife.speech.cs.cmu.edu/Communicator/index.html

[15] http://communicator.colorado.edu/