LINGVO/LASER: Prototyping Concept of Dialogue Information System with Spreading Knowledge

Kamil Ekštein and Tomáš Pavelka

Laboratory of Intelligent Communication Systems,
Dept. of Computer Science and Engineering,
University of West Bohemia,
Univerzitní 22, 306 00 Plzeň, Czech Republic

Abstract. The presented paper introduces a scalable modular prototyping concept and methodology framework for rapid development of domain-oriented dialogue information systems being developed at Laboratory of Intelligent Communication Systems (LICS). The whole system consists of two practically standalone aggregates of software modules and knowledge bases: A speech recognition and synthesis software—LASER, and a generic task-independent scalable dialogue information system—LINGVO. The article describes the architecture, design, and philosophy of the framework in details and discusses its benefits already experienced while used to build a voice-driven dialogue application.

1 Introduction

In these days the speech and dialogue research facilities including universities, specialized private research companies, as well as market leaders can offer various kinds of speech-enabling software. Such systems usually make it possible to build voice-driven (mostly telephone operated) dialogue information systems that provide their prospective users with certain information from preselected domains, such as e.g. arrivals and departures of trains, planes, public transportation vehicles at all, opening hours of some institutions, exchange rates, telephone numbers, etc. The wider (containing more expressions) the operated domain is, the more accurate the system must be to successfully satisfy requests of its users. As the overall recognition accuracy of even the state-of-the-art systems is still far from being sufficient to work on wide domains, the designers must back up on highly restrictive techniques giving the user in fact no chance to mislead the system—the dialogue then might look like the following ("S" is the system, "U" is the user):

S: Hello, this is ARS—Airplane-ticket Reservation System.
Where would you like to fly? Please, state the city.
U: *Porto.*
S: Do you mean Porto in Portugal? Please, reply "Yes" or "No".
U: *Yes.*
S: When would you like to fly? Please, state the date only.
U: *Twelfth of April.*

Ekštein K. and Pavelka T. (2004).
LINGVO/LASER: Prototyping Concept of Dialogue Information System with Spreading Knowledge.
In *Proceedings of the 1st International Workshop on Natural Language Understanding and Cognitive Science*, pages 159-168
DOI: 10.5220/0002665301590168
Copyright © SciTePress

S: What class would you like to fly? Please, reply “Business” or “Economy”.
U: *Economy.*
S: What is your credit card number? Please, read the number figure by figure.
U: *One, two, three, four, five, six.*
S: A seat in economy class is booked for you on 12^{th} April, flight to Porto, Portugal.
Your credit card number is 123456. Is it correct? Please, reply “Yes” or “No”.
U: *Yes.*
S: Okay. Thank you. Good bye.

As you can see, the dialogue is acceptable for, say, novice user[1]. But an experienced one would for sure like to “speed up” the dialogue causing the system to get confused and terminate the dialogue immediately by means of some kind of escape utterance. Usually the commercial systems put the user through to a human operator after the escape utterance. Clearly, the above depicted situation is an implementative solution of a human-computer dialogue but it is only a crutch after all. A full-featured, mature human-computer interaction on the given topic should look e.g. like this (“S” is the system, “U” is the user):

S: Hello, this is ARS—Airplane-ticket Reservation System.
How can I help you?
U: *I want to book a seat on flight to Porto on April, 12^{th}.*
S: Do you mean Porto in Portugal?
U: *Well, of course, do you fly to any other Porto?*
S: There is Porto in Corsica where you may get via Calenzana.
U: *Oh, I see. But I want to Porto in Portugal.*
S: What class would you like to fly?
U: *I prefer economy class.*
S: What is your credit card number?
U: *Twelve, thirty four, fifty six.*
S: A seat in economy class is booked for you on 12^{th} April, flight to Porto, Portugal.
Your credit card number is 123456. Is it correct?
U: *Yeah.*
S: Okay. Thank you. Good bye.

Such a dialogue with computer is generally possible—we tested e.g. CSLU Toolkit package (see [1]) to design it (for English). The DIS[2] worked well but the recognition accuracy was below the acceptable level for public release, mainly because of very wide grammar covering all the possible dialogue situations. The restrictive power of the grammar was so weak that it was not able to suppress the invalid recognized utterance hypotheses. Unfortunately, Czech language has properties that make this task even more complicated as described in section 2.

Our LICS research group examined the publicly available dialogue information systems operated in Czech language with the goal to explore their dialogue strategies. The

SCITEPRESS
Science and Technology Publications

[1] Unfortunately some kind of understanding the whole matter and a good deal of obedience is necessary. And these are in practical operation quite rare.

[2] Dialogue Information System

results were discouraging: All three of Czech mobile telephone network providers have automated support lines but these have no speech recognition at all and the interaction is enabled by means of DTMF technique, i.e. pressing the buttons of a mobile phone as a reply. The “dialogue” is in all three cases extremely time-consuming, irritating, and the human operator is hidden very deep in the dialogue structure. The same was observed in information systems of four big Czech bank houses and a public transportation DIS of the city of Liberec (see [3]), but in the last named the speech recognition is present.

The previously depicted situation led the LICS team to start the development of a dialogue information system prototyping concept which will make it possible to build voice-driven applications without *that high level of restriction* in spoken interaction.

The whole framework is called LINGVO after an Esperantist word for “speech”. The recognizer part is named LASER which is short for LICS Automatic Speech Engine/Recognizer. The fundamental inspiring idea of the whole design is *to extract as much information as possible at any level, and use it back at the lowest level possible*.

2 Language Modelling Considerations

As the *phoneme recognition accuracy* can hardly exceed some 80 %[3], the relatively high utterance recognition accuracy (reported about 95-97 % in the state-of-the-art systems) grounds in powerful, restrictive language modelling which is capable of rejection of incorrect hypotheses (referred to as out-of-grammar hypotheses).

In Czech case the restrictive power of grammar (as well as statistical language models) is significantly debilitated by syntactical properties of the language. At first, Czech language has *free word order*—a question “At what time does the plane to Porto depart?” may be translated like these:

1. Kdy odlétá letadlo do Porta?
2. Kdy letadlo odlétá do Porta?
3. Letadlo do Porta odlétá kdy? *(when)*
4. Do Porta odlétá letadlo kdy? *(Porto)*
5. Kdy do Porta odlétá letadlo? *(plane)*
6. Kdy letadlo do Porta odlétá? *(departs)*

The underlined word (translated in parentheses) is emphasised by the word order. The following translations of the same question are considered strange (even if they are understood by any Czech):

1. Kdy do Porta letadlo odlétá?
2. Do Porta kdy letadlo odlétá?
3. Letadlo do Porta, kdy odlétá?

Any other possible grouping of the used words is considered out-of-grammar.

The previous example shows that any grammar constructed to accept all possible forms of grammatically correct Czech sentence can generate up to 10 % of permutation

[3] We used HTK for testing this hypothesis and reached 82.94 % as the best value (i.e. with the best setup of parameters established after a series of trials). The recognizer was trained using 30 minutes of speech uttered by 20 speakers.

SciTePress Science and Technology Publications

of the used words, which obviously cannot help the recognizer to determine validity of a hypothesis. The same situation happens while using statistical language models (N-grams). They can suppress correctly recognized out-of-grammar hypotheses (which means that the speaker uttered an out-of-grammar sentence) when N is high enough[4]. Unfortunately such an ability does not improve the recognition performance.

Another property of Czech language is a full-featured flection: Nouns, pronouns, adjectives, and numerals are declined into 7 cases for each grammatical number (resulting in 14 different forms of a word), and verbs are conjugated in a very complex way (resulting in a nightmare of 223 different forms of a verb). Both declination and conjugation is (mostly) suffix-based. Misrecognized suffix may end up in a completely different meaning of the utterance. Taking the grammatical structure of recognition hypotheses into account may result in rejection of a generally correct hypothesis due to any single misrecognized suffix. Also the model perplexity rises significantly.

We carried out a series of tests with HTK 3.2 toolkit trained with three corpora made at LICS. The table below shows the best phoneme recognition accuracies for each corpus:

Corpus	Accuracy
LICS AC 2002	71.45 %
LICS AC Chess	82.94 %
LICS AC Phonemes	72.48 %

The values of phoneme recognition accuracy are relatively high. As opposed to these, the utterance recognition accuracies reached by a voice-driven chess game recognizer proves the influence of language models:

Grammar	Accuracy
Chess Grammar 1 – Most Restrictive	96.28 %
Chess Grammar 2 – Normal	77.00 %

The “Most Restrictive” grammar forces user to announce his or her intention in a very tight manner. Such a language model is not acceptable for any Czech as it gives no freedom of word ordering which is very natural for us. On the other hand the “Normal” grammar covers nearly all possibilities of free word order sentences applicable to express a chess move. It results in a dramatic drop of performance.

3 System Design Considerations

As the grammar or statistical language model cannot play its restrictive role in Czech language DIS, we decided to *derive the restriction from dialogue course*, generally at any level of the dialogue system. To clarify the idea, let us consider the following situation: The system is asking the user “Do you have a credit card?”. There is very high probability that the user answers either “Yes” or “No”. We examined hundreds of recordings and found only few rare cases when the user faced to a pure Yes/No-question replied anything else—if he or she did so, the dialogue was not co-operative at all anyway (see [2] and [6]).

[4] Less than N = 3 has no effect in Czech language.

The points of a dialogue system design where an appropriate restrictive information can be derived from, can be e.g. the following:

1. **Acoustic Front-End (Signal Processing):**
   (a) Measuring fundamental voice frequency F_0 can tell whether the speaker is male or female. Such knowledge can be used in (i) acoustic-phonetic decoder to switch to an appropriate set of models (HMM) or neural nets (ANN) trained by men or women respectively; (ii) language modelling to conceal the grammar components for female forms (endings and other gender-specific phenomena).
   (b) Measuring prosodic parameters (e.g. overall loudness) to detect anger or stress can help to switch to a human operator (if available) in due time.
2. **Domain Analysis:** May influence the language modelling knowledge base by means of iteratively narrowing the vocabulary and grammar to the discussed domain plus some *escape utterances*.
3. **Data Analysis:** Modifies situation modelling knowledge bases to exclude dialogue sequences leading to a query about a fact which is not known to the system or which the system cannot answer for any reason.
4. **Dialogue Manager:** Being the main decisive mechanism of the dialogue system, the dialogue manager is a source of wide set of information: For example following the dialogue situation can result in considerable restriction of a language model in those branches where the user has lesser freedom of choice and thus possible interaction is predictable according to the dialogue scenario.

4 System Architecture Description

Figure 1 depicts schematically the architecture of the whole LASER/LINGVO framework. The whole prototyping concept has been designed to enable applying of modelling restrictions according to knowledge acquired all around the system.

Modules and functional units of the system design are described in detail below:

4.1 LASER—Speech Recognizer

1. **Run-time Recorder (LRec)**: Controls computer audio device(s) and records an incoming speech into a stream of digital data. Incorporates VAD and AGC[5]. Parameter setup (sample rate, quantization) is user-adjustable via configuration file and/or command line options.
2. **Acoustic Front-End (LAFE)**: Transforms the recorded digitized speech signal into a stream of *parametric vectors* used for further processing. The contents of the parametric vector is user-adjustable by a script in SACL/PDL language which defines the exact way to treat the signal. Possible processing options include preemphasis, smoothing, windowing (Hamming, etc.), power spectral density estimates (smoothed), spectral warping (Mels), MFCC, PLP, liftering, mean and deviation normalization, and many others (see [4]).

[5] VAD stands for Voice Activity Detection, and AGC means Automatic Gain Control.

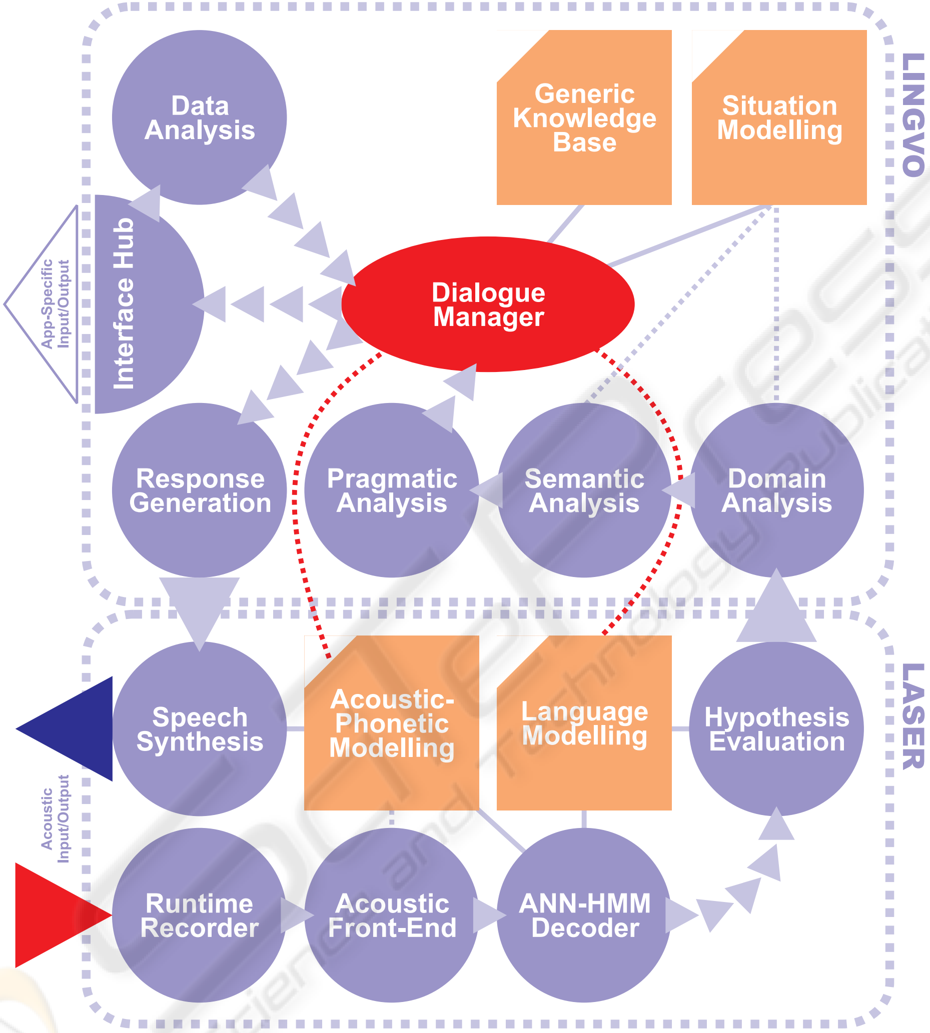

Fig. 1. Architecture of the LASER/LINGVO system: Circular elements present executory modules (routines, programs, software tools), rectangular elements stand for knowledge bases (files, databases, expert systems), triangular arrows show data flow throughout the system, solid lines connect parts that exploits one another, and dotted lines connect those that share and/or enriches knowledge bases.

3. **ANN-HMM Acoustic/Phonetic Decoder (LDec)**: Decodes the spoken (acoustic) utterance represented by parametric vectors into a phonetic information (series of phonemes represented by transcription alphabet symbols) by means of proposing recognition hypotheses based on acoustic and language modelling. Artificial neural network (namely MLP) estimates posterior probabilities of phonetic class assignment for each parametric vector. These values are used as output probabilities b_j in states of HMMs of phoneme-like units (see [7]).
4. **Hypothesis Evaluation**: Searches the proposed recognition hypotheses in the shape of word lattice and, according to the language modelling knowledge base (and thus information provided by upper level of the design, e.g. dialog manager), accepts (N) the most probable way(s) through the lattice, i.e. the valid hypothesis.
5. **Speech Synthesis (LSyn)**: Shares the acoustic-phonetic knowledge base to produce audible speech output.
6. **Acoustic-Phonetic Modelling Knowledge Base**: Contains models of acoustic phenomena and their phonetic class assignment in form of numeral parameter sets for HMM (matrices of a_{ij} and b_j, transition and emission probabilities). This knowledge base can be *enriched by external knowledge* according to the proposed design concept of gathering and spreading knowledge: (a) The acoustic front-end (LAFE) module can determine whether the prospective speaker is male or female and cause switching to the appropriate set of acoustic-phonetic models instead of using both two—resulting performance improvement is estimated about 10 %; (b) the same piece of information can come from dialogue manager (as Czech strongly differentiates grammatical gender).
7. **Language Modelling Knowledge Base**: Contains language models, i.e. grammar in e.g. extended Backus-Naur form (EBNF), numeral parameter sets of N-gram statistical models, etc. This base can be also strongly influenced by spreading knowledge from upper parts of the design: The information from situation modelling base (via dialogue manager) can suppress grammar branches that will not be used for sure in the next user's reply. Our contemporary technical solution of this task is a re-generation of the used grammar before each utterance analysis.

4.2 LINGVO—Dialogue System

1. **Domain Analysis**: At this point, a decision about what domain does the utterance belong to is taken. According to such a knowledge, an appropriate situation models are passed to the dialogue manager. Also an off-topic sentence can be identified here and the dialogue manager is consequently alerted to switch to an "escape" scenario. The module is based mainly on the vocabulary and syntax analysis (see [5] and [8]).
2. **Semantic Analysis**: Analyses the utterance with the goal to find the meaning of it, i.e. expressed intention of the speaker in communication towards the system. Semantic analysis is grounded on microsituation theory and several other semantic formalisms (see [5]): The method tries to fill in predefined semantic frames (data structures) using the information contained in the sentence—those frames that are filled more than certain given level are declared valid semantic hypotheses and passed to the next module.

3. **Pragmatic Analysis**: Verifies whether the semantic hypothesis is accomplishable given the contents of domain-specific databases. Pragmatic analysis also contributes to quantitative formulation of the cooperativeness level between the user and the system. Such an information helps to select suitable dialogue strategies within the situation modelling base (via dialogue manager).
4. **Data Analysis**: Scans the data produced by controlled (subordinated) applications and returned to the dialogue system through interface hub. The module is responsible for filtering singularities from the data and translating the data into semantic frames so that dialogue manager can operate on them.
5. **Interface Hub**: Ensures communication with controlled (subordinated) applications such as relational databases, system terminals, game engines, etc.
6. **Response Generation**: Translates filled-in data frames back to human speech in the form of a sequence of phonetic symbols, which is further passed to the speech synthesizer.
7. **Generic Knowledge Base**: Contains common facts needed to decode incomplete semantic frames or those carrying implicit entries like e.g. local date and time, position of the running system, etc. In the other words it holds a system-specific description of the world.
8. **Situation Modelling Knowledge Base**: Contains dialogue and subdialogue scenarios derived out of long-lasting research of real human-human dialogues, dialogue templates, and behavioural patterns (see [2]). This module is a prominent source of knowledge used to restrict recognizer grammar.

5 Current State of Implementation

The following units and modules are fully functional:

1. **LASER Recognizer Unit**—Provides the system with either the best recognized sentence hypothesis or N-best hypotheses. The experimental hybrid ANN/HMM decoder (see [4]) may be optionally replaced by HTK/ATK-based decoder. Implemented also as DLL[6], the recognizer may be utilised by various simple speech-enabled applications too.
2. **Domain Analysis**
3. **Interface Hub**
4. **Response Generation**

Interface routines (written in Perl) enable to incorporate executive modules or data from other systems, e.g. HTK, CSLU Toolkit or SPEX KIT. The **Semantic** and **Pragmatic Analysis** modules, and the **Dialogue Manager** are partially implemented, i.e. they are available in a simple form for testing and display purposes. Still they are not ready as generic full-featured data-driven modules. Currently a co-operative effort is exerted to bind LASER/LINGVO system to SPEX KIT dialogue platform (see [9]).

A complete methodology is prepared for the dialogue modelling: microsituations, dialogue flow, escape strategies, etc. Also several real recorded dialogues were modelled using the methodology to verify its efficiency (see [6]).

[6] Windows Dynamically Linked Library which can be loaded by an application at run time.

Several simple dialogue systems have been developed using the LASER/LINGVO framework: (i) LChess—a chess game controlled by voice interaction; (ii) DOD@live—a DIS for “Day of Open Doors” at DCSE answering questions of our prospective students about the studies at our department; (iii) CIC (or City Information Centre)—a municipality DIS providing information about city transportation, opening hours, etc. (under development).

6 Results and Future Work

The way how the prototypes (see section 5) nowadays function (on an isolated Windows-based workstation with a headset) does not allow an extensive testing under real operating conditions. We performed a simple test during the above mentioned “Day of Open Doors” when 113 uninitiated[7] students talked to the DOD@live dialogue system prototype. The results were as follows:

Wrong system response	**6.81 %**
Correct system response	**93.19 %**
| Correct hypothesis (A)	59.09 %
| Wrong hypothesis (B)	34.09 %

State (A) means that the recognizer provided the system with correct hypothesis and the system subsequently took an appropriate action (response) so that the user was satisfied. State (B) is a situation when the recognizer provided the system with (partially) incorrect hypothesis but the system was still able to derive the meaning of the utterance and take an appropriate action (response) to satisfy the user.

The weakest point of current LASER/LINGVO implementation state is definitely the semantic and pragmatic analysis as these modules can act as efficient restriction of recognition hypotheses. Also a rejection mechanism for totally out-of-dialogue hypotheses with high recognition score (Hypothesis Evaluation module) works at disputatious level of accuracy leading the system to dead ends.

Our future work will be focused towards implementing data-driven algorithms for semantic and pragmatic analysis. Another important branch is to improve the dialogue manager core to (i) handle exceptional dialogue states, (ii) support escape strategies, and (iii) cover wider field of dialogue situations (i.e. make the frame processing more generic). Moreover we’d like to incorporate some recently presented NLP techniques suitable for Czech language but unfortunately these are usually too theoretic and too demanding to be implemented in a real-time response system.

7 Conclusion

The dialogue information system paradigm described in the previous paragraphs has been proposed and built mainly because of the need of a design concept enabling to increase the cooperative performance between a human and a machine. Such result is strongly dependent on the speech recognition and semantic analysis accuracy as these

[7] They were not previously instructed how to speak to the system and what to say.

are key components in the process of artificial understanding of speech. The design was penetrated with a fundamental idea of highest possible modelling restriction so that the decoding algorithms have lesser freedom and thus gaining better results. The need for such a scheme came out of syntactical properties of Czech language for which both grammar and statistical language models allow too many possibilities and thus it can hardly help to reject invalid recognition hypotheses. The original idea of restricting the recognition grammar according to the position dialogue scenario was extended to the other parts of the framework and resulted in a general scheme of dialogue information system with spreading of extracted knowledge.

Acknowledgement

This research has been supported by Research Grant MSM 235200005.

References

1. Johan Schalkwyk, Paul Hosom, Ed Kaiser, Khaldoun Shobaki: CSLU-HMM: The CSLU Hidden Markov Modeling Environment. Center for Spoken Language Understanding, Oregon Graduate Institute of Science & Technology, USA (2000)
2. Schwarz J., Matoušek V.: Automatic Analysis of Real Dialogues and Generation of Training Corpora. In: Proceedings of the Int. Conference EUROSPEECH 2001, Volume 4. Aalborg, Denmark (2001) 2201–2204
3. Nouza J.: Speech Processing Technology Applied in Public Telephone Information Services. In: Proc. of 4th World Conference on Systemics, Cybernetics and Informatics (SCI 2000), vol. IV. Orlando (2000), USA 308–313
4. Ekštein K., Mouček R.: Detection of Relevant Speech Features Using Driven Spectral Analysis (The LASER Case). Proceedings (CD) of 4^{th} International PhD Workshop Information Technologies and Control. Institute of Information Theory and Automation, Prague, Czech Republic (2003)
5. Moucek R., Ekštein K.: Municipal Information System. Proceedings (CD) of 4th International PhD Workshop Information Technologies and Control. Institute of Information Theory and Automation, Prague, Czech Republic (2003)
6. Lorenzová, E.: Psycholingvistická analýza komunikace člověka s dialogovým informačním systémem (Psycholinguistic Analysis of Communication Between Human and Dialogue Information System). M.A. Thesis (in Czech only). University of West Bohemia, Plzeň, Czech Republic (2003)
7. Pavelka, T.: Hybrid Speech Recognizer Implementation. M.Sc. Thesis. University of West Bohemia, Plzeň, Czech Republic (2003)
8. Beneš, V.: Sémantická analýza doménově roztříděných dialogů (Semantic analysis of domain dependent dialogues). M.Sc. Thesis (in Czech only). University of West Bohemia, Plzeň, Czech Republic (2003)
9. SPEECH EXPERTS GmbH: Whitepaper, `www.speech-experts.com`, Regensburg, Germany (2003)