SQEL: A Multilingual and Multifunctional Dialogue System

Maria Aretoulaki
Bavarian Research Centre for Knowledge-Based Systems (FORWISS)
Am Weichselgarten 7, D-91058 Erlangen, Germany

and S. Harbeck and F. Gallwitz and E. Noeth and H. Niemann
University of Erlangen, IMMD5, Erlangen, Germany

and J. Ivanecky and I. Ipsic+ and N. Pavesic+ and V. Matousek@*
TU Kosice, University of Ljubljana+, University of Pilsen@*

ABSTRACT

Within the EC-funded project SQEL, the German EVAR spoken dialogue system has been extended with respect to multilinguality and multifunctionality. The current demonstrator can handle four different languages and domains: German, Slovak, and Czech (and their national train connections), and Slovenian (European flights). The SQEL demonstrator can also access databases on the WWW, which enables users without an internet connection to meet their information needs by just using the phone. The system starts up with a German opening phrase and the user is free to use any of the implemented languages. A multilingual word recognizer implicitly identifies the language, which is then associated with the appropriate domain and database. For the remainder of the dialogue, the corresponding monolingual recognizer is used instead. Experiments to date have shown that the multilingual and the (respective) monolingual recognizers attain comparable word accuracy rates, although the former is less efficient. The existence of language-independent task parameters, such as goal and source location, has meant that porting the system to a new language involves mainly the development of lexica and grammars (apart from the word recognizers) and not an extensive restructuring of the interpretation process within the Dialogue Manager. The latter is sufficiently flexible to switch between the different domains and languages.

1. The EVAR Dialogue System

The spoken dialogue system *EVAR*[1] has been connected to the German public telephone network since 1994 to answer enquiries on German InterCity train connections [3, 2]. One of the ambitions regarding EVAR has been to render it *multifunctional*. The application should be generalised to cover not just train connections, but also other means of transport, as well as hotel and holiday reservations. The first step towards this direction has been the development of the SQEL demonstrator, which covers multiple languages and domains. In Section 2, the multilingual recognizer and the multifunctional dialogue manager are described, including preliminary results with the former. Then in Section 3, the connection of the system to the World-Wide-Web is explained.

[1] Erkennen - Verstehen - Antworten - Rückfragen (Recognize, Understand, Reply, Ask back)

2. Multilinguality and Multifunctionality

The multifunctionality of the EVAR system was tested in the framework of the EC-funded Copernicus project COP-1634 *SQEL (Spoken Queries in European Languages)* [6]. The goal was partly to enhance the functionality of the system with regards to a number of domains, namely flight and train information. The main aim, however, was to achieve multilinguality for EVAR, that is the system should be capable of operating across the German, Slovak, Slovenian, and Czech languages. The core of this research has been the development of a multilingual word recognizer (Section 2.1) and the extension of the already flexible Dialogue Manager of EVAR (Section 2.2), giving rise to the SQEL demonstrator.

2.1. Speech Recognition

One of the major tasks of a multilingual dialogue system is the recognition of the user utterances. Inside the SQEL system, this is done by a multilingual Speech Recognizer (SR). One method to perform multilingual speech recognition is to run all existing recognizers in parallel and choose the most probable word chain. To reduce the computational load, a single recognizer was built instead that contains the words from all languages in its dictionary.

The basis for our multilingual SR is a series of monolingual SRs. Semi-continuous HMMs are used for acoustic modelling and bigrams for linguistic modelling. The monolingual recognizers are trained in the ISADORA environment which uses polyphones with maximum context as subword units [5]. The development of the multilingual SR involved the following steps:

1. The number of codebook density functions was increased to reflect the language-dependent codebooks. In the case of two languages, for example, with a codebook of 256 density functions for each, the multilingual recognizer would have 512 density functions.

2. Special weight coefficients were added to the HMM output density functions to reflect the increased number of available density functions. The new weight coefficients were set to zero, so that every density function belonging to different languages bears no effect on the output probability of the HMM.

3. A special bigram model was constructed which consists of the monolingual bigrams and does not allow any transitions between languages, as shown in Equation 1.

$$P(\text{word}_{language_i}|\text{word}_{language_j}) = 0 \quad \text{for } i \neq j \quad (1)$$

4. A special silence category was established for language-specific silence models, which allows transition to and from every language, so that the language can be switched by means of inserting pauses.

In order to reflect the quality of the acoustic models for the different languages, an additional a priori value was introduced for each language. In theory, there will only be word chains in the spoken language after a few seconds, using the standard beam search in forward decoding. The effect is that the number of words inside the active vocabulary will be the same as when using the respective monolingual recognizers.

Experimental Results Our approach to multilingual speech recognition has been evaluated with the four languages of the SQEL project; German, Slovenian, Slovak and Czech. Because of the special silence category used, the recognized word chain can contain words from different languages. In order to assess the accuracy, the language of the word chain is determined on the basis of the number of words in each language, selecting the one with the most words. All words found in other languages are deleted from the recognized word chain, each one counting as a deletion error. In the context of a dialogue system, only the first user utterance will be processed by the multilingual SR. The language identified at that point will be adopted for the whole of the remaining dialogue, which involves the use of a monolingual SR.

As shown in Table 1, the monolingual SRs are still superior to the multilingual SR, because of the instances of language identification failure salient in the latter. These failure instances occur especially within short sentences, as the time available for a robust discrimination between languages is insufficient in these cases (Table 2). In evaluating the mono- and the multilingual SRs on utterances with more than 5 words, there are only slight differences in the corresponding word accuracy rates, but the language identification rates are significantly higher. The Real Time Factor (RTF) for the multilingual system is more than two times higher than for monolingual recognizers with the language already established. However, the multilingual system is nearly twice as fast as using 4 monolingual recognizers in parallel. The reason for this is that, at the beginning, all possible languages are inside the beam and for every language all codebook densities have to be calculated, regardless of whether they are inside the active vocabulary or not.

Recognizer	**Recognition Rates (Word Accuracy)**				**RTF**
	Slovenian	Slovak	Czech	German	
Mono Slovenian	88% (90%)				1
Mono Slovak		88% (88%)			1
Mono Czech			84% (83%)		1.3
Mono German				90% (91%)	1.2
Multi	83% (87%)	86% (85%)	84% (83%)	84% (86%)	2.5

Table 1: Recognition rates and Real Time Factor (RTF) using monolingual and multilingual speech recognizers on all sentences of the SQEL test corpus; the recognition rates for sentences longer than 5 words are shown in brackets.

Test Set	**Slovenian**	**Slovak**	**Czech**	**German**
All sentences	97%	90%	92%	90%
Sentences longer than 5 words	97%	96%	97%	96%

Table 2: Language identification rate using the multilingual recognizer on all sentences and on sentences longer than 5 words

2.2. Dialogue Management

Apart from the language identification issues (Section 2.1), the Dialogue Manager (DMan) of the SQEL system has to be sufficiently flexible to switch between domains, as well as between parsers, generators, and databases, depending on the language. This has already been achieved: There is currently a multilingual and multifunctional version of the initial EVAR system (Section 1) which works for German, Slovak, and Czech (train connections), and Slovenian (flight connections). As the system starts up, the user is free to use whichever language they want. Once the language has been identified by the word recognizer, it is associated with the corresponding domain.

The switch to a new language / domain is triggered, when the *multilingual word recognizer* forwards –apart from the user's first utterance– a two-letter marker to the DMan, which identifies the language spoken; `ge` for German, `sk` for Slovak, `sl` for Slovenian, and `cz` for Czech. This language I.D. is received by the Linguistic Module (LI) of DMan, which constitutes its interface to the parser (Fig. 1). When LI reads a language marker before the recognized word string, it automatically switches the language flag for the whole system to the appropriate language and initialises the new parser. The application flag is also changed, if

necessary, from train to flight enquiries and vice versa, and the database is redefined.

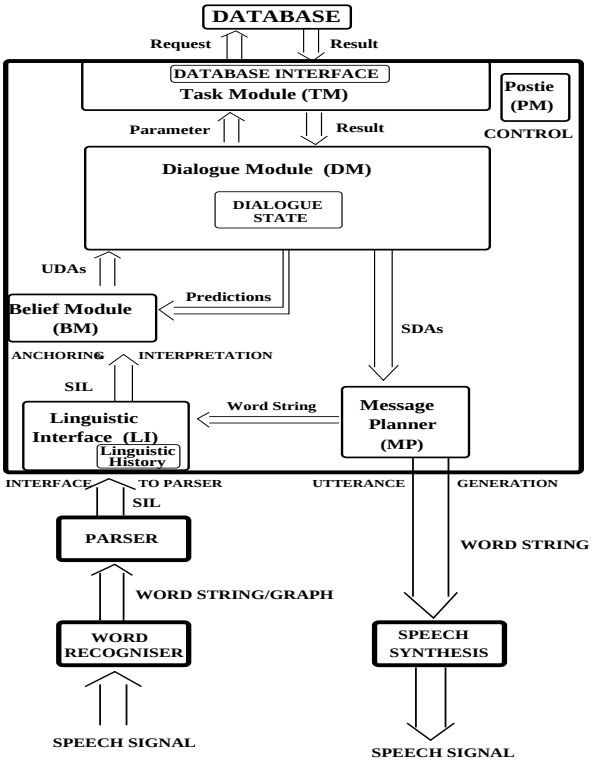

Figure 1: The Dialogue Manager of EVAR / SQEL

Once the new *parser* has been loaded, the utterance is parsed using the appropriate rules for the syntax of the language, as well as for the correlation of syntactic and lexical patterns to the semantics of the application and of the dialogue. These two types of semantic objects are defined in the SIL representation language. The EVAR system used a customised version of the generic SIL definitions that also covers domain-specific concepts. All the semantic objects relevant to the discourse interpretation of user utterances (e.g. `want`) are contained therein, as are all and only those objects relevant to the application (e.g. `sourcecity`, `goalcity`, `sourcetime`). Customisation to the flight domain involved a specification, for example, of all the cities covered by the airline and database query parameters, such as `source` and `goalairport` or `date`. A simultaneous call to both the train and the flight domains was achieved for the SQEL demonstrator by modifying the customisation procedure to produce a version of the generic SIL ontology which includes both sets of definitions. Thus, whether the system is in a mono- or a multilingual mode, the same ontology is used, ready to be applied to either domain. The language-specific parser carries out a translation of the user utterance into a number of SIL concepts, so that the utterance can be interpreted in the context of the domain and the dialogue history. This contextual interpretation is done mainly in the *Belief Module* (BM) (Fig. 1). It is here that underspecified representations are 'anchored' to already known objects (anaphora resolution) on the basis of predictions about the progression of the dialogue.

At this stage, the result of the processing of the user utterance is a disambiguated semantic representation of their intentions (UDAs in Fig. 1). UDAs are used by the *Dialogue Module* (DM), which keeps track of the progression of the dialogue in terms of system goals. The system's current goal is contrasted to that of the user and modified accordingly, as the default dialogue strategies are mixed-initiative and the user has the freedom to change the focus of the interaction. Apart from the introduction of a rule for switching between domains, the function of DM has remained identical; it extracts from the UDAs that were forwarded by BM the value(s) for various task parameters and passes them on to the *Task Module* (TM). TM constitutes the interface of DMan and the whole system to the application database (Fig. 1). In this module, the parameters that should be specified by the user before the database can be accessed are identified. TM records this instantiation process and either accesses the database or sends a request for a new system goal to acquire the missing values. For the development of the SQEL system, special preferences were added for database look-up in the cases of Czech and Slovenian, regarding the indispensability of each parameter. For each of the new languages, databases were set-up and interfaces between them and TM. The function of the latter is the collection and postprocessing of the database results before forwarding them to DM. Module reloading is accommodated for all languages, in accordance with the domain switch. Whether or not all necessary parameters for database look-up have been specified by the user, the next step is the planning of the next system utterance. This takes place in the *Message Planner* (MP) of DMan (Fig. 1). MP generates a request for information or clarification / confirmation (when not all parameters have been instantiated), or it supplies the database entries retrieved. This module was only modified to include a specification of the generator for each additional language. MP interfaces with the appropriate one in order for the next system goal to be realised. The generator itself is where the system phrasing is formulated, depending on the dialogue act to be communicated to the user.

3. Towards Multimodality

One of the ambitions regarding the original EVAR system has been its evolution into a *multimodal environment*, where speech, text, and even image processing are integrated over the phone and the internet (cf. [4]). To this end, a search engine was developed that poses the user's query over the phone to multiple travel information databases on the World-Wide-Web (WWW) and returns the appropriate entries [1]. The search engine constitutes the interface between the system and the WWW databases. Thus, it receives a semantic representation of the query and transforms it into the appropriate HTML forms, which in turn are forwarded to multiple databases: German Railways, Lufthansa, and Swiss Railways. During the search, HTML documents are dynamically created and accessed holding the intermediate results collected. These

are temporarily saved in a local cache which is continually updated until the search is ended. When the initial query does not match any of the stored data, constraints can be relaxed, so that a solution becomes available.

More specifically, the user query is passed on to the databases as a list of parameter-value pairs, which together constitute the user's requirements; e.g.

```
[date:[011298,011298],goalcity:berlin,
sourcecity:hamburg,goaltime:[1300,2100]]
```

The user query has already been translated into the engine's query format, something that involves mainly the addition of a **vehicle** parameter; e.g.

```
[date:[011298,011298],vehicle:air,goalcity:berlin,
sourcecity:hamburg,goaltime:[1300,2100]]
```

The various databases are accessed and a maximum of two results are returned to the Task Module of DMan and passed on to the Dialogue Module for the formulation of the next system goal/utterance (Section 2.2). This is planned in the Message Planner and generated by the Generator of the system.

Only a few changes were effected on the standard EVAR system: In TM, the interface between DMan and the WWW-search program was defined, by setting up query and result temporary files and deleting the socket connection rules used previously for accessing a local database. In addition, the data exchange between this and DM was established for the formulation of appropriate system goals / dialogue acts. Lastly, the Generator was extended to include the realisation of the **vehicle** parameter for the translation of the user query to the WWW format. The fully-integrated SQEL demonstrator constitutes a **multilingual** (German, Slovak, Slovenian, Czech), **multifunctional** (train and flight domains, Out-Of-Vocabulary Word facility [2]), and **multimodal** (WWW access for German queries only) environment. 175 multilingual dialogues have been collected to date with a system version which operates over the phone for German and Slovak (train domain), and for Slovenian (flight domain). The Czech modules have, since then, been integrated in the general architecture.

4. Conclusion

In this paper, the various extensions effected on the German train information system EVAR were reported which have given rise to the multilingual, multifunctional, and even multimodal SQEL demonstrator. The efficient and straightforward development of the integrated SQEL prototype have provided strong evidence that the system architecture is sufficiently *flexible* to be extended with additional modules for different languages, and the data circulating within this architecture is sufficiently abstract and *comprehensive* to cover different application domains. Although the domains are not completely unrelated to one another, their parallel implementation has not been a trivial task, especially given that spontaneous speech and even foreign languages are concerned. The multilingual recognizer employed is nearly twice as fast as using 4 monolingual recognizers in parallel, and the word accuracy rates attained are comparable to those with the monolingual recognizers, especially for utterances longer than 5 words. Currently, an analysis of the 175 dialogues collected with this *'multidimensional'* system is being carried out for the evaluation of its completion and transaction success rates, and an enlarged corpus is being compiled after the inclusion of Czech. The further adoption of the present Dialogue Manager for other domains and applications is also being envisaged, e.g. a travel information and booking system, a company database query system, or an automated Call Centre.

5. REFERENCES

1. J. Adelhardt. Sprachgesteuerte Parallele Abfrage von WWW-Datenbanken. Diplomarbeit, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen-Nürnberg, Erlangen, Germany, Aug 1997.
2. M. Boros, M. Aretoulaki, F. Gallwitz, E. Nöth, and H. Niemann. Semantic Processing of Out-Of-Vocabulary Words in a Spoken Dialogue System. In *Proceedings of the 5th European Conference on Speech Communication & Technology (EUROSPEECH '97)*, volume 4, pages 1887–1890, Rhodes, Greece, 1997.
3. W. Eckert, T. Kuhn, H. Niemann, S. Rieck, A. Scheuer, and E. G. Schukat-Talamazzini. A Spoken Dialogue System for German Intercity Train Timetable Inquiries. In *Proceedings of the 3rd European Conference on Speech Communication and Technology (EUROSPEECH'93)*, pages 1871–1874, Berlin, 1993.
4. D. G. Novick and D. House. Spoken-Language Access to Multimedia (SLAM): A Multimodal Interface to the World-Wide Web. Technical report, Center for Spoken Language Understanding, Oregon Graduate Institute, 1995. Submitted to SIGIR '95, Seattle, WA.
5. E. G. Schukat-Talamazzini, T. Kuhn, and H. Niemann. Speech Recognition for Spoken Dialog Systems. In H. Niemann, R. D. Mori, and G. Hahnrieder, editors, *Progress and Prospects of Speech Research and Technology*, number 1 in Proceedings in Artificial Intelligence, pages 110–120. Infix, 1994.
6. University of West Bohemia. *Proceedings of the 2nd SQEL Workshop on Multi-Lingual Information Retrieval Dialogs, April 27 – 29*, Pilsen, Czech Republic, 1997.