The LIMSI RailTel System: Field trial of a Telephone Service for Rail Travel Information[*]

L.F. Lamel, S.K. Bennacef, S. Rosset, L. Devillers, S. Foukia, J.J. Gangolf, J.L. Gauvain
Spoken Language Processing Group
LIMSI-CNRS
91403 Orsay, FRANCE
lamel@limsi.fr

July 1, 1997

Revised paper for Speech Communication (IVTTA'96)

Keywords: spoken language systems, speech recognition, speech understanding, natural language understanding, information retrieval dialog

Number of pages: 20
5 tables
10 figures

Abstract

This paper describes the RailTel system developed at LIMSI to provide vocal access to static train timetable information in French, and a field trial carried out to assess the technical adequacy of available speech technology for interactive services. The data collection system used to carry out the field trials is based on the LIMSI Mask spoken language system and runs on a Unix workstation with a high quality telephone interface. The spoken language system allows a mixed-initiative dialog where the user can provide any information at any point in time. Experienced users are thus able to provide all the information needed for database access in a single sentence, whereas less experienced users tend to provide shorter responses, allowing the system to guide them. The RailTel field trial was carried out using a common methodology defined by the consortium. 100 naive subjects participated in the field trials, each calling the system one time and completing a user questionnaire. 72% of the callers successfully completed their scenario. The subjective assessment of the prototype was for the most part favorable, with subjects expressing an interest in using such a service.

Cet article décrit le système RailTel développé au LIMSI, destiné à l'accès vocal en Français aux horaires des trains de la SNCF et permettant d'évaluer l'adéquation des techniques vocales pour les services interactifs. Le système utilisé pour le recueil de

[*]This work was partially financed by the LE MLAP project 63-022 RailTel.

corpus des tests a été développé à partir du système Mask du LIMSI. Il fonctionne sur une station Unix avec une interface téléphonique de haute qualité. Le système offre un dialogue à initiative partagée où l'utilisateur peut fournir les informations à tout instant. Les utilisateurs expérimentés peuvent fournir toutes les informations en une seule phrase, tandis que les utilisateurs moins expérimentés ont tendance à donner de courtes réponses laisant le système les guider. Les tests de RailTel ont été effectués selon une méthodologie commune définie par le consortium. 100 sujets naïfs ont participé aux tests, chacun a appelé le système une seule fois et rempli un questionnaire. 72% des sujets ont achevé leurs scénarios avec succès. L'évaluation subjective du prototype était en majorité favorable, avec un intérêt d'utiliser un tel service.

In dem vorliegenden Beitrag beschreiben wir das am LIMSI entwickelte RailTel System. Es ermöglicht, telefonisch Zugfahrplanauskünfte in Französisch zu erfragen. Im Rahmen der Systementwicklung wurden Experimente durchgeführt, um den technischen Stand heutiger Sprachtechnologien für interaktive Dienste zu bewerten. Zum Zweck der Datensammlung wurde das am LIMSI entwickelte Sprachverarbeitungssystem Mask verwendet. Es arbeitet auf einer UNIX Station unter Benutzung einer Telefonschnittstelle von gehobener Qualität. Die Dialogstrategie ist gemischt und erlaubt dem Benutzer, jederzeit zusätzliche Informationen zu liefern. Erfahrene Benutzer sind daher in der Lage, sämtliche Auskünfte, die für den Datenbankzugriff notwendig sind, in einem einzelnen Satz zusammenzufassen. Unerfahrene Anwender neigen hingegen eher dazu, kurz zu antworten und können somit vom System angeleitet werden. Eine gemeinsame Vorgehensweise bei den RailTel Experimenten wurde im Vorfeld vom Konsortium definiert. 100 unerfahrene Benutzer nahmen an den Experimenten teil, bei denen jeder von ihnen das System einmal anwählte und im Anschluß einen Fragebogen ausfüllte. 72 % der Teilnehmer schlossen ihr Szenario erfolgreich ab. Die subjektive Bewertung des Prototyps fiel in den meisten Fällen positiv aus, die Benutzer zeigten mit anderen Worten Interesse, einen derartigen Service auch in Anspruch zu nehmen.

1 Introduction

In this paper we describe the spoken language system developed at LIMSI as part of the LE-MLAP project "Railway Telephone Information Service" (RailTel), and a field trial carried out with the system. The goal of the RailTel) project was to assess the technical adequacy of available speech technology for interactive telephone services, in particular, for vocal access to rail travel information. Telephone information services require that all interaction with the user is vocal, making oral dialog management and response generation very important aspects of the system design and usability.

The RailTel system[1] is largely based on the spoken language system developed for the Esprit Mask project (Gauvain et al., 1995a,b), and allows users to obtain information taken from the French Railways (SNCF) static timetables and limited additional information about services offered on the trains, fares and fare-related restrictions and reductions. The system is composed of a speech recognizer, and natural language understanding, dialog management and response generation components. The speech recognizer transforms the input signal into the most probable word sequence and forwards it to the natural language understanding component, which carries out a caseframe analysis and generates a semantic frame representation. The dialog manager prompts the user to supply any missing information needed for database access and then generates a database query. The retrieved information is transformed into a

[1] The continuation of this work is being partially financed by the LE-3 project 4229 Arise.

natural language response by the response generator (taking into account the dialog context) and vocal feedback is provided to the user. To ensure high quality speech output, synthesis by waveform concatenation is used, where dictionary units are put together according to the generated text string.

The system runs on a Unix workstation with a telephone interface which can handle up to 4 telephone lines. The LIMSI prototype service for the French language was developed over the summer of 1995, and demonstrated at the *Eurospeech'95* conference. This system was used to collect telephone data (about 4000 queries) which were used to construct new acoustic and language models for the speech recognizer. This prototype service was used to carry out a field trial with 100 naive users during the fall of 1995, according to the common protocol designed for the project. Similar field trials were carried out by our Italian partners (Billi et al., 1996) and British partners with their prototype systems.

In this paper we provide an overview of the RailTel spoken language system, describing the technology used in each system component. Being of particular importance in telephone-based services, we devote sections to dialog management and natural language response generation. Sections 7 and 8 describe the field trial and provides results using objective and subjective evaluation measures. The 100 dialogs were analysed to determine the sources of errors (speech recognition, understanding, information retreival, or dialog management). Finally we conclude with some comments on the field trials and the continuation of this research in the context of the Arise project.

2 The RailTel Prototype System

An overview of the spoken language system for information retrieval (Lamel et al., 1995b) is shown in Figure 1. The main components are the speech recognizer, the natural language component which includes the semantic analyzer and the dialog manager, and the components for information retrieval and response generation. While our aim is to develop underlying technology that is speaker, task and language-independent, any spoken language system will necessarily have some dependence of the chosen task and on the languages known to the system (Lamel et al., 1995a). The spoken query is decoded by a speaker independent, continuous speech recognizer (Gauvain et al., 1994a), whose output is then passed to the natural language component. In our current implementation the output of the speech recognizer is the best word sequence, however, the recognizer is also able to provide a word lattice. The semantic analyzer carries out a caseframe analysis to determine the meaning of the query, and builds an appropriate semantic frame representation (Bennacef et al., 1994). The dialog history and default values derived from the task knowledge are used to complete the semantic frame. If the semantic frame is incomplete with respect to information required for database access, the dialog manager prompts the user to supply missing information. When all the required information is available, a database query is generated, accessing a static version of the train information database (Riho). The returned information is converted into a natural language response by the response generator and played to the user.

3 Speech Understanding

There are three stages in the speech understanding component which transforms the acoustic signal into a semantic-pragmatic representation: speech recognition, literal understanding

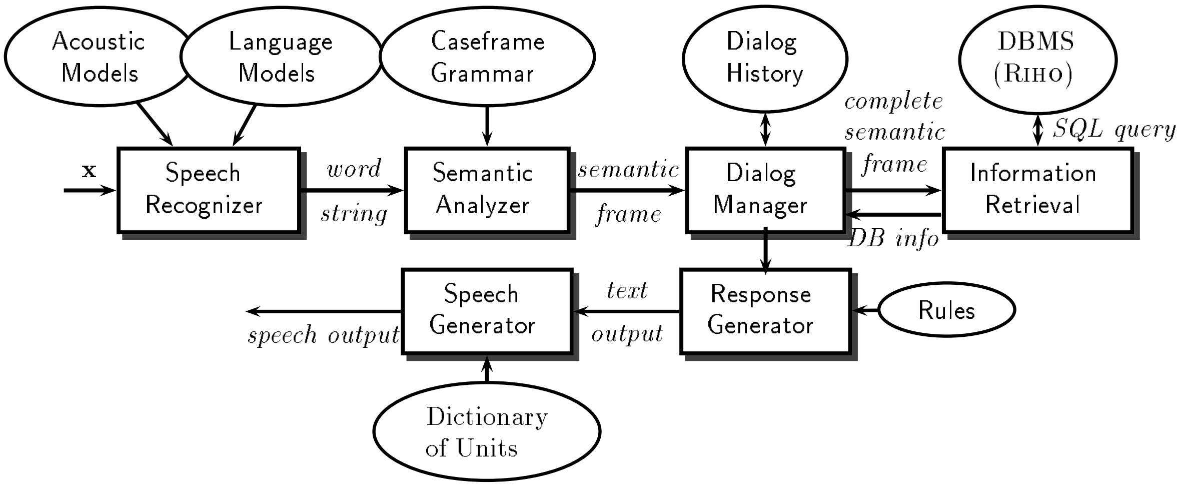

Figure 1: Overview of the RailTel data collection system for spoken language information retrieval. (**x** is the input speech signal.)

and contextual understanding.

3.1 Speech Recognition

The speech recognizer is a medium vocabulary, speaker-independent, continuous speech recognizer (Gauvain et al., 1996). It is a software-only system (written in ANSI C) that runs in real-time on a standard Risc processor. The recognizer uses continuous density HMM with Gaussian mixture for acoustic modeling (Gauvain et al., 1994b; Adda et al., 1997) and *n-gram* backoff language models (Katz, 1987). The feature vector contains 12 MFCC cepstral coefficients computed on the 0.3-4kHz telephone band and their first and second order derivatives (Gauvain et al., 1995). The *n*-gram statistics are estimated on the transcriptions of queries in the training data and from data recorded with the Mask system. Since the amount of language model training data is small, some grammatical classes (such as cities, days, months, etc) are used to provide more robust estimates of the *n*-gram probabilities. The current RailTel recognition vocabulary contains about 1500 words, including the 600 station/city names specified by the SNCF. The recognition vocabulary used in the field trial contained 800 words and included 58 station names.

3.2 Literal Understanding

After recognition, each utterance is analysed using a caseframe grammar (Fillmore, 1968; Bruce, 1975; Bennacef et al., 1994), in order to build one or several semantic frames which are saved in a graph. In the caseframe analysis, keywords are used to select an appropriate case structure for the query without attempting to carry out a complete syntactic analysis. A restricted local syntax is used to provide additional constraints in interpreting numbers which are used frequently in this task occurring in dates, times, and train numbers. The caseframe parser has been implemented in C++. The caseframe grammar is described in a declarative file so as to be able to easily modify the cases. The concepts for the RailTel task, shown in Figure 2, are **train-time**, **fare**, **change**, **type**, **reserve**, **service** and **reduction**. These

Semantic category	*Example*
train-time	*Quels sont les* **horaires** *des trains allant de Paris à Lyon ?* What are the **times** of trains from Paris to Lyon ?
fare	*Quel est le* **prix** *du billet ?* How **much** is the ticket ?
change	*Quels sont les* **changements** *?* What are the **connections** ?
type	*Quel est le* **type** *du train qui arrive à 20 heures 5 ?* What **type** of train is the one arriving at 20:05 ?
reserve	*Je veux* **réserver** *une place dans le train de 8 heures 10.* I want to **reserve** a seat on the 8:10 train.
service	*Quelles sont les* **prestations** *offertes dans ces trains ?* What **services** are available on these trains ?
reduction	*Qu'est-ce qu'un billet* **Jocker** *?* What is a reduction **Jocker** ?

Figure 2: RailTel concepts.

Je veux aller de Paris à Marseille demain matin vers 10h en passant par Lyon.
(I would like to go from Paris to Marseille via Lyon around 10 tomorrow morning.)

```
<train-time>
from:  paris
to:  marseille
stop:  lyon
relative-day:  demain (tomorrow)
morning-afternoon:  matin (morning)
relative-departure-time:  vers (around)
departure-hour:  10
```

Figure 3: Example query and semantic frame after literal understanding.

concepts were determined by analyzing the queries in the training corpus to augment the *a priori* task knowledge. While in the RailTel field trials only a subset of these concepts were directly used (**train-time** and **change**), subjects solved additional scenarios to test the other system capabilities. After literal understanding the resulting semantic frame contains a set of slots instantiated by the meaningful words of the utterance (Bennacef et al., 1996). An example query and semantic frame are shown in Figure 3.

3.3 Contextual Understanding

Contextual understanding consists of interpretating the utterance in the context of the ongoing dialog, taking into account common sense and task domain knowledge. The semantic frames resulting from literal understanding are reinterpreted using default value rules and qualitative values are transformed into quantitative ones.

Default value rules supply default values not specified by the user. For example, if the departure month was not specified *"I would like to go on the 6th"*, the current month is taken by default (or the next month if the 6th has already past).

Interpretative rules transform imprecise values given by the user into appropriate ones used by the system. For example, the utterance *"I want to leave this morning"* is understood as *"I want to leave between 6 am and 12 noon today"*.

Semantic frames corresponding to the current utterance are then completed using the **dialog history** in order to take into account all the information previously given by the user: departure and arrival cities, date of travel, etc..., as well as the questions posed by the system. These questions are saved as part of dialog history, the **generation history**. The generation history enables the system to avoid repeating itself and to interpret the user's responses to system initiatives so as to resolve certain ellipses. For example, without the dialog context a city name can be ambiguous. The generation history is used to obtain the correct interpretation.

4 Speech Synthesis

Since the average user of a telephone information service cannot be expected to be familiar with synthetic speech, nor to be tolerant of poor quality output, the vocal response must be very natural and highly intelligible. We use an approach combining simple playback of pre-recorded messages for fixed responses with synthesis by concatenation of pre-recorded units for variable responses. In order to limit the monotony of the system, we select among several formulations of each response. For variable responses, synthesis is performed by concatenation of the dictionary units (Lamel et al., 1993) according to the text string provided by the message generator. There are about 2000 variable-sized units which are stored in a dictionary. The units correspond to short carrier phrases and individual words for city names, dates, times, numbers, etc. In order to ensure natural prosody, all units are recorded in complete sentences, and unit boundaries are located by automatically aligning the text with the acoustic signal. Each text entry in the dictionary may be associated with several signals, so as to be able to generate different contextual variants. The sequence of speech units for a given text are selected from the stored entries using dynamic programming to optimize the overall quality of the synthesized message, taking into account the phonetic and/or word context, the pitch of successive units, and punctuation markers.

5 Dialog Management

The dialog manager ensures the smooth interface between the user and the computer. The dialog manager maintains the dialog history used in contextual understanding to complete the semantic frame. A mixed-initiative dialog strategy is used where the user is free to ask any question at any time. The system will prompt the user for information needed for database access, however, the user is free to supply additional or different information than was requested by the system.

The completed semantic frame is used to generate an SQL-like request to the database management system, RIHO. Interpretative and history management rules are applied prior to generation of the DBMS request, and post-processing rules are used to interpret the returned information before presentation to the user. For example, in order to provide a more cooperative dialogue, the system relaxes constraints on the departure time when no train is found corresponding to the user's request. In this case the system will return the closest train(s) to the specified time.

Since there is no visual support in the telephone communication, response generation plays an important role in the overall system. Response generation is complex because if too much information is given, it may be difficult for the user to extract the important part. In contrast, if not enough information is returned, the interaction will take longer, as the user will need to ask for more detailed or additional information. The system responses depend on the dialog context and on the information returned from the database management system. Careful attention has been paid to construct texts containing the appropriate information and to generate natural sounding utterances (Bennacef et al., 1996).

5.1 Dialog Structure

The information retrieval dialog is divided in three phases (Bennacef et al., 1995;1996): the <**opening formality**>, the main <**information exchange**> and the <**closing formality**>. Each dialog is structured into a hierarchy of sub-dialogs with a particular functional value. This value may concern the task to be accomplished, the dialog itself, or the metadialog. Subdialogs concerning the task are application-dependent, in so far as the information exchanged includes values directly related to the task. The metadialog corresponds to the parts of discourse which do not directly concern the information enquiries, but concern the way communication is handled.

S:	Welcome to RailTel...		<**opening**>
U:	I would to go to Marseille tomorrow		<**information exchange**>
S:	What is your departure city?	< *precision* >	
U:	Lyon, around 8 in the morning		
S:	There is a train from Lyon to Marseille at 8:35 am tomorrow		
U:	Thank you, goodbye		<**closing**>
S:	Goodbye		

We have combined formal grammars with the theory of speech acts in order to formalize the dialog structure. The dialog is modeled with a set of rules (Bennacef et al., 1995), where the grammar non-terminals correspond to subdialogs, and terminals correspond to dialog acts. Some example rules for initiating different subdialogs are given below. Each rule generates a dialog act which controls the opening, closing and message generation of the subdialog.

Opening subdialog: The system initializes the dialog with a welcome message and an introductory prompt. If the user responds with a greeting, the system asks a specific question (*"what information do you need?"*) to guide the user via a *restart subdialog.*

Precision subdialog: If the semantic frame is incomplete with respect to the information required for database access, the system asks the user to supply the needed information.

Explanation subdialog: If the user does not respond to a system request for information, but instead asks for an explanation (*"what is a youth fare?"*), an explanation subdialog is initiated.

Reformulation subdialog: If the semantic analyser is unable to build a semantic frame, the dialog manager asks the user to repeat the previous request with a reformulation message

(*"I am sorry, I have not understood, can you please repeat that?"*).

Confirmation subdialog: If an incoherency is detected in the semantic frame, the dialog manager attempts to resolve the problem. For example, if someone says (or the system has understood) *"I want to go from Paris to Paris"* the system informs the user that the departure and arrival cities are the same, and asks the user first for the departure city and then for the arrival city.

Closing subdialog: When the user closes the dialog with a politeness form, the system responds by thanking the user for having used the service.

Metadialog: In telephone-based dialog, messages are important to keep the user informed and online. For example, if the database access time is long, a *hold-on sub-dialog* generates the message (*"Hold-on please, we are trying to satisfy your request"*) to inform the user that they need to wait.

5.2 Dialog Strategies

The spoken language system uses a mixed-initiative dialog strategy, where the user is allowed to ask any question, at any time. However, in order to aid the user, the system prompts the user for any missing information needed for database access. Experienced users are thus able to provide all the information needed for database access in a single sentence, whereas less experienced users tend to make shorter requests, allowing the system to guide them. Example dialogs solving the first scenario in Figure 5 are given in Figure 4.

System Opening greeting

Expert *I'd like to know the time of a direct train to Lille, leaving Paris around 10am on March 14th.*

System Opening greeting

Novice *I would like to go to Lille.*

System What city are you leaving from?

Novice *I'm leaving from Paris.*

System What date do you wish to travel?

Novice *March 14th.*

System What time of day do you want to leave?

Novice *Oh, I guess around 10.*

Figure 4: Example dialogs for expert and novice users solving scenario **A** in Figure 5.

Another strategy of the system is to never give a negative response to the user, unless the information is really not available. To do so the system relaxes the constraints provided by the user in order to propose a solution. For example, if the time period specified by the user is too restricted, the system suggests the closest train to the specified departure or arrival time.

Another important issue is management of the dialog history. This requires adding and removing information from the history as a function of the user's response and the result of database access. A set of rules determine which constraints previously specified by the user should be forgotten when, so as to provide a natural and flexible dialog. The main idea is to attach to each constraint a set of other constraints, i.e. *functional dependencies*. Each time the user modifies a constraint, all dependent constraints are removed from history. For example, if the user changes the departure-city, the system removes all linked constraints in the history such as the arrival-city, departure-time, etc..., except for the departure-day. If the user explicitly changes his request, by asking for example about all trains, the system forgets all previously specified information, with the exception of the departure and arrival cities, and the date of travel.

6 Message Generation

In contrast to the MASK kiosk where different media are used to return information to the user, the only possibility is to return information orally. We try to generate responses which provide the essential information in a concise, easily understood form. The generation history is used to avoid repeating information already returned to the user. There are risks of boring the user by repetition and confusing the user if not enough information is given.

Different types of responses can be generated according to the dialog structure: system presentations, prompts (hold-on sub-dialog), restarts, requests for specific information (precision sub-dialog), responses, reformulations, confirmations and domain explanations. The response generator is based on a formal grammar, where non-terminals are conditioned by the dialog context. At each user dialog act, the response generator builds a sentence, filling gaps from the content of the current semantic frame, the dialog history and the DBMS response. Careful attention has been paid to construct natural sounding sentences that contain the appropriate contextual information. When possible, the information is summarized in a single sentence.

The top level grammar rules for interaction with the user are:

- If there are more than 10 possible trains, inform the user of this and ask for additional information about the time period to further limit the number of possibilities.
- If there are between 4 and 10 trains, tell the user the number of trains, giving the departure time and type (or fare) for the first and last trains. Ask the user to provide a more precise departure time.
- If there are 3 or fewer trains, return the departure time, type (and optionally fare) for each train.
- To obtain more information, such as the train number, changes or services, the user must select a single train.

7 Field Trial Methodology

In this section we summarize the field trial carried out for the RAILTEL project (Lamel et al., 1996). The methodology was jointly defined by the partners and used to evaluate the three prototype systems (RailTel, 1995a). The methodology specified the number of subjects, the scenarios types, and the objective and subjective performance measures.

The LIMSI RAILTEL information system was accessible 24 hours a day via a toll-free number, with a single telephone line. For legal reasons a recorded message was presented at the start of each call, informing the caller that their voice would be recorded for the purposes of research and development, and that if they did not agree to be recorded, they should hangup.

Scenario A
Vous recherchez un train au départ de *[ville_A]* et à destination de *[ville_B]*, [*date*] à [*heure*].(You want to find out the departure time of a train from *[city_A]* to *[city_B]*, on [*date*] at [*time*].)

Ex. Vous recherchez un train direct de Paris à Lille, le 14 mars, partant à 9 h.(You want to take a direct train from Paris to Lille on March 14th leaving at 9 am.)

Note that city_A *and* city_B *must be connected by a direct train. The time and date of travel are specified.*

Scenario B
Trouvez l'heure d'arrivé d'un train en provenance de *[ville_A]* et à destination de *[ville_B]*, le [*date*] entre [*heure1 et heure2*].(Find the arrival time of an *[time-period]* train from *[city_A]* to *[city_B]*, *[relative-date] [time-period]*.)

Ex. Vous voulez connaître l'heure d'arrivée d'un train en provenance de Lyon et à destination de Grenoble mercredi prochain entre 11 heures et midi. (You would like to know the arrival time of an evening train from Lyon to Grenoble next Wednesday between 11:30 and noon.)

Note that traveling from city_A *to* city_B *must require a change of trains. The time and date of travel are specified in general terms.*

Figure 5: Commonly defined scenarios used in the field trials.

100 subjects were recruited from 3 sources. The 77 subjects recruited by LIMSI responded to a newspaper announcement and were paid for their participation. The remaining subjects were employees (or their family members) of the SNCF (14 callers) or the Vecsys company (9 callers). Each subject was asked to make a single call to the system (solving a single scenario of type **A** or type **B** shown in Figure 5), and to complete the enclosed questionnaire immediately after interacting with the system. Scenarios of type **A** supplied the user with an exact date and time of travel, and are representative of relatively simple, but frequent, information requests. In the type **A** scenarios the two cities were connected by a direct train. The type **B** scenarios allowed more flexibility on the part of the user, as well as a range of interpretations since the time and date of travel were specified only in general terms. The need to change trains was included to assess the response generation and synthesis components on longer, more complex sentences. For each kind of scenario, at least six different formulations were used. For example, another wording for scenario **A** is: *You want to go from Paris to Lille on March 14th, leaving around 9 am.* Combining different town names, dates and times, 50 different scenarios of each type were generated.[2]

[2] The subjects recruited by LIMSI completed 5 calls to the system, the first call was used for the field trial. Three extra scenarios types were designed for data collection purposes by changing the presentation style, and having callers ask for additional types of information, such as fares and train services. In some cases we asked subjects to solve scenarios involving concepts not yet handled by the system. This enabled us to collect data for a wider variety of situations, and to see how users reacted when the system was unable to provide them the information they wanted, such as for example, when a station or city-name was not known to the system. This data will help us to develop ways to detect such situations.

After finishing the call, each subject completed a questionnaire to gather their immediate impression of the prototype system. The questionnaire, which was designed in coordination with the other partners, contained 20 commonly agreed upon statements to assess the user's opinion about the system. The polarities of the statements were balanced for negative and positive assessment. In addition to the standard questionnaire, we asked subjects what they considered to be the good aspects of the system, how it should be improved, and whether or not they would use such a potential service. Information was also obtained about the subject's travel habits (how often they travel by train, how they obtain their ticket) and their computer experience.

The common objective performance measures were the overall call duration and the number of turns, and the dialog success rate. It was also agreed to compare performance measures for the system components and to identify sources of dialog failure.

8 Field Trial Results

The field trial results are based on the first 50 calls of each type for which a completed questionnaire was returned. Table 1 provides general information about the callers. Although no specific selection was made to balance for gender or age, there are roughly 50% callers of each sex. Over 90% of the callers are younger than 50 years old. The recruitment origin of the callers may also reflect their experience. For example, those recruited by LIMSI are not expected to have had any experience with vocal servers, computers nor any particular travel habits. The 14 subjects recruited by the SNCF can be expected to have a good knowledge of the rail system, and to be frequent travellers. Those recruited by VECSYS may be expected to be somewhat familiar with computers and may have had experience with voice technology.

	Sex		*Age*		
Scenario	*male*	*female*	< 25	$25 - 50$	> 50
A	30	20	15	31	4
B	22	28	21	26	3
A+B	52	48	36	57	7

Table 1: Field trial sample overview: gender and age of subjects.

In addition to global performance measures, a multilevel evaluation is used, which distinguishes errors at 3 different levels in the system: recognition, understanding and dialog. The results given below are all based on 100 calls, 50 of each scenario type.

8.1 Global Evaluation

The average call duration and the number of pairs of turns, where a pair includes both the subject's query and the system response, are shown in Table 2. The average dialog duration is 193 secs for type **A** scenarios and 245 secs for type **B** scenarios.[3] The longer duration for

[3] The duration of a turn pair is measured from the start of user's speech until the end of the system's response. The long average turn duration of almost 50s is due to several factors. First, there is a fixed duration of 25s due to the introductory message. Second, the version of the recognizer used in the field trial

the type **B** scenarios is correlated with the larger number of turns, 5 compared to 3 for type **A**. This is primarily due to the refinement of the times for scenario **B**, which were specified in general terms. The overall dialog success rate was 72%, where a dialog was judged successful if the subject obtained the correct information for the given scenario. The type **A** scenarios were apparently easier for the subjects with 76% being successfully completed, compared to 68% for scenarios of type **B**.

Scenario	*#Calls*	*Turn pairs per call*	*Duration*	*Success Rate*
A	50	3	193 secs	76%
B	50	5	245 secs	68%
A+B	100	4	219 secs	72%

Table 2: Objective measures for field trial data.

Overall, there was a recognition error in 34.8% of the queries. These errors did not necessarily result in a failed scenario, as the scenario failure rate is 28%. For the unsuccessful calls, 80% of the failures are due to recognition and understanding errors, 14% can be contributed to dialog management, and the remaining 6% result from information retrieval errors. (Databases access was not specifically evaluated in the field trials, as this step was assumed to be error free.)

A multilevel error analysis has been carried out on the field trial data distinguishing errors due to recognition, understanding and dialog. Each level is evaluated by separating out the errors caused at the current level from errors propagating from the lower levels. Thus, to evaluate the understanding level we separate out errors due to recognition errors from those arising from the understanding component. Similary, the dialog is evaluated by distinguishing those errors due to the recognition and understanding levels and those occuring at the dialog level.

8.2 Speech Recognition Performance

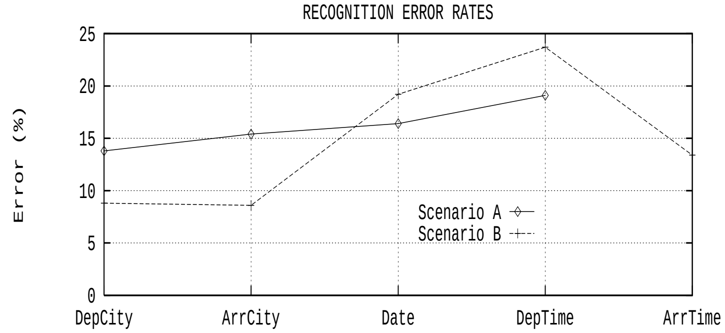

Figure 6: Recognition error rate as a function of slot type for the 100 calls.

was real-time but not time-synchronous, and awaited the detection of the end of speech (a silence of 0.5s) before processing the input. Finally, there are the database access times (about 5s) and for the oral system response times, which can be as long as 15s.

The speech recognition component was evaluated on an independent set of test sentences, and has a word error of about 18%. However, this number can be misleading as the word accuracy measures all differences between the exact orthographic of the query and the recognizer output. Many recognition errors (such as masculine/feminine forms, or plurals) are not important for understanding.

Therefore, we evaluated the recognition performance for the slots relevant for understanding. There were a total 1284 input attempts in the 100 calls. 16 of these inputs were rejected (1.2%), of which 6 were empty and 2 contained only a telephone tone. For the remaining queries the percentage of slots incorrectly recognized are shown in Figure 6 for the two scenario types. In each case, the number of erroneous slot instantiations are divided by the total number of instantiated slots of that type after literal understanding. The number of slot instantiations for the different slot types are given in Table 4. Due to the way the scenarios were defined, there are no **ArrTime** instantiations for the type **A** scenarios.

The slot recognition error is on the order of 10-15% for **DepCity** and **ArrCity** and includes errors due to misrecognition of the actual city name and as well as errors on premarkers signaling "to" or "from". For dates and times the main errors are due to the insertion of extra digits, such as "12:30" (douze heure trente) being recognized as "12:37" (douze heure trente sept). The type **A** scenarios had more recognition errors on cities, while type **B** had more errors on dates and times.

8.3 Spoken Language Understanding Performance

To evaluate the understanding performance it is necessary to differentiate errors due to the understanding component from recognition errors. Table 3 shows the recognition and understanding query error rates for scenarios **A** and **B**, calculated by averaging the error rates for all instantiated slots. For each semantic frame, all slots which are incorrectly instantiated are marked with the error source, recognition or understanding. It is then straightforward to compute the incorrect slot instantiation rate (due to recognition or understanding) for the semantic frame by simply dividing the number of erroneous slots by the total number of instantiated slots.

Scenario Type	*Recognition*	*Understanding*
A	23.2%	10.7%
B	20.0%	6.0%
A+B	21.6%	8.4%

Table 3: Average slot understanding error rates per semantic frame.

There are more than twice as many understanding errors caused by misrecognitions, than are made by the caseframe parser. For example, a recognition error on a city name systematically results in an understanding error. These errors are usually corrected in the ensuing dialog and do not cause the dialog to fail, unless for example when the desired city name is outside of the recognition vocabulary.

Table 4 shows the percentage of slots not understood for the two types of scenarios. The per slot understanding error is very low for the departure and arrival cities as almost all of the errors are caused by recognition errors. An example of an understanding error on a city

	Slot Type				
Type	*DepCity*	*ArrCity*	*Date*	*DepTime*	*ArrTime*
A *#slots*	72	69	202	264	-
Und.Err (%)	0.0%	0.0%	8.8%	17.2%	-
B *#slots*	98	96	144	191	142
Und.Err.(%)	0.2%	1.1%	0.0%	24.4%	17.3%

Table 4: Total number of slot instantiations for each slot type after literal understanding and the understanding error rates for each type of slot. There are a total of 607 slots for type **A** queries and 671 slots for type **B** queries.

occurred when a new formulation was observed for the first time during the field trial. The phrase "à destination de Paris" instantiated the slot **Departure-City:** for Paris instead of the slot **Arrival-City**. The understanding errors for arrival and departure times are on the order of 20%. Not all understanding errors are important for dialog success, for example, interpreting time period as "around 10 pm" instead of "after 10 pm" may not affect the information obtained from the database, and therefore has no effect on the dialog. It has been our observation that such minor understanding errors pass unobserved by the user, whereas more important understanding errors will lead to longer dialogs, as the user tries to correct the error.

The dialog is evaluated by looking at the response of the system. When the response is judged to be incorrect, the source of the error is indicated as recognition and/or understanding (reco/und) or dialog management. The dialog errors are calculated by the ratio of incorrect responses and the total number of system responses. The Table 5 shows the dialog error rates for both scenario types. Errors due to recognition and understanding errors (Rec/Und) are more important for type **A** scenarios than for type **B** scenarios. This is due to digit recognition errors which were more common in the type **A** scenarios which specified explicit dates and times. These errors generally did not result in dialog failure, as the user usually corrected the error in a later turn, successfully completing the call. The higher number of dialog errors for type **B** scenarios were due to scenarios which required the user to depart the previous evening in order to arrive at the specified time. This was not correctly handled in the system and implied dialog failure. The error was actually due to a problem in post-processing of the retrieved information, but we have considered it to be a dialog error as the response of the system was not correct in the context of the user's query. The field trials also turned up a database connection problem, that was corrected early on (column DB).

Scenario	*#*		*Cause of error*		
Type	*Dialogs*	*Correct*	*Rec/Und*	*Dialog*	*DB*
A	50	58.5%	34.3%	0.9%	6.3%
B	50	60.5%	29.2%	10.3%	-

Table 5: Source of error per system response. Rec/Und: recognition or understanding error, Dialog: dialog error, DB: database access error.

An example of a dialog error is illustrated in Figure 7. This occurred when the user asked for "the next train" (the train already returned by the system was at 12:33). The query was both correctly recognized and understood, but an incorrect response was given to the user. The system returned all trains after 12:33 instead of giving just the next one at 13:06, and asked the user to provide a more specific departure time.

U: I'd like to know the next train (0zl005)
`[train already given:  12:33]`
<train-time>
    order-stand: next. (1)
    }
S: There are more than 10 trains from Arras to Lille-Flandres Thursday 28/09 corresponding to your request. The first leaves at 13:06 and the last at 21:54. Please give a more specific departure time. (D-0)

Figure 7: Example of a dialog error.

8.4 User Evaluation

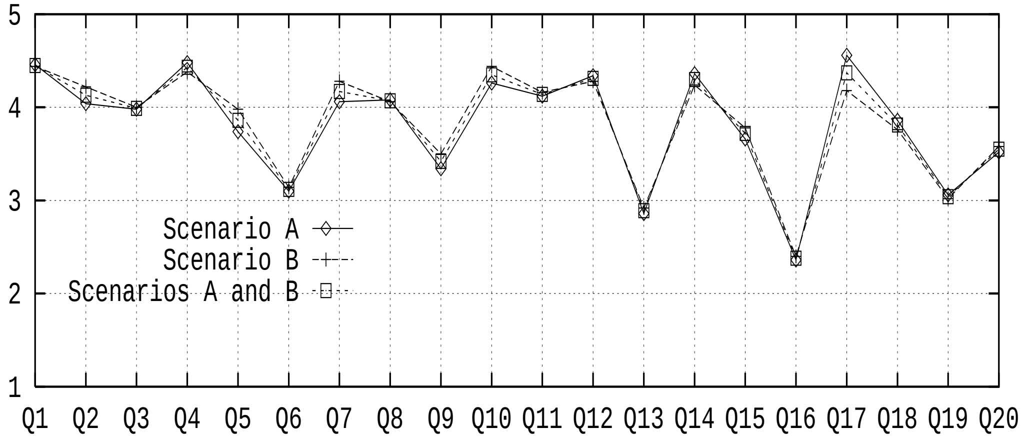

Figure 8: Overall results for usability profiles as a function of scenario type. Q1: ease-of-use, Q2: confusing, Q3: friendliness, Q4: complex, Q5: use again, Q6: reliability, Q7: control, Q8: concentration, Q9: efficiency, Q10: fluster, Q11: too fast, Q12: stress, Q13: prefer human service, Q14: complicated, Q15: enjoyable, Q16: needs improvement, Q17: politeness, Q18: information obtained, Q19: faster than human, Q20: understood

The responses to the 100 questionnaires were used to generate "usability profiles". The overall user assessment on a scale of 5 is shown in Figure 8, individually for the two scenarios types and combined. Both scenarios types were rated at the same level by the subjects. Although not shown in the figures, there is a slight tendancy of younger subjects to assess the system more favorably than the older subjects, which is likely to be correlated with a larger familiarity of younger subjects with computers and automated services. Only small differences were observed according to the recruitment source or sex of the subject. Female subjects were slightly more favorable in their assessment than male subjects, but they also expressed less interest in using such a service. Due to the small number of subjects these differences are not significant.

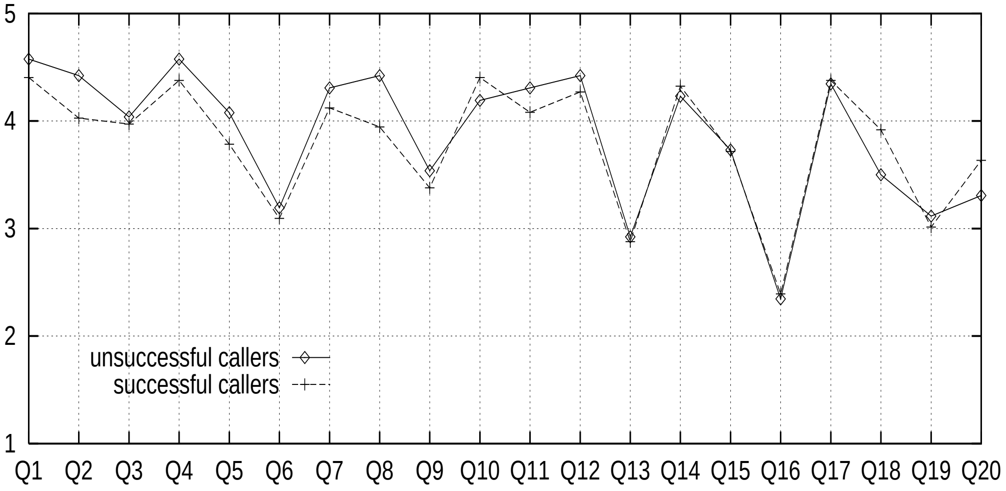

Figure 9: Results for successful and unsuccessful callers.

In Figure 9 the responses of subjects are divided into groups corresponding to successful and unsuccessful calls. We observe that these two groups rated the system at the same level.

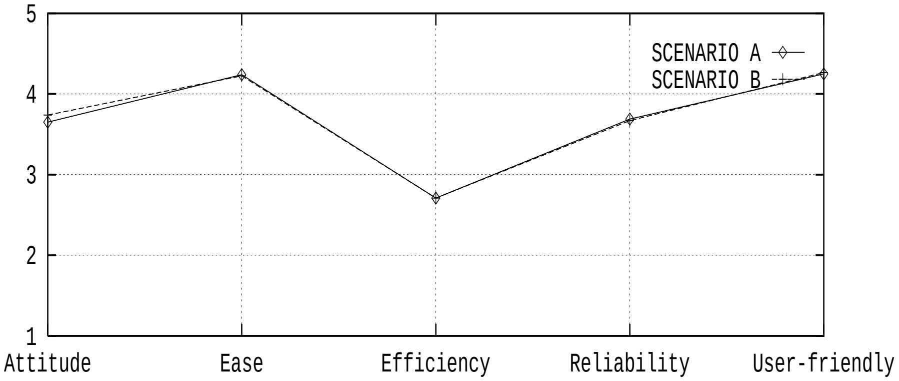

Figure 10: Overall results for the 5 categories.

The questions were grouped into the following 5 categories: attitude (A: 5, 12, 13, 15), ease of use (EU: 1, 2, 4, 8, 14), efficiency (E: 9, 16, 19), reliability (R: 6, 18, 20) and user-friendliness (UF: 3, 7, 10, 11, 17). Figure 10 shows the overall results for these 5 categories. While there is a tendency of subjects to assess the system favorably (EU and UF), they don't find it particularly efficient (E), and some subjects doubted the reliability of the system (R). This is likely to be linked to the responses to question Q16, that the system needs to be improved.

It is possible to relate these results with the two last questions in the questionnaire, which asked the subjects to specify what they liked about the system and to suggest improvements.

Concerning the negative points, subjects expressed concern about the reliability of the information returned by the system. Concerning the positive points, subjects commented on the user-friendliness and the speed of the system, several judging it to be faster to use than a human service.

9 Discussion

The goal of the RailTel field trials was to assess the potential of telephone services based on existing spoken language technology. The field trial methodology was a compromise taking into account the time constraints and different state of advancement of the prototype systems. The two scenario types reflect the desire of the partners to assess the system components: speech recognition, understanding, dialog, response generation and speech synthesis, in a controlled manner; many other combinations could have been envisioned. In order to allow users to speak in a natural manner, different formulations for dates and times of travel need to be recognized and understood. The scenario types were designed to test the recognition and understanding capabities for numbers and dates (type A) and less explicit and relative travel specifications (type B). The type B scenarios entailed generating more complex responses to inform the user of the information concerning the change of trains (the station and time).

An evident weakness of the use of scenarios for the field trials is that the subjects, while plausible eventual users, were not accomplishing a real task. This means that the subjects were not particularly concerned about the response given by the system, as this information was not really needed. While we had discussed the possibility of carrying out a field trial without using scenarios, we decided as a consortium, that this approach would make it extremely difficult to compare the results. If the test subjects had been free to request any travel information we would have had problems ensuring a diversity of requests, and would have had to decide which calls to include if some queries were out-of-domain. Developing techniques to deal gracefully with such calls, while necessary for an real service, was beyond the scope of the prototype systems developed for the project.

The use of scenarios may explain why the user assessment was the same for both successful and unsuccessful dialogs (Figure 9,) and for the two scenario types (Figure 8). Although the type **A** scenarios are easier in the sense that they had a higher success rate and were faster to solve, users rated the system at the same level. A contributing factor can be that we considered a dialog to be a failure if final response of the system differed, even only slightly, from what was specified in the scenario. This judgement was used even if the subject did not exactly respect the scenario. Therefore, subjects may have been happy with a response that we considered erroneous.

A correlation between objective measurements and the subjective assessment (via the questionnaire) was observed with respect to the age of subjects. Older subjects (> 50) expressed less satisfaction with the system, and had higher dialog error rates. This can be partly attributed to the lack of experience of older users with computers and automated services, and also that our training corpus does not include much data from older speakers.

10 Conclusion

We have described a prototype system for access to train travel information over the telephone. This system, based on our Mask system (Gauvain et al., 1995a), was brought up very quickly so as be able to carry out the RailTel field trial with 100 naive subjects. The performance snapshot resulting from the field trial had 72% of the callers successfully completing their scenarios. The failures were due mainly to recognition and understanding errors (80%), with 14% due to dialog management, and the remaining 6% resulting from information retrieval errors. The subject assessment of the service was largely favorable, although there was a clear expression of the need for improvement. Most subjects expressed a potential interest in using such a service. The field trial demonstrated that such services should be easily accepted by the general public.

Evaluation of spoken language systems remains an open research issue. The prototype has been evaluated in terms of global performance measures, component performance, and subjective user assessment. A multilevel approach has been used for evaluation of the system components, which distinguishes errors caused at a given level from errors propagating from the lower levels.

Further developments are now being pursued in the context of the LE project Arise, including a French/English service for the high speed train between Paris and London. Improvements in speech recognition are expected by using more training data for acoustic and language modeling. We are also investigating the use of confidence measures and dialog dependent language models (Popovici and Baggia, 1997) to improve the recognition performance. We are addressing issues related to dialog management, such as modeling user intention, history maintenance, and confirmation strategies. In fact, one of the hardest control problems is to detect that the dialog is finished. Different confirmation strategies, combining implicit and explicit methods are being tested. Within the Arise project, two field trials will be carried out in coordination with the SNCF.

Acknowledgment

We would like to thank the SNCF for providing the information database Riho for use in the RailTel project.

11 References

RailTel (1995a), "Definition of the evaluation methodology for the Field Trials," RailTel/Mais *Project deliverable D4*, Saritel, June.

RailTel (1995b), "Results of Field Trials," RailTel *Project deliverable D8*, November.

G. Adda, M. Adda-Decker, J.L. Gauvain, L. Lamel (1997), "Le systḿe de dicté vocale du LIMSI pour l'v́aluation AUPELF'97," *Proc. Journées Scientifiques et Techniques du Réseau Francophone d'Ingénierie de la Langue de l'AUPELF-UREF*, Avignon, France, pp. 35-40, April.

C. Popovici, P. Baggia (1997), "Specialized Language Models using Dialogue Predictions", *Proc. IEEE ICASSP-97*, Munich, Germany, pp. 815-818, April.

S.K. Bennacef, H. Bonneau-Maynard, J.L. Gauvain, L.F. Lamel, W. Minker (1994), " A Spoken Language System For Information Retrieval," *Proc. ICSLP'94*, Yokohama, Japan, **3**, pp. 1271-1274, September.

S.K. Bennacef, F. Neel, H. Bonneau-Maynard (1995), "An Oral Dialogue Model based on Speech Acts Categorization," *Proc. ESCA Workshop on Spoken Dialog Systems*, Vigsø, Denmark, pp. 237-240, Spring.

S. Bennacef, L. Devillers, S. Rosset, L. Lamel (1996), "Dialog in the RailTel Telephone-Based System," *Proc. ICSLP'96*, Philadelphia, PA, pp. 550-553, October.

R. Billi, G. Castagneri, M. Danieli (1996), "Field Trial Evaluations of Two Different Information Inquiry Systems," *Proc. IEEE IVTTA-96*, Basking Ridge, NJ, pp. 129-134, October (and this issue of *Speech Communication*).

B. Bruce (1975), "Case Systems for Natural Language," *Artificial Intelligence*, **6** pp. 327-360.

Ch.J. Fillmore (1968), "The case for case," in *Universals in Linguistic Theory*, Emmon Bach & Robert T. Harms (eds.), Holt, Rinehart and Winston, Inc.

J.L. Gauvain, S.K. Bennacef, L. Devillers, L.F. Lamel, S. Rosset (1995a), "The Spoken Language Component of the Mask Kiosk," *Proc. Human Comfort & Security Workshop*, Brussels, October. (Published in *Human Comfort and Security of Information Systems*, K. Varghese and S. Pfleger (eds.), Springer Verlag, pp. 93-103, 1997.)

J.L. Gauvain, J.J. Gangolf, L. Lamel (1996), "Speech Recognition for an Information Kiosk," *Proc. ICSLP'96*, Philadelphia, PA, pp. 1672-1675, October.

J.L. Gauvain, L.F. Lamel, G. Adda, M. Adda-Decker (1994a), "Speaker-Independent Continuous Speech Dictation," *Speech Communication*, **15**, pp. 21-37, September.

J.L. Gauvain, L.F. Lamel, G. Adda, M. Adda-Decker (1994b), "Continuous Speech Dictation in French," *Proc. ICSLP'94*, Yokohama, Japan, pp. 2127-2130, September.

J.L. Gauvain, L. Lamel, M. Adda-Decker (1995), "Developments in Continuous Speech Dictation using the ARPA WSJ Task," *Proc. IEEE ICASSP-95*, Detroit, MI, pp. 65-68, May.

S.M. Katz (1987), "Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer," *IEEE Trans. ASSP*, **35**(3), pp. 400-401, March.

L. Lamel, J.L. Gauvain, S.K. Bennacef, L. Devillers, S. Foukia, J.J. Gangolf, S. Rosset (1996), " Field Trials of a Telephone Service for Rail Travel Information," *Proc. IEEE IVTTA-96*, Basking Ridge, NJ, pp. 111-116, October.

L.F. Lamel, J.L. Gauvain, B. Prouts, C. Bouhier, R. Boesch (1993), "Generation and Synthesis of Broadcast Messages," *Proc. ESCA-NATO Workshop on Applications of Speech Technology*, Lautrach, Germany, pp. 207-210, September.

L.F. Lamel, S.K. Bennacef, H. Bonneau-Maynard, S. Rosset, J.L. Gauvain (1995a), "Recent Developments in Spoken Language Sytems for Information Retrieval," *Proc. ESCA Workshop on Spoken Dialog Systems*, Vigsø, Denmark, pp. 17-20, Spring.

L.F. Lamel, S. Rosset, S.K. Bennacef, H. Bonneau-Maynard, L. Devillers, J.L. Gauvain (1995b), "Development of Spoken Language Corpora for Travel Information", *Proc. Eurospeech'95*, Madrid, Spain, pp. 1961-1964, September.

List of footnotes.

Footnote 0: This work was partially financed by the LE MLAP project 63-022 RailTel.

Footnote 1: The continuation of this work is being partially financed by the LE-3 project 4229 Arise.

Footnote 2: The subjects recruited by LIMSI completed 5 calls to the system, the first call was used for the field trial. Three extra scenarios types were designed for data collection purposes by changing the presentation style, and having callers ask for additional types of information, such as fares and train services. In some cases we asked subjects to solve scenarios involving concepts not yet handled by the system. This enabled us to collect data for a wider variety of situations, and to see how users reacted when the system was unable to provide them the information they wanted, such as for example, when a station or city-name was not known to the system. This data will help us to develop ways to detect such situations.

Footnote 3: The duration of a turn pair is measured from the start of user's speech until the end of the system's response. The long average turn duration of almost 50s is due to several factors. First, there is a fixed duration of 25s due to the introductory message. Second, the version of the recognizer used in the field trial was real-time but not time-synchronous, and awaited the detection of the end of speech (a silence of 0.5s) before processing the input. Finally, there are the database access times (about 5s) and for the oral system response times, which can be as long as 15s.

List of Tables

List of Figures