Methodology for Dialogue Design in Telephone-Based Spoken Dialogue Systems: a Spanish Train Information System

Grupo de Tecnología del Habla. Departamento de Ingeniería Electrónica. UPM E.T.S.I. Telecomunicación. Ciudad Universitaria s/n, 28040 Madrid, Spain
lapiz@die.upm.es
http://www-gth.die.upm.es

Abstract
In this paper, we propose a new methodology for designing dialogue managers in telephone-based spoken dialogue systems. This methodology comprises five steps: database analysis, design by intuition, design by observation, simulation and iterative improvement. At each step, several measures to evaluate the designing alternatives are presented. We introduce confidence measures in recognition to define an efficient confirmation strategy in each case. The use of user-modeling techniques adapts the system to the user ability. This methodology is applied for designing a telephone-based system that provides rail travel information for the main Spanish intercity connections, including timetables, simulated fares and reservations. With 30 users completing 4 scenarios, the average duration for a fully automatic call was 204 seconds. The users validated the applicability and usability of the system with a global score of 3.9 (out of 5).

1. Introduction
The important improvements in the Speech Technology allow developing systems for real conditions. At this point, telephone-based spoken dialogue systems have appeared as an important field for incorporating these technologies. In the last decade, a great amount of restricted domain systems have been developed for travelling information and tickets reservations services [1][2]. An other important project is the DARPA Communicator http://fofoca.mitre.org. [3][4]. These services enable a natural conversational interaction by telephone callers to access information about airline flights, hotels and rental cars. We propose a new methodology for designing the dialogue managers in these kind of systems that has been applied to a train timetable information system for the main Spanish intercity connections.

The paper is organised as follows. Section 2 describes the first steps of the methodology: database analysis and design by intuition. Section 3 shows the details of the observation analysis step. In section 4, the experiments for the simulation phase, using a Wizard of Oz approach, are presented. The iterative improvement step is described in section 5 where we introduce confidence measures for confirmations managing and user-modelling techniques for adapting the system to the user ability level. Section 6 presents the evaluation of the fully automatic system. Finally, in Section 7 we draw the main conclusions.

2. Database Analysis and Design by intuition
In the database analysis, we aim at describing the information the database contains to offer the service. This description consists of building an Entity-Relationship Diagram (E-R) that shows a semantic representation of the data. Sometimes the database has to be filled with additional information in order to provide a specific service. In this diagram, the main entity sets, their attributes and keys (attributes that define uniquely an element in an entity set), and entity set relationships must be defined. The E-R diagram is not unique and the solution depends strongly on the system designer and the service. After this analysis, we propose a “brain-storming” (design by intuition) over a specific E-R obtained from the database analysis. The main target at this step is to propose alternatives. These proposals are concerned with the goals to provide (parts of service such as timetable information, reservations, fares, etc.), the sequence of offering them and the items (pieces of information obtained from the user) that are needed to satisfy each goal. Generally, at this step any dialogue system is implemented. The result of this analysis is a table or form including all the possibilities/alternatives. This table will be used at the next step of the methodology to compute the frequency of each alternative: the goals required, the information provided in each case, etc.

3. Design by Observation
The design by observation consists of analyzing user-operator dialogues in a similar service and tracking off different events observed. This design evaluates and measures the impact of the alternatives proposed in previous step:

About goals:
a) Which goals are most frequently required by the user and the goals sequence. The results of this analysis must be the number of times that a specific goal is required and its position: it is the first goal in the interaction, the second, etc...

Considering 100 call transcriptions we got:

<table>
<thead>
<tr>
<th>Goal Analysis</th>
<th>%</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1st</td>
<td>2nd</td>
</tr>
<tr>
<td>Timetable</td>
<td>64</td>
<td>57</td>
</tr>
<tr>
<td>Round trip Timetable</td>
<td>20</td>
<td>-</td>
</tr>
<tr>
<td>Fares</td>
<td>46</td>
<td>6</td>
</tr>
<tr>
<td>Reservations</td>
<td>26</td>
<td>14</td>
</tr>
<tr>
<td>Train frequency</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Itinerary</td>
<td>14</td>
<td>8</td>
</tr>
<tr>
<td>Bargains</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Others</td>
<td>12</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 1: Goals analysis.

Several goals can appear in the same call. For our system we have considered the goals that appear in more than a 10% except Itinerary and Others because we have no information about itineraries in the database and Others contains very different goals and none of them reaches 10%.
b) The information provided by the operator to satisfy each goal. How much information is provided and which data is selected.
About Items:
- a) Items needed to satisfy each goal and the sequence for asking them (similar to the goals analysis).
- b) Analyzing the different ways to specify an item value by the user and its importance.

About the Negotiation:
There is a "negotiation" when the user have to choose one of the options proposed by the system:
- a) what information helps the user decide?, (p.e. number of times the user consider the fares, the departure time, the trip duration, etc... when choosing the final option).
- b) Choose the negotiation strategy: it can consist of presenting the best option and let the user ask for the previous/next one or present several alternatives. In this case, it is important to analyze the number of alternatives the operator gives at the same time (the user can manage simultaneously). With this analysis, only human-human interactions are considered and they are different from human-system ones. Because of this, high level dialogue characteristics can be learnt, but specific behaviors when interacting with an automatic system are not detected. In the other hand, this analysis permits to evaluate alternatives without having implemented any system.

4. Design by Simulation

Now, the specific characteristics of human-system interactions are analyzed by simulating the system with the Wizard of Oz approach. In this simulation, we focus on the design of the dialogue flow, the questions to ask and the information needed to satisfy each goal. Dialogue alternatives are implemented and evaluated. In this evaluation, several users call the system completing several scenarios and fill a questionnaire (15 users completed 6 scenarios). The evaluating measures come from system annotations and user answers in the questionnaire. The user is asked about subjective aspects difficult to measure by means of the system.

About goals:
- a) To validate the goal sequence defined and the goal cover.
  - System: Number of times a goal is required, Time and Number of questions to satisfy a goal.
  - Questionnaire: new goals suggested.
- b) For the item sequence: in some parts, several item sequences are proposed and randomly selected in each call.
  - System: audio files with the user answer. With a first version of the recognizer, it is possible to evaluate the Sequence Recognition Rate calculated as the product of all independent item recognition rates.
  - Questionnaire: asking to the user for his/her preference.

About the Negotiation:

Users prefer to manage 3 options at the same time. In this case, negotiation takes more time and less questions. We decided to keep the negotiation in a 3 by 3 basis reducing the information to the train type plus the departure and arrival times.

5. Design by Iterative improvement

At this step, we implement the first version of a fully automatic system. This step consists of an iterative process in which, at the same time the system is tested and modified. With the Wizard of Oz, a first version of the dialogue flow was defined. To make a fully automatic system it is necessary to design the confirmations strategies. These mechanisms must be designed taking into account the recognition confidence, so an analysis of confidence measures is required.

5.1. Confirmation Strategies

Depending on the number of items to confirm:
- One item. (i.e. "Have you said Madrid?")
- Several items. (i.e. "Do you want to go from Madrid to Sevilla?")

Depending on the confidence obtained in recognition [5]:
- Explicit confirmation: the system confirms one or several item values by asking directly. (i.e. "I understood you want to depart from Madrid. Is that correct?")
- Implicit confirmation: the system does not permit the user to correct, it only informs him/her. (i.e. "You want to leave from Madrid. Where are you arriving at?")
- Semi-implicit confirmation: it is similar to the implicit confirmation but the user can correct (i.e. "You want to leave from Madrid. In case of error, say correct, otherwise, indicate your arrival city").
- No confirmation: when the system does not inform about the value recognized (i.e. in yes/no questions).
5.2. Confidence Measures in recognition

The recognition module used in our system is a very-large telephone speech recognizer, that can recognizes isolate words and simple expressions such as "On Monday", "Next week" or "In the morning". The recognizer [6] is based on a hypothesis-verification approach. The best features considered for confidence annotation are concerned with the verification step and are based on the ideas proposed in [7]:

- **First Candidate Score:** acoustic score of the best verification candidate.
- **Candidate Score Difference:** difference acoustic score between the 1st and 2nd verification candidates.
- **Candidate Score Mean and Variance:** average score and variance along the 10 best candidate names.
- **Score Ratio:** difference between the score of the phone sequence (hypothesis stage) and the score of the best candidate name (verification stage).

All the features are divided by the number of frames. We have considered a Multi-Layer Perceptron (MLP) to combine the features. Considering a database with 2,204 cases, 39.1% of wrong recognized words are detected at 5% false rejection rate, reducing the minimum classification error from 15.8% (recognition error rate) to 14.0%.

5.3. Confirmation Mechanism design

For designing the confirmation mechanism is necessary to plot the correct words and errors distributions depending on the confidence value and to define different confidence thresholds. Let us see Figure 2.

![Confidence](image)

**Figure 2. Bottom detail for Errors and Correct words distributions depending on the confidence value.**

We have defined 4 levels (3 thresholds) of confidence:

1.- **Very High confidence:** the number of correctly recognized words is much higher than the number of errors.
2.- **High confidence:** the number of correctly recognized words is higher than the number of errors.
3.- **Low confidence:** Both distributions are similar. The system is not sure about the correctness of the recognized word.
4.- **Very Low confidence:** In this case, there are more errors than correct recognized words, so we reject the recognized word and the system asks again.

Let us consider the example of the departure and arrival cities step. Let us define CL(D) and CL(A) as departure/arrival city Confidence Levels (CL). Depending on the CL we implement the following confirmation strategies:

- **CL(D)=1 and CL(A)=1:** implicit confirmation of both items. "You want to travel from Madrid to Sevilla, When do you want to leave?"
- **CL(D)=2 or CL(A)=2:** explicit confirmation of both items. "Do you want a Madrid-Sevilla trip?"
- **CL(D) = 3 or CL(A) = 3:** implicit confirmation of each item. "Do you want to travel from Madrid?"
- **CL(D) = 4 or CL(A) = 4:** rejection of each item. "Sorry, I have not understood. Where are you departing from?"

Under hard conditions, the system asks the user to spell the city name (new confidence analysis is needed in this case [8]).

When implicit confirmations are incorporated to automatic systems, it is necessary to define some mechanisms to permit the user to recover from system errors.

**START OVER:** with this command the system permits the user to start from scratch. Instead of resetting all the items, we begin confirming groups of items explicitly (dialogue steps).

When one of this group is not confirmed, the system starts from that point:

- S: "The selected option is an Intercity train..."
- U: "start over" (The system is in the level 3)
- S: "Let us start the query"
- U: "Do you want a Madrid-Barcelona trip?" (The system recognizes "in the evening")
- S: "You want to travel from Madrid to Sevilla, When do you want to leave?"
- U: "No"
- S: "Do you want to leave this week, next week or later?"

**CORRECT:** When the system makes a mistake and consider a wrong item value as right, the user can correct the system by saying this command at any point of the dialogue. In this case, the system asks again the last confirmed item.

5.4. User-Modeling

The user modeling technique used is based on the ideas proposed in [9]. We consider 4 user skill levels. Depending on the current level, the prompt sentences are clearer (p.e they contain more information about how the user should answer, possible values, etc...) or the system provides more or less information per time unit. The levels considered are:

- **1st level.** The prompts explains how to interact with the system, the item asked, the possible accepted values and how to specify one of the values, (i.e. for the period of the day: "Please speak after the tone. Say the period of the day you want to travel; in the morning, in the afternoon or in the evening.")
- **2nd level.** The prompts include the item needed, the accepted values for this item and the way to specify them, ("Say the period of the day you want to travel; in the morning, in the afternoon or in the evening.")
- **3rd level.** Only the required item is included in the prompt, ("Say the period of the day you want to travel.")
- **4th level.** The user knows everything and we assume relax the question, ("When do you want to leave?").

Current level depends on the initial state and the number of errors (or positive confirmations) along the interaction. In our case, the system starts at the 2nd level (after providing general explanation on how to interact with the system). When several errors (or positive confirmations) occur it decreases (or increases) the level. The number depends on the current level. Let us see the next example:

- [The system is in the level 3]
  - S: "Say the period of the day you want to travel."
  - U: "after lunch"
- [The system recognizes "in the evening"]
  - S: "Have you said in the evening?"
U: “no”  
S: “The system decreases the level from 3 to 2”  
U: “in the afternoon”

6. Field evaluation

In this evaluation, 30 users called the system for completing 4 scenarios (120 calls). The evaluating measures come from the system annotations and user answers in a questionnaire.

<table>
<thead>
<tr>
<th>Measure</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Call duration (seconds)</td>
<td>204</td>
</tr>
<tr>
<td>Number of questions per call</td>
<td>21.2</td>
</tr>
<tr>
<td>% of implicit confirmations</td>
<td>61.3</td>
</tr>
<tr>
<td>Number of START OVER commands</td>
<td>0.08</td>
</tr>
<tr>
<td>Number of CORRECT commands</td>
<td>0.43</td>
</tr>
<tr>
<td>Duration of Negotiation (seconds)</td>
<td>58</td>
</tr>
</tbody>
</table>

Table 4. Measures calculated by the system.

The 35.4% of the calls asked information of the two legs in a round-trip and the 31.5% completed the reservation for single or round-trips. About the recognition rates, we got more than 95% for small vocabularies (less than 50 words/expressions) (i.e. weekday, options proposed, etc...). For departing and arrival cities, we have considered a 770-words vocabulary. In case of in vocabulary and not rejected answers, we got 90.1% recognition rate, resolving 2.5% of the wrong cases with the second candidate and 3.6% with the spelled name recognizer. In the remaining cases, it was necessary more interactions. In these experiments, we got 4.1% out-of-vocabulary cities, detecting 1.7% with the spelled name recognizer [8]. For the 2.4% remaining, the user hung after several trials.

As we can see the call duration is 204 s, higher than the average duration in the service with human operators (152 s), but similar to others similar automatic services [4].

<table>
<thead>
<tr>
<th>Measure</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>User experience in these kind of systems.</td>
<td>1.8</td>
</tr>
<tr>
<td>The system understand what I say.</td>
<td>3.6</td>
</tr>
<tr>
<td>I understand what the systems says.</td>
<td>4.5</td>
</tr>
<tr>
<td>I get train information fast.</td>
<td>4.0</td>
</tr>
<tr>
<td>The system is easy to learn.</td>
<td>3.9</td>
</tr>
<tr>
<td>In case of error the correction was easy.</td>
<td>3.1</td>
</tr>
<tr>
<td>The system asks me in a logical order.</td>
<td>4.6</td>
</tr>
<tr>
<td>Generally, it is a good system.</td>
<td>3.9</td>
</tr>
</tbody>
</table>

Table 5. Measures (out of 5) obtained from the questionnaire.

At the end, we asked the user preference when obtaining train information: 75.4% of the people preferred the system, 24.6% by web and nobody preferred to go to the station. The dialogue point with more problems was the departing date specification because we have isolated speech recognition and it needs several interactions to get a date. The best score was obtained for the intelligibility of the system. This has been possible thanks to a female synthesis for restricted domain developed in [10].

7. Conclusions

In this work, we propose a new methodology for designing dialogue managers in telephone-based spoken services and has been applied to train information system for Spanish. The first step is a database analysis building an E-R diagram and a design by intuition where we consider a “brain-storming” over the E-R, for proposing different dialogue alternatives and defining a table to evaluate them in further steps. In the design by observation, using human-human dialogues transcription, we can evaluate each of the proposals suggested in the intuition step without any implementation. The limitation is that the human-human interactions are different from the human-system one. The Wizard of Oz approach solves the problem permitting to simulate the human-system interaction.

In the design by iterative improvement, recognition confidence measures are incorporated for defining and managing the confirmation mechanisms. The confidence measures have permitted us to obtain a recognition rate higher than 90% perceived by the user for all the recognition vocabularies. Two mechanisms for error recovering are described: Start-Over and Correct and user-modeling techniques for adapting the system to the user ability.

Considering the final evaluating measures, we conclude that this methodology has permitted us to implement a full automatic system with a good user acceptability. In the field experiments, the call duration was 204 seconds, similar to [4] and the users validated the applicability and usability of the system providing a general score of 3.9 (out of 5).

8. References