MIMUS: A Multimodal and Multilingual Dialogue System for the Home Domain

J. Gabriel Amores
Julietta Research Group
Universidad de Sevilla
jgabriel@us.es

Guillermo Pérez
Julietta Research Group
Universidad de Sevilla
gperez@us.es

Pilar Manchón
Julietta Research Group
Universidad de Sevilla
pmanchon@us.es

Abstract

This paper describes MIMUS, a multimodal and multilingual dialogue system for the in–home scenario, which allows users to control some home devices by voice and/or clicks. Its design relies on Wizard of Oz experiments and is targeted at disabled users. MIMUS follows the Information State Update approach to dialogue management, and supports English, German and Spanish, with the possibility of changing language on–the–fly. MIMUS includes a gestures–enabled talking head which endows the system with a human–like personality.

1 Introduction

This paper describes MIMUS, a multimodal and multilingual dialogue system for the in–home scenario, which allows users to control some home devices by voice and/or clicks. The architecture of MIMUS was first described in (Pérez et al., 2006c). This work updates the description and includes a life demo. MIMUS follows the Information State Update approach to dialogue management, and has been developed under the EU–funded TALK project (Talk Project, 2004). Its architecture consists of a set of OAA agents (Cheyer and Martin, 1972) linked through a central Facilitator, as shown in figure 1:

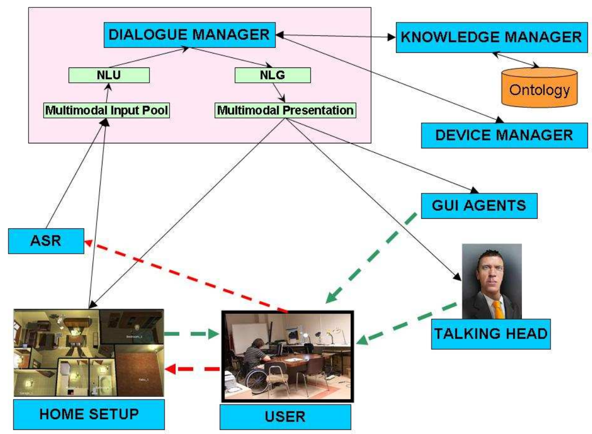

Figure 1: MIMUS Architecture

The main agents in MIMUS are briefly described hereafter:

- The system core is the **Dialogue Manager**, which processes the information coming from the different input modality agents by means of a natural language understanding module and provides output in the appropriate modality.

- The main input modality agent is the **ASR Manager**, which is obtained through an OAA wrapper for Nuance. Currently, the system supports English, Spanish and German, with the possibility of changing languages on–the–fly without affecting the dialogue history.

- The **HomeSetup** agent displays the house layout, with all the devices and their state. Whenever a device changes its state, the HomeSetup is notified and the graphical layout is updated.

- The **Device Manager** controls the physical devices. When a command is sent, the Device Manager notifies it to the HomeSetup and the Knowledge Manager, guaranteeing coherence in all the elements in MIMUS.

- The **GUI Agents** control each of the device–specific GUIs. Thus, clicking on the telephone icon, a telephone GUI will be displayed, and so on for each type of service.

- The **Knowledge Manager** connects all the agents to the common knowledge resource by

Proceedings of the ACL 2007 Demo and Poster Sessions, pages 1–4,
Prague, June 2007.

means of an OWL Ontology.

- The **Talking Head**. MIMUS virtual character is synchronized with Loquendo's TTS, and has the ability to express emotions and play some animations such as nodding or shaking the head.

2 WoZ Experiments

MIMUS has been developed taking into account wheel–chair bound users. In order to collect first–hand information about the users' natural behavior in this scenario, several WoZ experiments were first conducted. A rather sophisticated multilingual WoZ experimental platform was built for this purpose.

The set of WoZ experiments conducted was designed in order to collect data. In turn, these data helped determine the relevant factors to configure multimodal dialogue systems in general, and MIMUS in particular.

A detailed description of the results obtained after the analysis of the experiments and their impact on the overall design of the system may be found in (Manchón et al., 2007).

3 ISU–based Dialogue Management in MIMUS

As pointed out above, MIMUS follows the ISU approach to dialogue management (Larsson and Traum, 2000). The main element of the ISU approach in MIMUS is the dialogue history, represented formally as a list of dialogue states. Dialogue rules update this information structure either by producing new dialogue states or by supplying arguments to existing ones.

3.1 Multimodal DTAC structure

The information state in MIMUS is represented as a feature structure with four main attributes: **D**ialogue Move, **T**ype, **A**rguments and **C**ontents.

- **DMOVE**: Identifies the kind of dialogue move.
- **TYPE**: This feature identifies the specific dialogue move in the particular domain at hand.
- **ARGS**: The `ARGS` feature specifies the argument structure of the `DMOVE/TYPE` pair.

Modality and Time features have been added in order to implement fusion strategies at dialogue level.

3.2 Updating the Information State in MIMUS

This section provides an example of how the Information State Update approach is implemented in MIMUS. Update rules are triggered by dialogue moves (any dialogue move whose DTAC structure unifies with the Attribute–Value pairs defined in the TriggeringCondition field) and may require additional information, defined as dialogue expectations (again, those dialogue moves whose DTAC structure unify with the Attribute–Value pairs defined in the DeclareExpectations field).

Consider the following DTAC, which represents the information state returned by the NLU module for the sentence *switch on*:

$$\begin{bmatrix} \text{DMOVE} & \text{specifyCommand} \\ \text{TYPE} & \text{SwitchOn} \\ \text{ARGS} & \begin{bmatrix} \text{Location, DeviceType} \end{bmatrix} \\ \text{META\_INFO} & \begin{bmatrix} \text{MODALITY} & \text{VOICE} \\ \text{TIME\_INIT} & \text{00:00:00} \\ \text{TIME\_END} & \text{00:00:30} \\ \text{CONFIDENCE} & \text{700} \end{bmatrix} \end{bmatrix}$$

Consider now the (simplified) dialogue rule "**ON**", defined as follows:

```
RuleID:    ON;
TriggeringCondition:
  (DMOVE:specifyCommand,
   TYPE:SwitchOn);
DeclareExpectations: {
  Location,
  DeviceType }
ActionsExpectations: {
  [DeviceType] =>
      {NLG(DeviceType);} }
PostActions: {
   ExecuteAction(@is-ON); }
```

The DTAC obtained for *switch on* triggers the dialogue rule **ON**. However, since two declared expectations are still missing (**Location** and **DeviceType**), the dialogue manager will activate the ActionExpectations and prompt the user for the kind of device she wants to switch on, by means of a call to the natural language generation module NLG(DeviceType). Once all expectations have been fulfilled, the PostActions can be executed over the desired device(s).

4 Integrating OWL in MIMUS

Initially, OWL Ontologies were integrated in MIMUS in order to improve its knowledge management module. This functionality implied the implementation of a new OAA wrapper capable of querying OWL ontologies, see (Pérez et al., 2006b) for details.

4.1 From Ontologies to Grammars: OWL2Gra

OWL ontologies play a central role in MIMUS. This role is limited, though, to the input side of the system. The domain–dependent part of multimodal and multilingual production rules for context–free grammars is semi–automatically generated from an OWL ontology.

This approach has achieved several goals: it leverages the manual work of the linguist, and ensures coherence and completeness between the Domain Knowledge (Knowledge Manager Module) and the Linguistic Knowledge (Natural Language Understanding Module) in the application. A detailed explanation of the algorithm and the results obtained can be found in (Pérez et al., 2006a)

4.2 From OWL to the House Layout

MIMUS home layout does not consist of a pre–defined static structure only usable for demonstration purposes. Instead, it is dynamically loaded at execution time from the OWL ontology where all the domain knowledge is stored, assuring the coherence of the layout with the rest of the system.

This is achieved by means of an OWL–RDQL wrapper. It is through this agent that the Home Setup enquires for the location of the walls, the label of the rooms, the location and type of devices per room and so forth, building the 3D graphical image from these data.

5 Multimodal Fusion Strategies

MIMUS approach to multimodal fusion involves combining inputs coming from different multimodal channels at dialogue level (Pérez et al., 2005). The idea is to check the multimodal input pool before launching the actions expectations while waiting for an "inter–modality" time. This strategy assumes that each individual input can be considered as an independent dialogue move. In this approach, the multimodal input pool receives and stores all inputs including information such as time and modality. The Dialogue Manager checks the input pool regularly to retrieve the corresponding input. If more than one input is received during a certain time frame, they are considered simultaneous or pseudo–simultaneous. In this case, further analysis is needed in order to determine whether those independent multimodal inputs are truly related or not. Another, improved strategy has been proposed at (Manchón et al., 2006), which combines the advantages of this one, and those proposed for unification–based grammars (Johnston et al., 1997; Johnston, 1998).

6 Multimodal Presentation in MIMUS

MIMUS offers graphical and voice output to the users through an elaborate architecture composed of a TTS Manager, a HomeSetup and GUI agents. The multimodal presentation architecture in MIMUS consists of three sequential modules. The current version is a simple implementation that may be extended to allow for more complex theoretical issues hereby proposed. The main three modules are:

- Content Planner (CP): This module decides on the information to be provided to the user. As pointed out by (Wahlster et al., 1993), the CP cannot determine the content independently from the presentation planner (PP). In MIMUS, the CP generates a set of possibilities, from which the PP will select one, depending on their feasibility.
- Presentation Planner (PP): The PP receives the set of possible content representations and selects the "best" one.
- Realization Module (RM): This module takes the presentation generated and selected by the CP–PP, divides the final DTAC structure and sends each substructure to the appropriate agent for rendering.

7 The MIMUS Talking Head

MIMUS virtual character is known as *Ambrosio*. Endowing the character with a name results in personalization, personification, and voice activation. Ambrosio will remain inactive until called for duty (voice activation); each user may name their personal assistant as they wish (Personalization); and they will address the system at personal level, reinforcing the sense of human–like communication (Personification). The virtual head has been implemented in 3D to allow for more natural and realistic gestures and movements. The graphical engine used is OGRE (OGRE, 2006), a powerful, free and easy to use tool. The current talking head is integrated with Loquendo, a high quality commercial synthesizer that launches the information about the phonemes as asynchronous events, which allows for lip synchronization. The dialogue manager controls the talking head, and sends the appropriate commands depending of the dialogue needs. Throughout the dialogue, the dialogue manager may see it fit to reinforce the communication channel with gestures and expressions, which may or may not imply synthesized utterances. For instance, the head may just nod to acknowledge a command, without uttering words.

8 Conclusions and Future Work

In this paper, an overall description of the MIMUS system has been provided.

MIMUS is a fully multimodal and multilingual dialogue system within the Information State Update approach. A number of theoretical and practical issues have been addressed successfully, resulting in a user–friendly, collaborative and humanized system.

We concluded from the experiments that a human–like talking head would have a significant positive impact on the subjects' perception and willingness to use the system.

Although no formal evaluation of the system has taken place, MIMUS has already been presented successfully in different forums, and as expected, "Ambrosio" has always made quite an impression, making the system more appealing to use and approachable.

References

Adam Cheyer and David Martin. 2001. The open agent architecture. *Journal of Autonomous Agents and Multi–Agent Systems*, 4(12):143–148.

Michael Johnston, Philip R. Cohen, David McGee, Sharon L. Oviatt, James A. Pitman and Ira A. Smith. 1997. Unification–based Multimodal Integration *ACL* 281–288.

Michael Johnston. 1998. Unification–based Multimodal Parsing *Coling–ACL* 624–630.

Staffan Larsson and David Traum. 2000. Information State and dialogue management in the TRINDI Dialogue Move Engine Toolkit. *Natural Language Engineering*, 6(34): 323-340.

Pilar Manchón, Guillermo Pérez and Gabriel Amores. 2006. Multimodal Fusion: A New Hybrid Strategy for Dialogue Systems. *Proceedings of International Congress of Multimodal Interfaces (ICMI06)*, 357–363. ACM, New York, USA.

Pilar Manchón, Carmen Del Solar, Gabriel Amores and Guillermo Pérez. 2007. Multimodal Event Analysis in the MIMUS Corpus. *Multimodal Corpora: Special Issue of the International Journal JLRE*, submitted.

OGRE. 2006. Open Source Graphics Engine. *www.ogre3d.org*

Guillermo Pérez, Gabriel Amores and Pilar Manchón. 2005. Two Strategies for multimodal fusion. E.V. Zudilova–Sainstra and T. Adriaansen (eds.) *Proceedings of Multimodal Interaction for the Visualization and Exploration of Scientific Data*, 26–32. Trento, Italy.

Guillermo Pérez, Gabriel Amores, Pilar Manchón and David González Maline. 2006. Generating Multilingual Grammars from OWL Ontologies. *Research in Computing Science*, 18:3–14.

Guillermo Pérez, Gabriel Amores, Pilar Manchón, Fernando Gómez and Jesús González. 2006. Integrating OWL Ontologies with a Dialogue Manager. *Procesamiento del Lenguaje Natural* 37:153–160.

Guillermo Pérez, Gabriel Amores and Pilar Manchón. 2006. A Multimodal Architecture For Home Control By Disabled Users. *Proceedings of the IEEE/ACL 2006 Workshop on Spoken Language Technology*, 134–137. IEEE, New York, USA.

Talk Project. Talk and Look: Linguistic Tools for Ambient Linguistic Knowledge. 2004. 6th Framework Programme. *www.talk-project.org*

Wolfgang Wahlster, Elisabeth André, Wolfgang Finkler, Hans–Jürgen Profitlich and Thomas Rist. 1993. Plan–Based integration of natural language and graphics generation. *Artificial intelligence*, 63:287–247.