E-Partner: A Photo-Realistic Conversation Agent

Bo Zhang[1*] Changbo Hu[2*] Qingsheng Cai[1] Baining Guo[3] Harry Shum[3]

[1]Dept. of Computer Sci. & Tech.
Univ. of Sci. & Tech. of China
Hefei 230027, P.R.China

[2]Institute of Automation
Chinese Academy of Science
Beijing 100080, P.R.China

[3]Microsoft Research China
Microsoft Corporation
Beijing 100080, P.R.China

Abstract. An E-Partner is a photo-realistic conversation agent, which has a talking head that not only look photo-realistic but also can have a conversation with the user about a given topic. The conversation is multimedia-enriched in that the E-Partner presents relevant multimedia materials throughout the conversation. To address the challenges presented by the complex conversation domain and task, and to achieve adaptive behaviors, we have derived a novel dialogue manager design consisting of five parts: a domain model, a dialogue model, a discourse model, a task model, and a user model. We also extended existing facial animation techniques to create photo-realistic talking heads that facilitate conversational interactions. Some practical issues like how to handle the uncertainty from speech level are also discussed.

1 Introduction

Computer-animated characters, particularly talking heads, are becoming increasingly important in a variety of applications including video games and web-based customer services. A talking head attracts the user's attention and makes user interaction more engaging and entertaining. For a layperson, seeing a talking head makes interaction with a computer more comfortable. Subjective tests show that an E-commerce web site with a talking head gets higher ratings than the same web site without a talking head [7].

We describe a multimedia system for creating *photo-realistic conversation agents*, i.e. talking heads that not only look photo-realistic but also can have a conversation with the user. The conversation is *multimedia-enriched* in that the conversation agent presents relevant multimedia materials (audios, images, videos, etc.) throughout the conversation. We have built such a system, called *E-Partner*, which combines a number of component technologies including speech recognition, natural language processing, dialogue management, and facial animation. Integration of these technologies into an end-to-end system is by no means an easy task. In this paper, we describe the design of our system, with a focus on novel aspects of dialogue management and facial animation. We will also discuss practical issues, such as how to handle the uncertainty from speech level by using low-level information from an off-the-shelf speech API.

---

[*] This work was performed while the authors were visiting Microsoft Research China.

The dialogue manager in our system is designed so that the system can have an intelligent conversation with the user about a given topic. Specifically, our conversation agent is

Informative: the user can learn useful information through a multimedia-enriched conversation, and

Adaptive: the conversation agent can respond to the user's requests (e.g., skip part of the speech when being asked to) and adapt the conversation as the agent learns more about the user.

Later we will demonstrate these features through an example, in which the agent talks with the user about a tourist site.

Because it is informative, our conversation agent differs from chatter bots [10] and similar systems that use clever tricks to fake a conversation. More important, our system is different from simple-service dialogue systems [11], which allow the user to retrieve information or conduct transactions, and plan assistant dialogue systems [5], which can help the user execute a task. The conversation domain of our system is much more complex than that of the simple-service and plan assistant systems, and our conversation agent exhibits adaptive behaviors. The task of our conversation agent is also different from that of a plan assistant. To address the challenges presented by the complex conversation domain and task, and to achieve adaptive behaviors, we have derived a novel dialogue manager design consisting of five parts: a domain model, a dialogue model, a discourse model, a task model, and a user model.

Facial animation in our system is also crafted to facilitate conversational interactions. Existing techniques allow us to create a photo-realistic talking head for each individual sentence spoken by the conversation agent [2, 4]. In this work, we extend [2] so that the facial animations for individual sentences can be stitched together to provide a continuous presence of the talking head, which pauses between sentences and appears to listen (instead of just freeze) when the user speaks. The talking head can smoothly switch back and forth between the listening and talking states.

The remainder of this paper is organized as follows. In next section, we present the architecture of the E-Partner system. The dialogue manager is described in section 3. Section 4 is about the secret of the talking head. We conclude in section 5.

2 System Architecture

Fig.1 is a snapshot of a user interacting with Maggie the tour guide – our first application of the E-Partner system. The user converses with Maggie through a microphone. Maggie resides in the lower right corner of the screen. She talks with the user about different places of the Forbidden City, and presents multimedia materials (images, videos, etc.) in the *content window* in the upper left corner of the screen.

Fig. 1. A snapshot

1. M> Hi, I'm Maggie, your tour guide. Welcome to …
2. U> Ok
3. M> [ShowPic 1] Below is an introduction to the Forbidden City.
4. M> Also known as the Imperial Palace Museum or Gugong, …
5. U> I don't want to listen to this introduction part
6. M> Ok. Do you want to go to the Gate of Meridian?
7. U> Yes, go to that place
8. M> [ShowPic 2] Welcome to the Gate of Meridian.
9. M> The Meridian Gate is the southern entrance of the …
10. U> Can you tell me the height of the meridian gate
11. M> The height of the Gate of Meridian is 35.6 meters. Do you want to know why it got its name Meridian Gate?
12. U> Yes
13. M> The Meridian Gate was so named because …
14. M> We will leave the Gate of Meridian now. Do you want to go to the Gate of Supreme Harmony?
15. U> No, I want to go to that hall
16. M> Where do you want to go?
17. U> The hall of middle harmony
18. M> [ShowPic 3] Welcome to the Hall of Middle Harmony.
19. M> The Hall of Middle Harmony originally built in 1420 and …
20. U> I want to leave this place
21. M> Ok. …

Fig. 2. A dialogue example (M: Maggie, U: user)

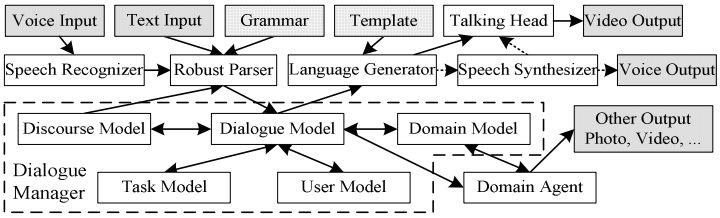


Fig. 3. Overall architecture of our system.

We illustrate some features of E-Partner through a dialogue example in Fig.2. If Maggie's sentence is too long for the user, he can barge in at any time (lines 1-2). The pictures are shown in the content window (line 3). Maggie has a tour plan for visiting different places within the Forbidden City. However, she can adapt to the user and skip sites when asked to (lines 5-6). She can answer the user's questions (lines 10-11). More important, she learns more about the user through these questions and takes high-level initiatives at appropriate situations (line 11). She also has the ability of limited story-telling (line 13). She takes the low-level initiative for form-filling (line 16), as well.

The main components of our E-Partner system (Fig.3) are:

- **Speech Recognizer:** get voice input and send the result to the parser. We use Microsoft speech recognition engine. In integration with other components, we gather rich low-level information (sound or phrase start event, recognition hypothesis, etc.) to aid the dialogue manager to handle the uncertainty. One example is to detect user barge-in before the recognition result. We also use the parser grammar to guide the selection of the candidate words.

- **Robust Parser:** get text input or recognition result; send the semantic interpretation to the dialogue manager. We use the robust parser [9] from the speech group of Microsoft Research. With LEAP grammar, it can handle many ill-formed sentences and is also capable of partial parsing.
- **Dialogue Manager:** select appropriate actions; send them to language generator or domain agent.
- **Domain Agent:** execute the non-verbal action, i.e., show pictures. It interacts with domain model intensively.
- **Language Generator:** execute the verbal action; send the result text to the talking head or speech synthesizer. A template-based approach is used to generate responses.
- **Speech Synthesizer:** generate voice from text; send it to the talking head. Now we use pre-recorded speech (Microsoft TTS engine if no talking head).
- **Talking Head:** synthesize video sequences from speech.

3 Dialogue Manager

Compared with simple-service systems and plan assistant systems, E-Partner differs in both the form of the information source and the way of conveying the information to the user. In a simple-service system, the information source is usually in a similar form and stored in some well-structured databases. In a plan assistant system, it can be stored in some knowledge bases. While in our system, although all the information remains related to a particular topic, it may be in many different forms, and is too complex to fit in databases or knowledge bases. The task of the system is also different. E-Partner wants to actively provide user with different useful information about a particular topic, while in a simple-service system, a correct answer for the user's current request is usually adequate, and a plan assistant system must help the user accomplish the task known to the system.

Our major problem here is how to represent and present the knowledge. E-Partner has many kinds of knowledge sources, which will play different roles in the dialogue system. The mixture of all these knowledge sources has a number of drawbacks [6]. In our system, we divide the knowledge sources into five parts: a domain model, a dialogue model, a discourse model, a task model, and a user model (Fig.3).

Besides the form of the domain model, the key difference from other systems is the complexity of the task model and the user model. The task model gives the E-Partner the ability to take the initiative in a high level, conveying the information to the user actively. The user model contains user preferences, which are built from the interaction between E-Partner and the user. User preferences can improve the quality of conversation, as well as user satisfaction.

Our dialogue manager receives the semantic representation from the parser, and then decides what to do by rule matching, which also takes the user model into consideration. If it does not get enough information, it prompts the user to give more information. If the user does not have a particular request, it takes the initiative

according to the task scripts. By combining rule matching with form filling and task execution, we have a two-layer mixed-initiative dialogue system.

Dialogue Model

Our dialogue model is a combination of the dialogue grammar and frame-based approach [3]. We use rules to construct our dialogue model into hierarchical sub-dialogues. The dialogue manager chooses the appropriate sub-dialogue according to the discourse context and parser output. We can handle many kinds of conversation by providing carefully designed sub-dialogues.

Each rule specifies a list of actions to be executed sequentially when in a certain dialogue state (CONTEXT condition) and for a particular sentence input by the user (USER condition) (Fig.4). A USER condition is a form with many slots. If the dialogue manager finds any unfilled slot, it repeatedly asks for the missing slot, using a natural prompt specified by the designer, until all the missing slots are filled or the user gives up the current form. The parser focus is also set to reflect the form-filling status. Because the user can answer in any order they prefer, this can result in a low-level mixed-initiative behavior.

Discourse Model

The discourse model represents the current state of the dialogue. We use two kinds of data structures to represent the dialogue history: data objects constructed recently, in the form of a list of filled or unfilled slots, and a list of context variables, which can be read and written in a rule, indicating some special states.

Domain Model

The domain model holds knowledge of the world. It includes many kinds of information. The main structure is a tree, which provides a structured representation of many related concepts, i.e. a place tree contains different levels of places. Many other facts are directly or indirectly related to the different places in this place tree. LEAP grammar and the language template are also the implicit representation of part of the domain knowledge.

Task Model

The main task of the E-Partner is to actively provide the user with useful information related to a given topic. It is not a simple one-shot task. Usually, this task consists of many sub-tasks or primitive actions. The E-Partner should have the ability to fulfill these sub-tasks during the interactions with the user. Since the execution of a sub-task may be interrupted by the user when the user initiate a sub-dialogue, E-Partner should take the initiative again to continue its task after this sub-dialogue. So we have a high level mixed-initiative behavior when combining the task model with the original rule-based system.

A script language (Fig.4) is devoted to define the task structure using a hierarchical task tree. The leaf node represents the terminal task consisting of action sequence while the internal node represents the non-terminal task. Since there are many related concepts in the domain model, the tasks can be associated with different concepts.

When switching between different concepts, we also switch the current task to the associated task.

To prevent the task from disturbing the user, E-Partner executes the task only when it is not engaged in any sub-dialogue. The user can also ask E-Partner to skip the current task or parent task.

User Model

Our user model only includes the user preferences:

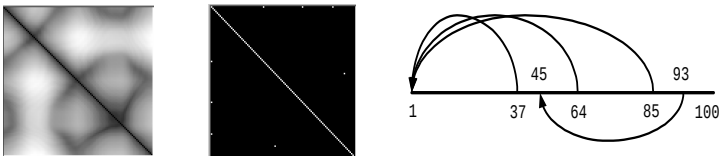
- If the user is in a hurry, skip some less important information. We assume that a hurried user likes to interrupt E-Partner.
- If the user does not like short stories, reduce the frequency of the story telling. We assume a user does not like stories if he often answers "no" when being asked whether he wants to hear a story, or interrupts a story.
- If the user likes to ask many questions, increase the frequency of question requests, which is a trick of saying something to lead the user to ask some particular questions that can be answered. This is very useful in increasing user satisfaction when many of their previous questions cannot be answered satisfactorily.

As long as the dialogue proceeds, the user's preferences will be changed dynamically and our dialogue manager can adapt to this kind of change by storing the parameters of the user preferences in the user model. Additional useful user preferences that may fit in the user model are currently being considered.

4 Facial Animation

For facial animation we use video rewrite [2], which is one of the most effective techniques for creating photo-realistic talking heads. However, video rewrite by itself cannot support conversational interactions as it is only a technique for creating a talking head for each individual sentence spoken by the conversation agent. We have extended [2] so that the facial animations of individual sentences can be stitched together to provide a continuous presence of the talking head, which pauses between sentences and appears to listen (instead of just freeze) when the user speaks. The talking head can smoothly switch back and forth between the listening and talking states.

The facial animation synthesized by video rewrite is a composite of two parts. The first part is a video sequence of the jaw and mouth, which is lip-synched according to the spoken sentence. The second part is a "background video" of the face. The mouth video sequence is generated from an annotated viseme database extracted from a short training video using a hidden Markov model (HMM). To support conversational interactions, we need a "background" video of the talking head that is alive and having a neutral expression. This ``background" video is taken from the training video. As mentioned, the training video is short. If the same background video is used repeatedly, the talking head appears artificial. Using video texture [8], we generate a "background" video that continues infinitely without apparent repetition. Similarly,



Distance matrix Transmission loops Possible transition graph

Fig. 5. Video texture of a background video with 100 frames

we use video texture to provide the talking head for the conversation agent when it is in the listening state.

Video texture generates infinitely long video from a short video sequence by jumping back to an earlier frame whenever there is a smooth transition from the current frame and an earlier frame. The smoothness of the transition is measured by the L2 distance between the frames. Fig.5 shows an analysis of a short video sequence. For this sequence, after dead-end avoidance and transitions pruning [8], the possible transmissions are (1, 37), (1, 64), (1, 85), (93, 45). In each transition point, we set a transition probability to decide the whether to jump back to an earlier frame.

A challenging issue in using video rewrite for conversational interactions is facial pose tracking. Accurate facial pose tracking is critical for the smooth transition between the talking and listening states, as well as for allowing natural head motion of the conversation agent when it is in the talking state. To allow a variety of head motions, the system warps each face image into a standard reference pose. The system tries to find the affine transform that minimizes the mean-squared error between face images and the template images [1]. The quality of facial animation largely depends on continuity of the facial pose. Even with a little error, the continuous mouth motion and the transition between talking and listening states become jerky.

We apply two strategies to improve the facial pose estimation. One is to apply second-order prediction to determine the initial pose. The other is to low-pass filter the pose parameters if abrupt motion exists. But in some cases there really exists large motions; we must eliminate the false alarm. We first interpolate the parameters of the pre frame and the post frame to obtain several different new parameters, and then compute the residue errors by each group of parameters. If the new error is greater than the original one, we believe that the abrupt motion is true. Otherwise, the parameters take the filter value. Linear interpolation directly on the affine parameters is not reasonable, because the parameters of the affine matrix do not correspond to the physical motion. So we decompose the affine matrix as the following:

$$\begin{bmatrix} a_1 & a_1 & a_2 \\ a_3 & a_4 & a_5 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} sx & k & 0 \\ k & sy & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Note that the decomposition is not unique. If we maintain the order and assume some condition, the parameters can represent some physical meaning. For example, when k is small enough and (sx-sy) is very small, the θ in fact is the rotation. We use k and θ to determine if abrupt motion occurs, and to predict and filter on the physical

parameter. Then the new affine matrix can be computed. We conduct experiments to show that the parameters are more continuous than the original method, and that the jerkiness is eliminated.

5 Conclusion

In this paper, we present a photo-realistic conversation agent called E-Partner. The first application of the E-Partner project – Maggie the tour guide of the Forbidden City – has been demonstrated at Microsoft since last September. We have received positive feedback from many visitors. They think such a cyber companion will play a very important role in many similar applications. For example, in E-commerce applications to sell cars, an E-Partner can be a virtual expert on cars who can talk about cars with users. Other potential applications include some information-providing applications, interactive games, a lovely partner in a living room, etc.

References

1. Black, M.J., Yacoob, Y: Tracking and recognizing rigid and non-rigid facial motion using local parametric models of image motion. In *Proceedings of IEEE Intl. Conf. Computer Vision*, Cambridge, MA, 374-381, 1995.
2. Breglar, C., Covell, M., Slaney, M.: Video rewrite: Driving visual speech with audio. In *Proceedings of SIGGRAPH'97*, 353-360, July 1997.
3. Chu-Carroll, J.: Form-based reasoning for mixed-initiative dialogue management in information-query systems. In *Proceedings of Eurospeech'99*, 1519-1522, 1999.
4. Cossatto, E., Graf, H. P.: Photo-realistic talking-heads from image samples. *IEEE Trans. on Multimedia*, 2(3), September 2000.
5. Ferguson, G., Allen, J.: TRIPS: An Intelligent Integrated Problem-Solving Assistant. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*(AAAI-98), Madison, WI, 567-573, July 1998.
6. Flycht-Eriksson, A.: A survey of knowledge sources in dialogue systems. In *Proceedings of IJCAI-99 Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, Stockholm, 1999.
7. Pandzic, I., Ostermann, J., Millen, D.: User evaluation: synthetic talking faces for interactive services. *The Visual Computer*, 15:330-340, 1999.
8. Schodl, A., Szeliski, R.: Video textures. In *Proceedings of SIGGRAPH'99*, 1999.
9. Wang, Y.: A robust parser for spoken language understanding. In *Proceedings of Eurospeech'99*, 1999.
10. Weizenbaum, J.: ELIZA - a computer program for the study of natural language communication between man and machine. *C. ACM*, 9:36-43, 1966.
11. Zue, V. et al.: JUPITER: A telephone-based conversational interface for weather information. *IEEE Transactions on Speech and Audio Processing*, 8(1), January 2000.