DIALOGUING WITH AN ONLINE ASSISTANT IN A FINANCIAL DOMAIN: THE VIP-ADVISOR APPROACH

Josefa Z. Hernandez[1], Ana Garcia-Serrano[1] and Javier Calle[2]
[1]Technical University of Madrid, Spain; [2]Carlos III University, Madrid, Spain

Abstract: Virtual assistants are a promising business for the near future in the web era. This implies that the supporting applications have to be endowed with advanced capabilities to service offerings and to communicate with the users in a more direct and natural way. This paper presents the agent-based architecture of the virtual assistant and focuses on the dialogue module. The content exchange between the agents is based on *communicative acts* to cope with the complexity of unrestricted language used by human users communicating with online assistants. The assistant is capable to interact with users and to provide the right output through the exploitation of different information sources. The approach was applied and tested on the insurance field in the frame of the European research project VIP-Advisor[1].

Key words: Applications of Artificial Intelligence, Integration of AI with other Technologies, Speech and Natural Language Interfaces

1. INTRODUCTION

The work described in this paper is related with the development and deployment of a virtual assistant that advices the user for risk management in the insurance and finance field. With the aim of perform a satisfactory interaction with the user, the VIP-ADVISOR assistant main features are:

- *competent* by reasoning with a rule-based for financial knowledge and an additional case based knowledge source for information retrieval support
- *user-adapted,* given that automatically adopts to the user's needs

[1] VIP-ADVISOR project IST-2001-32440 (www.vip-advisor.fi.upm.es)

- *reactive and easy to use* through the use of natural language speech (translated into text with technologies from Linguatec GmbH)

The virtual assistant is capable of interacting with the user in natural language, adapt its recommendations to the requirements and characteristics of the customers and provide explanations along the interaction.

One of the main aspects of the interaction in the web is the capability of the web site of generating some kind of trust feeling in the user, just like in a person-to-person interaction. Things like understanding the aspects that influence the degree of the different risks identified, assisting the user in the final decision to get a product are not easy things to achieve in an online advisor site. Dialogue models and related techniques can play a crucial role in providing this kind of enhancement.

The paper is structured as follows: next section focuses on the design and deployment of the dialogue management performed by the Interaction Agent of the assistant; section 3 includes a brief description of the VIP-ADVISOR project. Section 4 is devoted to explain the semantic and pragmatic agents communication language that cover specific peculiarities of the dialogue and the domain. In section 5 is included some related work and, finally, in section 6 some conclusions are presented.

2. THE VIP-ADVISOR PROJECT

The virtual assistant developed in the VIP-Advisor project was conceived to support users on risk management. An existing 'conventional' online application for risk management was embedded in the virtual assistant as one of the information sources.

The architecture of the online assistant using an agent-based approach includes:

- An *Interface agent* responsible for the multimedia interaction with the users, supported by a graphical user interface, including a 3D avatar, speech recognition and synthesis, eventually with online translation, and natural language processing.
- *Intelligent agents,* identifying the information sources exploitable by the assistant, which are responsible for generating the information required by the user.
- An *Interaction agent* or interaction manager responsible for the coherent evolution of the user-system interaction, i.e. what to say and how to say it according to the context of the conversation and the recent dialogue.

The communication between the different agents performs briefly as follows: User demand is received by the Interface Agent through mouse clicks or with a spoken or written sentence. If this sentence is in German it is translated to English and delivered to the natural language processor.

The NL interpreter analyses the English sentences to extract both the content and the intention of the input and deliver it to the Interaction agent. In the case the input is given through mouse clicks the content and intention of these clicks is also extracted and delivered to the Interaction agent.

According to the input, the Interaction agent decides how to proceed with the conversation, identifying what kind of answer needs to be provided and which is the Intelligent Agent that has to generate it. In the VIP-Advisor system two Intelligent Agents are used: the existing risk management application and a case-based reasoner to deal with unexpected situations/queries. Once the response from the corresponding Intelligent Agent is received it is recorded –for future interactions- and delivered to the Interface Agent together with indications on how to provide it to the user (e.g. the kind of language to be used, the gestures of the 3D avatar). The Interface Agent distributes the output information among the different modules involved –the NL generator, the speech synthesizer, the 3D avatar and the graphical interface- and the response is displayed to the user.

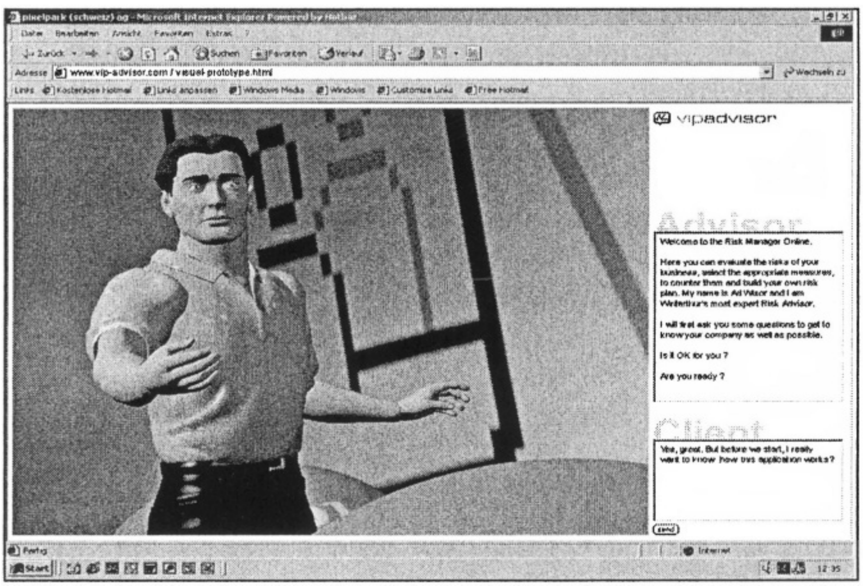

Figure 1. Screenshot of the VIP-Advisor prototype

The two prototype versions of the VIP-Advisor intelligent assistant have been developed with an agent-based architecture, where all agents are web services and the inter-agent communication is based upon SOAP protocol and the exchange of XML files. Single platforms for implementation and programming languages used are varied. In particular, the Interaction agent and the natural language modules of the Interface Agent were developed in Java and Ciao Prolog (www.clip.dia.fi.upm.es). Figure 1 shows a snapshot of the VIP-Advisor prototype.

From the previous description it can be guessed that a powerful language for the communication between all the agents has to be involved in this process. In the next section a detailed description of the communicative acts role in the semantic and pragmatic agents communication language is included.

3. SEMANTIC AND PRAGMATIC AGENTS COMMUNICATION LANGUAGE

The system has to be able to manage the meaning of the user actions and the intentions that he/she has when performing those actions (clicking, speaking, writing, ...). These two items are encoded using communicative acts, an adapted and extended version of the Searle's theory of speech acts[1,2] to the e-services environment.

The communication between the agents is based on a set of communicative acts consisting in a set of fourteen of these structures distributed into six categories:

1. *Representative,* the emitter shows a link with the reality, sharing with his interlocutor;
2. *Directive,* the emitter should direct his interlocutors next actions;
3. *Authorizative,* the emitter wants his interlocutor to perform some action;
4. *Courtesy,* social conventions that both interlocutors should observe just for protocol, tuning, or politeness;
5. *Non-verbal,* Required for non-verbal social conventions such us connection; and
6. *Null,* when there is no effective communication but imply an intervention (such content empty discourses and pauses).

Several parameters characterise the occurrence of each dialogue act in a way that can be shown as 'labelled' communicative acts to be handled by the agents. A detailed description is provided in Garcia-Serrano and Calle[3].

The dialogue-based interaction approach applied is supported by a discourse model with a twofold dimension: informational and intentional. The informational approach establishes that the coherence of the discourse follows from semantic relationships between the information conveyed by successive utterances (inference-based approach as major computational tool). The intentional approach claims that the coherence of discourse derives from the communicative intention of both speakers and that mutual understanding depends on the capability to recognise those intentions (following speech act theories). The dialogue-based interaction applied also supports a genuine integration of semantics and pragmatics, since a good analysis of dialogue requires semantic representation (representation of the content of what the participants are saying) and pragmatic information (what kinds of communicative acts they are performing - asking/answering a question, making a proposal, ...-, what information is available to each speaker, what is the purpose behind their various utterances, etc.)

Table 1. Example of representative acts

Parameters:	[t]: type	Data: objective information Confirmation: bivaluated Suggest: subjective opinions
	[m]: matter	Approve: confirms the subject Deny: negates the subject Identity: identifies the subject Feature: marks an attribute Has: ownership Interest: aim
	[s]: subject	Product, User, System, Offer...
	[c]: content	[value | var=value]
Inform	I'm AI:	inform(data,identity,user,AI)

According to the previous considerations, the Interaction Manager has three main responsibilities: (1) managing the evolution of the conversation in a coherent way which means keeping track of the on-going dialogue and deciding how to proceed next, (2) asking the corresponding Intelligent Agent to perform the necessary task to generate the information required by the user, and (3) delivering to the Interface Agent the required information together with indications on how to provide it.

Three modules support the Interaction Agent to undertake these responsibilities:

- *Dialogue manager* for controlling the evolution of the conversation. It records the state of the dialogue according to the active dialogue strategy, through a dialogue state component, and the intention or communicative goal that needs to be satisfied next, through a thread joint stack component.
- *Session model* for handling the static information of the interaction
- *Discourse maker* for generating the next intervention of the system, i.e. the stream of communicative acts to be delivered to the Interface Agent. It contains a task model for determining what the effects of a discourse

should be and how the external modules (i.e. the risk manager and/or case-based reasoner results) should contribute to the dialogue.

The general performance of the interaction agent is summarized in the following. During user's turn (in a dialogue), the interaction agent will be waiting for his action. Then, a stream of communicative acts (or several of them) will be delivered to the agent. If it is only a null act informing of a pause, or an act derived from the G.U.I. (if the user clicked on a menu, for example), or some verbal intervention interpreted by the Natural Language Interpreter. Anyway, the information carried by these communicative acts will be interpreted all the same.

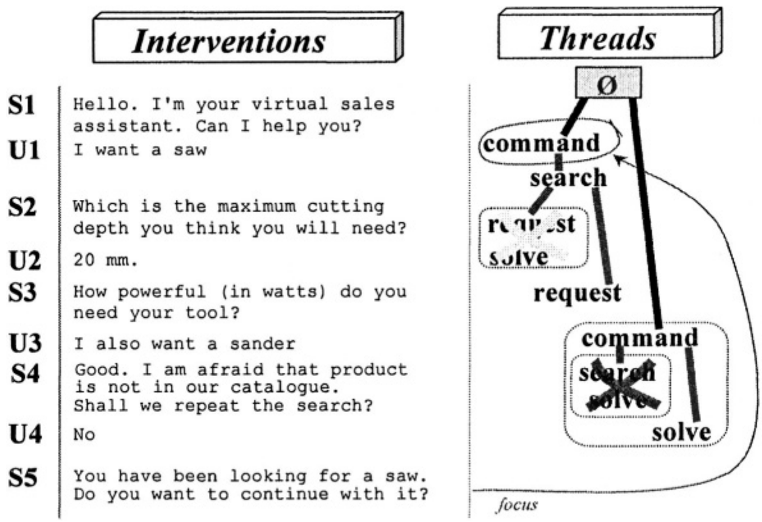

Figure 2. Threads management in an interaction example

Regarding the threads management, the intervention can follow the focused thread, any other previous thread, or originate a new one (initiative threads, or just 'initiatives'). Threads are arranged as a tree like structure, and leaf nodes should be closed with the inclusion of a 'solve' thread, which means that the thread of that branch has been satisfied (of course, the thread won't be erased, but 'faded'). Any interlocutor might at any moment 'jump' to a branch in the thread tree different than the focused one (a previous thread or a new one that is not a descendent of the focused thread, as example in figure 2).

In the example before intervention (S2) the intelligent agent has to provide a product identification. Since there were several of them, the discourse maker decides to create a new thread to obtain from the user more information (particularly, the desired cutting depth for that product) in order to refine the solution. So, entry point of the discourse maker changes (new focused thread), and this time, it has no need of external help for obtaining all needed contents, hence chooses some patterns for expressing its desired discourse.

Notice that the two intelligent agents are not the only ones that might pose an initiative. In fact, almost every component in the interaction could need to introduce initiatives. For example, the session model could need to reinforce the credibility of a context piece (whenever the use of the piece determines a certain threshold, and the credibility is under that minimum required). Then, this component should introduce an initiative for reinforcing the 'quality' of the information (that will probably lead to a subdialogue starting with the question 'this feature has this value, hasn't it?'). Even the very thread model could pose initiatives (when the 'quality' of a thread is under a predefined threshold).

The evaluation of the second prototype was performed using several scenarios, with different situations and related aspects to be addressed during Main results of the evaluation focusing on the dialogue management are:

- Only the 1 % of total interventions were buggy ones (cause user and system not to be attuned into the same plane of interpretation) and are always recovered through only the 1,52 % of total interventions.
- 9 % of total interventions were of no use (most of them due to doubts)
- The average number of steps performed through any intervention was 0,84 for the user and 1,51 for system's interventions. These results reveal quick progress of any dialogue.

Finally, some remarks about future improvements suggested by the evaluation are included. Although there were a few buggy interventions, intentional commitment should be attained for avoiding these situations. Even if the system directed dialogues were not very aggressive, the dialogue should be reinforced (avoiding user's 'acts of faith').

4. RELATED WORK

Much of the research in the area has been restricted to task oriented dialogue domains. However, even most advanced systems in interaction are far from achieving good pragma-linguistic levels. Different systems were implemented according a few of most relevant theories on pragmatics, with remarkable results.

Current approaches in dialogue modeling[4] have assumed goal related to introducing system participation in the dialogue. Thus user and system have 'autonomous' role in the interaction yet they need to be suitable attuned for success in attaining their common goals through joint action. In this regard three approaches for dialogue modelling are applied: dialogue grammars, plan-based models, and joint action theories applied to dialogue.

Late research on dialogue grammars for task-oriented dialogues try to combine conversational games models with powerful reasoning tools. An example provided by Pulman[5] proposes combining this technique with the reasoning power of the Bayesian networks either for the recognition/interpretation of the user moves, the planning of next moves (tactics) and the planning of next games (strategy).

In the plan-based models the system is endowed with the capability to interpret/understand user' action plans to achieve goals. One approach considers plans as long-term intentions the system aims to accomplish. Hence, the dialogue management will maintain a stack of system goals to attain, the so-called agenda proposed in Boye et al.[6]. Changing the plan for this agenda could pose different dialogue strategies. Another approach for plan-based models considers a plan always focused on a fixed task. Hence, interaction should be seen as a way for acquiring some information needed for the task and the state of the dialogue that determined by the information already acquired (or 'information state'). Satisfactory results have being obtained within this approach, being the more representative those from the IST projects Trindi and Siridus[7,8].

Finally, joint-action models are based upon the premise that a dialogue is feed by both talking entities, and that they have at least one common reference point, a commitment. Those joints have to be kept 'alive' by both entities with an acceptable level of certainty and efficiency, so they can refer to it. The previously mentioned IST projects Trindi and Siridus claim for the goodness of including such a "common ground" between user and system, and the Advice project applied it with satisfactory results.

Another example on the use of these models is the IBM MIND[9] (Multimodal Interpretation for Natural Dialog), with features as in the previous project NLSA system: multimodality, architectural, semantic structures. But the discourse level analysis is more complex, mainly based upon Grosz and Sidner[10] theories of intention and attention. It differentiate between two main discourse elements: the unit (intention partially developed within a turn) and the segment (full intention, developed through several turns, at least one). The segment will be composed of five attributes: Intention, Attention, Initiator, Addressee, and State. Also two types of discourse relations can be observed: structural relations (intention-sub intention structures), and transitional relations (transitions between conversation segments and between conversation units; they can be further sorted in two types, intention switch and attention switch respectively). These intentional structures (segments) will help to handle the dialogue structure, as observed in the figure, for attaining coherent dialogues.

Multimedia based systems are very well spread, and there exist several examples of successful use of agent-based technology as QuickSet[11]. This system stands for improving HCI by posing very efficient interfaces, using for example a pen on a tactile screen, but also quicker human interfaces to deal with verbal expression or pointing using a gesture recognizer.

Regarding the tasks a user might ask through interaction, they are always defined by the domain (often knowledge based). In anticipation of the forthcoming growth of the number of these tasks for most systems, an organization of them should be modeled along with the consequences their results could have for the interaction. Another open issue is the standardization of the activation of tasks (often performed by external agents), and the interpretation of the task results and the mechanisms to introduce them into the dialogue.

5. CONCLUSIONS

For attaining natural interaction with the user, specialized agents are needed to perform mayor tasks: the interface ones, the dialogue management and the intelligent decision making in order to generate/interpret the content of the communication. The proposed architecture is successful in this direction also for multi-user performance[12].

The development of an Interaction Agent able of performing any complete interaction within an specific domain is a very difficult task, so is necessary to separate, as in this work, the knowledge of how-to-interact and the rest of the (domain dependent) knowledge. The proposal for the interaction agent is based on the three components presented at section 3.

Next step should be done is the adaptability of the assistant to the different kind of users. A final open problem is to attain the independence of the domain and then the learning of the assistant from the available domain knowledge and available dialogue corpus.

ACKNOWLEDGEMENTS

We would like to thank the ISYS research group at the Technical University of Madrid for their support during the design, development and deployment of the interaction agent and NL components of the interface agent, and everyone involved in the project VIP-ADVISOR.

REFERENCES

1 N. Asher and A. Lascarides, 2003, *Logics of conversation* Cambridge University Press.

2 J. R. Searle, 1969, *Speech Acts: An Essay in the Philosophy of Language.* Cambridge University Press

3 A. García-Serrano, J. Calle, 2002, *A Cognitive Architecture for the Design of an Interaction Agent.* In: Cooperative Information Agents VI, LNAI 2446. Springer Verlag.

4 Cohen, P.R. *Dialogue Modeling.* Survey of the state of the art in Human Language Technology, 1997; chap. 6, pp. 234-240.

5 Pulman, S.G. *Conversational Games, Belief Revision, and Bayesian Networks.* In CLIN VII (7th Computational Linguistics in the Netherlands), 1997.

6 Boye, J,, Wirén, M., Rayner, M., Lewin, I., Carter, D., Becket, R. *Language Processing Strategies and Mixed-Initiative Dialogues.* In IJCAI-99 Workshop on Knowledge and Reasoning in Practical Dialogue Systems, 1999.

7 Cooper, R., Larsson, S., 1998. *Dialogue Moves and Information States.* In Proceedings of the third IWCS. Tilburg, 1999.

8 Amores, J.G., Quesada, J.F. *Dialogue Moves for Natural Command Languages.* Procesamiento del Lenguaje Natural, 27, pp. 81-88. Jaén (Spain), 2001.

9 Chai, J. Pan, S., Zhou, M. MIND: *A Context-based Multimodal Interpretation Framework in Conversational Systems.* In Natural, Intelligent and Effective Interaction in Multimodal Dialogue Systems, Eds. O. Bernsen, L. Dybkjaer and J. van Kuppevelt; Kluwer Academic Publishers, 2003.

10 Grosz B. and Sidner C. *Attention, intention and the structure of discourse,* Computational linguistics, V12, N3,1986

11 Cohen, P.R., Johnston, M., McGee, D.R., Oviatt, S.L., Pittman, J., Smith, I., Chen, L., and Clow, J. (1997) QuickSet: Multimodal interaction for distributed applications, in the Proceedings of the Fifth International Multimedia Conference (Multimedia '97), ACM Press: Seattle, WA, pp. 31-40

12 J. Correas, F. Bueno, D. Teruel, A. García-Serrano, An Agent-based Architecture for Multi-user Applications. In: Challenges in Open Agent Systems, AAMAS'02, Bologna, Italy, July 2002.