NASTIA: Negotiating Appointment Setting Interface

NASTIA: Negotiating Appointment Setting Interface

Layla El Asri[1,2], **Rémi Lemonnier**[1], **Romain Laroche**[1], **Olivier Pietquin**[3] and **Hatim Khouzaimi**[1,4]
[1]Orange Labs / Issy-les-Moulineaux, France
[2]UMI 2958 GeorgiaTech-CNRS / Metz, France
[3]University Lille 1, LIFL (UMR 8022 CNRS/Lille 1) - SequeL team / Lille, France
[4]Laboratoire d'Informatique d'Avignon-CERI / Avignon, France
layla.elasri@orange.com remi.lemonnier@ens-cachan.fr
romain.laroche@orange.com olivier.pietquin@univ-lille1.fr
hatim.khouzaimi@orange.com

Abstract

This paper describes a French Spoken Dialogue System (SDS) named NASTIA (Negotiating Appointment SeTting InterfAce). This system implements various strategies for appointment scheduling. These strategies were tested on 1734 dialogues with 385 users who interacted at most 5 times with the SDS and gave a rating on a scale of 1 to 10 for each dialogue. Previous appointment scheduling systems were evaluated with the same experimental protocol. The highest system task completion rate with these systems was 81% whereas NASTIA had an 88% average and its best performing strategy even reached 92%. This strategy also significantly outperformed previous systems in terms of overall user rating with an average of 8.28 against 7.40.

1 Introduction

NASTIA (Negotiating Appointment SeTting InterfAce) is a French Spoken Dialogue System (SDS). Its task is to schedule an appointment with a user who needs the intervention of an engineer on site. This SDS pursues the research on the appointment scheduling task, started in [Lacson, 2004] and continued during the CLASSiC EU FP7 project[1] [Laroche et al., 2011]. Several issues were identified during the evaluation of the CLASSiC systems, which served to design NASTIA. The system embeds reinforcement learning [Sutton and Barto, 1998] to decide between different actions in function of the current dialogue state and the belief over user profile. Reinforcement learning has been introduced into SDS more than a decade ago [Levin et al., 1997, Williams and Young, 2007, Pietquin and Dutoit, 2006] and was proven efficient to optimise the interaction strategy. A user evaluation campaign was carried. During this campaign, actions were picked randomly, following a uniform probability law. 1734 dialogues were collected with 385 users interacting with the system at most five times. After each dialogue, the user filled in an evaluation questionnaire about the dialogue.

This article discusses the results of the experiment and provides general recommendations about dialogue management in the case of appointment scheduling.

[1]Computational Learning in Adaptive Systems for Spoken Conversation, http://www.classic-project.org/

2 Related work

2.1 The SCHEDULER

The SCHEDULER [Lacson, 2004] is an SDS dedicated to the management of medical appointments. The first action performed by the SCHEDULER is to ask for the patient's name to check s/he is already in hospital records. Then the user may choose to create, cancel or check an appointment. For the creation task, they have to indicate the practitioner's name and a day. If some information are missing and the system cannot query the database, the user is asked for the missing items. Users may also specify a time preference and if they do not, the system simply proposes its first availability for the given day. The SCHEDULER was evaluated according to the three following criteria: task success, task ease and difficulties encountered during the dialogue. Nevertheless, the evaluation only concerned 15 calls which were not scenario-based so user constraints were not modelled in this experiment.

2.2 Systems designed during CLASSiC

The CLASSiC project gave birth to three systems enabling users to schedule an appointment with an engineer in the case of a dysfunction of their landline. Those three systems are referred to as Systems 2, 3 and 4 in Laroche et al. [2011].

System 2 was a state-of-the-art POMDP-based (Partially Observable Markov Decision Processes) SDS [Sondik, 1971, Williams and Young, 2007]. After each user input, the dialogue manager received the N-best list of semantic hypotheses, updated its dialogue state and chose its next dialogue act accordingly.

System 3 was also Reinforcement Learning-based (RL). In this case, RL was cast as a Module-Variable Decision Process (MVDP, [Laroche et al., 2009]). This system was designed to assess the influence of Text-To-Speech (TTS) prosody on users' behaviour. Each system utterance could be synthesised with one of these intonations: calm, neutral or dynamic.

System 4, on the other hand, was not RL-based. Its negotiation strategy was designed to leave more freedom to the user: the system either proposed a time slot or asked for different constraints such as week, day, half-day until it was able to make a proposition matching the constraints. The system chose between the two strategies on the basis of the number of remaining slots.

These three systems were tested and compared on scenario-based dialogues. The main results of this study will be discussed in Section 4.

These experiments enabled to point out the parts of the appointment scheduling process that needed to be improved. The conception of NASTIA resulted from the analysis provided in [Laroche et al., 2011].

3 NASTIA

3.1 Issues previously identified

Many of the problems detailed in the CLASSiC evaluation [Laroche et al., 2011] could be explained in terms of uncooperative behaviour of the system. For instance, it was noticed that users were sometimes confused by system feedbacks. Let us take the example of a user saying s/he would like to book an appointment on Friday afternoon. In this case, most of the time, the user meant the upcoming Friday afternoon.

Yet, in accordance with Grice's quality maxim [Grice, 1989], systems 3 and 4 would not make any assumption on the desired week. Thus, if the first appointment available was Friday afternoon of the following week, both systems 3 and 4 would have directly proposed this appointment without stipulating that the upcoming Friday was not available. Users tended to distrust speech recognition so that, in this case, they often chose to refuse the proposition and repeat their request.

For NASTIA, we chose to disambiguate these cases prompting the user with an implicit confirmation. To that user utterance, NASTIA would answer *Friday the 16th* to let the user know that it was supposing they were meaning the upcoming Friday. This new formulation respects Grice's quantity maxim as it provides to the user the necessary amount of information for them to understand the course of the dialogue.

Other modifications of the same nature were made resulting in many prompts being reformulated to move towards a better accordance with the Gricean cooperativity principles and make the system less of a black box to the user.

3.2 Dialogue modelling in NASTIA

Appointment scheduling is modelled as a slot-filling task with three parameters: day, week and half-day (morning or afternoon). NASTIA's dialogue manager is based on a finite state machine. Each node of the automaton is a dialogue phase. Dialogue phases in NASTIA are for example: Welcome, Confirm, Ask_open_Question, Ask_For_Day, Recovery (from speech recognition rejection or user time out), etc.

RL was integrated into this automaton thanks to the MVDP hybrid framework [Laroche et al., 2009]. MVDP enables to integrate one or several point(s) of choice in a dialogue phase, each point of choice having to choose between different actions according to its current internal context. NASTIA contains five points of choice.

The first one concerns the negotiation strategy. The User Initiative (UI) strategy consists of asking the user: "When would you like to book an appointment ?". System Initiative (SI) asks the user which day, which week and which half-day in three different dialogue turns. In addition to these classical strategies, we implemented an option where the system could directly propose a List of Availabilities (LA) to the user, waiting for them to interrupt the listing after an adequate appointment has been proposed. This last option was inspired by recent work on incremental dialogue management [Schlangen and Skantze, 2011]. The latter and barge-in in particular are at the heart of current research on SDS and more and more complex

Initiative strategy	User Initiative (UI); System Initiative (SI); List of Availabilities (LA)
ASR rejections; User inactivity	Play a help message; Tell the user their utterance was not understood
Confirmation strategy	Explicit confirmation; Implicit confirmation; No confirmation
System calendar information	Give information; Do not give information
Help message	Recall dialogue context; Give the possibility to cancel + Recall dialogue context + Recall available commands; Give the possibility to cancel + Recall dialogue context

Table 1: Actions of the appointment setting system.

models can be found in the literature [Selfridge et al., 2013]. Interruptions are common in human dialogue [Strombergsson et al., 2013], which makes it much more interactive than turn-taking human/machine conversation. Thus, including incrementality in a dialogue system is likely to make it more reactive and human-like.

The system chooses which strategy to follow at the beginning of each dialogue and after each appointment setting failure. This leaves to NASTIA the opportunity to adapt its way of realising the task in function of the course of the dialogue as it was proposed for instance by Chu-Carroll [2000] and Litman and Pan [2002].

A second point of choice concerns the contextual help. The user may express a help request at any moment of the dialogue. If so, the system may combine three components of help messages:

- (a) Tell the user: "You have required the help section" and let them the possibility to answer "no" in case the system misunderstood the user's request.
- (b) Recall the current context of the dialogue (*e.g.* "You were asked when you would like to make an appointment") and tell the user what they can say (*e.g.* "You can answer saying for instance *this Friday afternoon*, *this week in the morning* or *Monday the 19th*.")
- (c) Recall the available commands (Repeat and Help)

NASTIA chooses amongst three combinations: (b); (a) + (b) + (c) or (a) + (b).

The third point of choice is visited after a user has proposed a time slot. NASTIA may follow three confirmation strategies. Following the first strategy, the system does not ask for any confirmation. The implicit confirmation strategy simply consists of repeating what was understood. The explicit strategy asks: "I understood you were available on [understood date]. Is it correct ?".

The fourth point of choice has been implemented to compare two strategies for speech recognition rejections and user time outs recovery. The SDS may play the (b) help message or inform the user that they were not understood/heard so that the user repeats/says something.

The fifth point of choice concerns the information provided by the system after an appointment setting failure or after the user has expressed some constraints. For instance, if a user says s/he is available this week during the morning, the system may answer "This week, during the morning, only Tuesday and Friday are available". Then, the user directly chooses amongst the proposed days or gives new constraints. All NASTIA's points of choice as well as their action sets are gathered in Table 1.

4 Results

4.1 Comparison to CLASSiC Systems 2, 3 and 4

We compare the performance of NASTIA with the ones of CLASSiC's systems 2, 3 and 4 on the basis of System and User Task Completion (resp. STC and UTC), elapsed time (in seconds) and overall user evaluation on a scale of 1 to 10. These results are given in Table 2. NASTIA was evaluated on 1734 dialogues. There were 628 evaluated dialogues for System 2, 740 for System 3 and 709 for System 4. During the CLASSiC experiment, Systems 3 and 4 largely outdid System 2 concerning STC and UTC. System 3 was the one that did best in terms of overall evaluation and led to the shortest dialogues.

NASTIA performed similarly to System 3 in terms of overall evaluation although dialogues were in average 6 seconds longer. Moreover, NASTIA's UTC and STC rates are significantly

System	STC	UTC	Time (sec)	Rating	Number of calls
System 2	$79 \pm 3\%$	$68 \pm 4\%$	97 ± 5	5.21 ± 0.23	628
System 3	$81 \pm 3\%$	$83 \pm 4\%$	69 ± 3	7.40 ± 0.17	740
System 4	$83 \pm 3\%$	$85 \pm 3\%$	98 ± 5	6.54 ± 0.18	709
NASTIA	$88 \pm 2\%$	$92 \pm 1\%$	75 ± 3	7.75 ± 0.09	1734
NASTIA UI	$87 \pm 3\%$	$89 \pm 3\%$	84 ± 6	7.57 ± 0.16	587
NASTIA LA	$92 \pm 3\%$	$95 \pm 2\%$	61 ± 5	8.28 ± 0.14	562
NASTIA SI	$87 \pm 3\%$	$92 \pm 2\%$	79 ± 5	7.43 ± 0.17	585

Table 2: Performance comparison between NASTIA and CLASSiC's systems 2, 3 and 4. STC is System Task Completion and UTC, User Task Completion. Time is measured in seconds. We provide 95% confidence intervals for the mean of the binomial (STC and UTC) and the normal law (Time and Rating).

higher even though the system did not follow an optimised policy during the dialogues since its policy was uniformly random. This shows that all of the implemented strategies were of fair enough quality in the eyes of the users and of a good efficiency.

4.2 Strategies comparison

The system has to choose an initiative strategy at the beginning of every dialogue. In table 2, we have also computed 95% confidence intervals for the key performance indicators in function of the first decision made by the system. It shows that LA entailed significantly higher evaluations and shorter dialogues, no matter the policy followed by the system afterwards. Dialogues are shorter and the mean evaluation with LA is clearly higher than the mean rating of CLASSiC's System 3 (7.40). In future work, RL will be applied to learn the best strategy for each point of choice.

5 Conclusion

This paper discussed dialogue management for the appointment scheduling task. Future work will consist of selecting dialogue features for state space definition, learning a policy from the DINASTI corpus and then testing this policy with the same experimental protocol to compare its results with the ones presented in this paper.

References

Abdeslam Boularias, Hamid R. Chinaei, and Brahim Chaib-draa. Learning the reward model of dialogue pomdps from data. In *Proceedings of NIPS*, 2010.

Jennifer Chu-Carroll. Mimic: An adaptive mixed initiative spoken dialogue system for information queries. In *Proceedings of ANLP*, pages 97–104, 2000.

Layla El Asri, Romain Laroche, and Olivier Pietquin. Reward function learning for dialogue management. In *Proceedings of STAIRS*, 2012.

Paul Grice. *Studies in the Way of Words*, chapter Logic and Conversation. Harvard University Press, Cambridge MA, 1989.

Ronilda Lacson. The medical appointment scheduler. In *Proceedings of MEDINFO*, 2004.

Romain Laroche, Ghislain Putois, Philippe Bretier, and Bernadette Bouchon-Meunier. Hybridisation of expertise and reinforcement learning in dialogue systems. In *Proceedings of Interspeech*, 2009.

Romain Laroche, Ghislain Putois, Philippe Bretier, Martin Aranguren, Julia Velkovska, Helen Hastie, Simon Keizer, Kai Yu, Filip Jurcicek, Oliver Lemon, and Steve Young. Report D6.4 : Final evaluation of classic towninfo and appointment scheduling systems. Technical report, CLASSIC Project, 2011.

Esther Levin, Roberto Pieraccini, and Wieland Eckert. Learning dialogue strategies within the markov decision process framework. In *Proceedings of IEEE ASRU*, 1997.

Diane J. Litman and Shimei Pan. Designing and evaluating an adaptive spoken dialogue system. *User Modeling and User-Adapted Interaction*, 12:111–137, 2002.

Olivier Pietquin and Thierry Dutoit. A probabilistic framework for dialog simulation and optimal strategy learning. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(2):589–599, 2006.

David Schlangen and Gabriel Skantze. A general, abstract model of incremental dialogue processing. *Dialogue and Discourse*, 2:83–111, 2011.

Ethan Selfridge, Iker Arizmendi, Peter Heeman, and Jason D. Williams. Continuously predicting and processing barge-in during a live spoken dialogue task. In *Proceedings of SIGDIAL*, 2013.

Edward Jay Sondik. The optimal control of partially observable markov processes. Technical report, Stanford Electronics labs, 1971.

Sofia Strombergsson, Anna Hjalmarsson, Jens Edlund, and David House. Timing responses to questions in dialogue. In *Proceedings of Interspeech*, 2013.

Hiroaki Sugiyama, Toyomi Meguro, and Yasuhiro Minami. Preference-learning based Inverse Reinforcement Learning for Dialog Control. In *Proceedings of Interspeech*, 2012.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning. An introduction*, pages 56–57. MIT Press, 1998.

Marilyn A. Walker. An application of reinforcement learning to dialogue strategy selection in a spoken dialogue system for email. *Journal of Artificial Intelligence Research*, 12: 387–416, 2000.

Jason D. Williams and Steve Young. Partially observable markov decision processes for spoken dialog systems. *Computer Speech and Language*, 21:231–422, 2007.