DINASTI: Dialogues with a Negotiating Appointment Setting Interface

Conference Paper · May 2014

3 authors:

Layla El Asri
Maluuba

Romain Laroche
Microsoft Maluuba

Olivier Pietquin
Google DeepMind

DINASTI: Dialogues with a Negotiating Appointment Setting Interface

Layla El Asri[1,2]**, Romain Laroche**[1] **and Olivier Pietquin**[3]
[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
layla.elasri@orange.com romain.laroche@orange.com
olivier.pietquin@univ-lille1.fr

Abstract

This work takes place within the context of reinforcement learning-based spoken dialogue systems. The DINASTI (DIalogues with a Negotiating Appointment SeTting Interface) corpus contains 1734 dialogues with 21587 system-user exchanges from 385 users. Each dialogue was annotated with automatically computable features and evaluated by the user right after the dialogue. This corpus is meant for research on reinforcement learning modelling for dialogue management.

1 Introduction

NASTIA (Negotiating Appointment SeTting InterfAce) is a French[1] Spoken Dialogue System (SDS) for scheduling an appointment with an engineer in case of landline dysfunction. Similar systems were designed and evaluated during the CLASSiC EU FP7 project[2] [Laroche et al., 2011] about machine learning optimisation for SDS. NASTIA was conceived following this evaluation so that dialogue management would be more suited to user demands concerning the appointment scheduling task. It was tested on 1734 scenario-based dialogues with 385 volunteers who were asked to interact at most 5 times with the system. To allow comparison, we followed the same experimental protocol as the one that served to evaluate the appointment scheduling systems designed during CLASSiC. This paper recalls this protocol and describes the resulting dialogue corpus.

Reinforcement Learning (RL) [Sutton and Barto, 1998] has been a popular technique to optimise the behaviour of SDS [Levin et al., 1997, Williams and Young, 2007, Pietquin and Dutoit, 2006]. NASTIA's dialogue manager is an RL agent implemented as a Module-Variable Decision Process (MVDP) [Laroche et al., 2009]. Dialogue is modelled as a sequence of states and actions among which the system has to choose. For example, NASTIA disposes of different strategies to carry out its task. The appointment scheduling task indeed requires to go through several negotiation rounds with the user in order to find a common availability. During corpus collection, system actions were picked randomly following a uniform probability law. Consequently, given a state of the system, the possible actions were visited approximately the same number of times. We claim that the resulting corpus can be used to infer a reward function, extract the dialogue features that are relevant for reward prediction and carry out research on user simulation. There is a strong need for such corpus in the field of research on RL modelling for SDS [Paek, 2006].

2 NASTIA

NASTIA's dialogue manager is based on an automaton and embeds Reinforcement Learning (RL) *via* the MVDP framework. Dialogue is modelled as a succession of phases, each phase being a node in the automaton. A phase contains one or several points of choice, each of which chooses an action among a set of possibilities according to its current internal state (corresponding to the belief over the dialogue state).

First, NASTIA has a point de choice where it chooses between several negotiation strategies. Three approaches were implemented: User Initiative (UI), System Initiative (SI) and List of Availabilities (LA). UI gives the *dialogue initiative* to the user, asking them: "When would you like to book an appointment?". SI sets the initiative to the system, asking the user to stipulate constraints (week, day, half-day) until only one available slot

[1]In all that follows, system utterances and user answers are translated from French

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

S: **[GREETING]** Welcome to the appointment scheduling service. An engineer must intervene to repair your line. **[ASK_FOR_DAY]** On which day are you available ?
U: Friday, the 16th.
S: **[NOT_AVAILABLE]** There are no available slots on Friday the 16th. **[GIVE_INFO]** This week, only Thursday morning and Saturday afternoon are available.
U: [silence]
S: **[ASK_OPEN_QUESTION]** When would you like to book an appointment ?
U: Thursday the 15th during the morning.
S: **[CONFIRM_APPOINTMENT]** Please confirm the following appointment with yes or no: Thursday, the 15th of July, during the morning.
U: Yes.
S: **[GOODBYE]** Your appointment has been recorded. Thank you for using the service. Goodbye.

Figure 1: An example of dialogue with NASTIA. Dialogue phases are between brackets.

Juillet 2010

	Lundi 12	Mardi 13	Mercredi 14	Jeudi 15	Vendredi 16	Samedi 17	Dimanche 18
Matin	Aujourd'hui		OK				
Après-midi		OK		OK	OK	OK	

	Lundi 19	Mardi 20	Mercredi 21	Jeudi 22	Vendredi 23	Samedi 24	Dimanche 25
matin	OK	OK		OK	OK	OK	
apres-midi							

Figure 2: Example of user calendar for the scenario-based dialogues. The green slots are the available ones.

is identified or user constraints stop matching system availabilities. Finally, LA is about the *task initiative* [Chu-Carroll, 2000]: the system proposes a list of availabilities to the user and asks them to interrupt the listing once a suitable appointment has been proposed. An example of dialogue illustrating the SI and UI strategies is given in Figure 1.

NASTIA has four other points of choice. The second point of choice is about the help message to play after a user has requested it. NASTIA has three possibilities: recall the dialogue context; give to the user the possibility to cancel the help command then recall the dialogue context then recall the available commands (repeat and help); give to the user the possibility to cancel the help command then recall the available commands.

The third point of choice deals with the confirmation strategy. After each user appointment proposition, the system follows one of the three following confirmation strategies. NASTIA may choose not to ask for a confirmation. The implicit confirmation strategy simply consists of repeating what was understood. Finally, following the explicit strategy, NASTIA asks “I understood you were available on [understood date]. Is it correct?”.

The fourth point of choice is visited after a speech recognition rejection or a user time out. The SDS may play a help message or inform the user that they were not understood/heard and wait for them to repeat/say something.

The fifth point of choice decides if the system should provide information about its calendar after an appointment setting failure or after the user has expressed some constraints. For instance, if a user says s/he is available next week, the system may answer “Next week, only Tuesday morning and Friday afternoon are available”.

3 Corpus collection

3.1 Recruitment

All volunteers to the experiment were Orange employees recruited by e-mail. We had 627 answers to the first recruitment campaign. We sent an e-mail to these subscribers with 5 hyperlinks. A code was associated with each hyperlink to make sure each call was unique. Each code was composed of the call identifier (5 digits) and the scenario number (2 digits). A last digit was added for Cyclic Redundancy Check (CRC).

A user guide was attached to the mail explaining the scenario, how to make a call and then fill in the questionnaire. After clicking one of the links, the user was sent to a web page explaining the scenario which was the following:

Today is Monday, July 12th and your landline is non-functional. After it diagnosed that the intervention of an engineer on site was required, the technical service has redirected you to a spoken dialogue system to book an appointment. Your aim is to set an appointment at one of the available slots on the following calendar.

Then the user was displayed a calendar as the one shown on Figure 2.

After performing the call, users filled in the evaluation questionnaire on the same page where the calendar was displayed.

In total, 385 participants made 1 to 5 calls, with an average of 4.6 calls per participant. This resulted in 1734 dialogues and 21587 system-user exchanges, among which 7508 are decision turns, *i.e.* turns where the system needed to choose amongst several actions.

4 Evaluation

Users were asked to fill in an online questionnaire after each dialogue with NASTIA. This questionnaire is translated in Appendix A. Questions 1 and 2 required a yes/no answer. For Question 3, the user had to select the appointment date if an appointment had been set. Questions 4 to 10 were evaluated according to a six-point Likert scale: completely disagree, disagree, mostly disagree, mostly agree, agree, completely agree. Another option was added to Question 5 in case there had been no speech recognition mistakes. For Question 11, the users were asked to rate the dialogue on a scale of 1 to 10. Finally, Question 12 was free text, to report any problem or give a general opinion on the system.

5 Corpus annotation

Corpus annotation was performed on the basis of the parameters described in [Schmitt et al., 2008]. This feature set is made of features returned by the speech recognition, natural language understanding and dialogue management modules.

Nevertheless, DINASTI only includes computable features because it is aimed at online RL [Daubigney et al., 2011, Gasic et al., 2011]. Indeed, a behaviour learnt on the scenario-based corpus will not be perfectly suited for real-life situations. There is indeed a difference of commitment between a user who is pretending to book an appointment and one who is really facing problems with their landline [Laroche et al., 2011]. Moreover, real users' availabilities are not likely to be distributed according the same patterns as the ones proposed in our scenarios.

6 Corpus usage

The corpus was collected for manifold purposes. First, DINASTI may be used for testing feature selection optimisation algorithms. In this line of research, Paek and Chickering [2005] modelled dialogue management as an influence diagram and used a Bayesian structure search algorithm to infer the relevant features for reward prediction. Another method was proposed by Rieser and Lemon [2011] who used correlation-based feature selection to model the state space of a car-embedded SDS. Li et al. [2009] and Chandramohan et al. [2010] also integrated feature selection in RL algorithms for dialogue management. We release the DINASTI corpus to encourage more research on the subject: a feature set may be inferred at each point of choice to optimally predict user satisfaction.

Secondly, another crucial parameter of RL is the reward function. Inverse Reinforcement Learning (IRL, [Russell, 1998, Ng and Russell, 2000, Klein et al., 2012]) learns a reward function from a set of examples where a learning agent follows an optimal policy. Paek and Pieraccini [2008] suggested to apply IRL on Human-Human dialogues to learn a reward function that would provide the SDS with the ability to mimic human operators behaviour. Following this idea, Boularias et al. [2010] learnt a reward function for a POMDP-based SDS in a Wizard-of-Oz (WOZ) setting, where a human expert takes the place of the dialogue manager. The expert is provided with the user interaction as understood by the natural language processing module and, given this noisy written entry, s/he chooses the next action of the system. Another way to learn a reward function is to infer it from a set of evaluated dialogues [Walker, 2000, Sugiyama et al., 2012, El Asri et al., 2012]. User overall evaluation might be used as a reward function to learn an optimal policy for the system but it was shown in [El Asri et al., 2013] that learning was accelerated by inferring from these scores a diffuse reward function. Such a function gives a reward after each system decision instead of waiting for the end of the dialogue. Besides, as said in Section 5, it is important to have a function that can be used online to adapt to real users behaviour.

Finally, research on user simulation [Schatzmann et al., 2006, Pietquin et al., 2009, Chandramohan et al., 2011] may also be carried on the corpus. The negotiation task implies unusual constraints on user simulation design. Indeed, it is not a slot-filling task with a static goal. In DINASTI, the user's goal might change during the dialogue, when an appointment is unavailable. Besides, the user may take over the task or dialogue initiative at any point of the dialogue, which is interesting for user adaptivity and expertise modelling research.

7 Conclusion

This document described a corpus of annotated and evaluated dialogues dedicated to research on reinforcement learning modelling. The potential applications of the corpus are state space and reward function modelling as well as user simulation design. This corpus is now under preparation for publication during the course of next year.

References

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

Senthilkumar Chandramohan, Matthieu Geist, and Olivier Pietquin. Sparse approximate dynamic programming for dialog management. In *Proceedings of SIGDIAL*, 2010.

Senthilkumar Chandramohan, Matthieu Geist, Fabrice Lefèvre, and Olivier Pietquin. User simulation in dialogue systems using inverse reinforcement learning. In *Proceedings of Interspeech*, 2011.

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

Lucie Daubigney, Milica Gasic, Senthilkumar Chandramohan, Matthieu Geist, Olivier Pietquin, and Steve Young. Uncertainty management for on-line optimisation of a pomdp-based large-scale spoken dialogue system. In *Proceedings of Interspeech*, pages 1301–1304, 2011.

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

Layla El Asri, Romain Laroche, and Olivier Pietquin. Reward shaping for statistical optimisation of dialogue management. In *Proceedings of SLSP (to be published)*, 2013.

Milica Gasic, Filip Jurcicek, Blaise Thomson, Kai Yu, and Steve Young. On-line policy optimisation of spoken dialogue systems via live interaction with human subjects. In *Proceedings of IEEE ASRU*, 2011.

Edouard Klein, Matthieu Geist, Bilal PIOT, and Olivier Pietquin. Inverse Reinforcement Learning through Structured Classification. In *Proceedings of NIPS*, 2012.

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.

Lihong Li, Jason D. Williams, and Suhrid Balakrishnan. Reinforcement learning for dialog management using least-squares policy iteration and fast feature selection. In *Proceedings of Interspeech*, 2009.

Andrew Y. Ng and Stuart Russell. Algorithms for inverse reinforcement learning. In *Proceedings of ICML*, pages 663–670, 2000.

Tim Paek. Reinforcement learning for spoken dialogue systems: Comparing strengths and weaknesses for practical deployment. In *Proceedings of Interspeech, Dialog-on-Dialog Workshop*, 2006.

Tim Paek and David M. Chickering. The markov assumption in spoken dialogue management. In *Proceedings of SIGDIAL*, pages 35–44, 2005.

Tim Paek and Roberto Pieraccini. Automating spoken dialogue management design using machine learning : An industry perspective. *Speech Communication*, 50:716–729, 2008.

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.

Olivier Pietquin, Stéphane Rossignol, and Michel Ianotto. Training Bayesian networks for realistic man-machine spoken dialogue simulation. In *Proceedings of IWSDS 2009*, 2009.

Verena Rieser and Oliver Lemon. Learning and evaluation of dialogue strategies for new applications: Empirical methods for optimization from small data sets. *Computational Linguistics*, 37, 2011.

Stuart Russell. Learning agents for uncertain environments (extended abstract). In *Proceedings of COLT*, 1998.

Jost Schatzmann, Karl Weilhammer, Matt Stuttle, and Steve Young. A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *The Knowledge Engineering Review*, 21 (2):97–126, 2006.

Alexander Schmitt, Carolin Hank, and Jackson Liscombe. Detecting problematic dialogs with automated agents. In *Perception in Multimodal Dialogue Systems*, Lecture Notes in Computer Science, pages 72–80. 2008.

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.

Appendix A: Evaluation questionnaire

1. Have you booked an appointment?
2. Was the appointment booked on one of your available slots?
3. When did you book the appointment?
4. During your dialogue with the system, you knew what to say.
5. You could easily recover from system misunderstandings.

6. Understanding the system was easy.
7. The system provided enough information for the dialogue to be easy to follow.
8. The dialogue with the system was efficient.
9. The dialogue with the system was fluid.
10. The system was concise.
11. Overall evaluation.
12. Do you have any remarks or comments?