Lilia, A Showcase for Fast Bootstrap of Conversation-Like Dialogues Based on a Goal-Oriented System

Matthieu Riou, Bassam Jabaian, Stéphane Huet, Fabrice Lefèvre

Lilia, a Showcase for Fast Bootstrap of Conversation-like Dialogues Based on a Goal-oriented System

Matthieu Riou, Bassam Jabaian, Stéphane Huet, and Fabrice Lefèvre

LIA/CERI — Avignon University
339 Chemin des Meinajaries, 84140 Avignon, France
name.surname@univ-avignon.fr

Abstract. Recently many works have proposed to cast human-machine interaction in a sentence generation scheme. Neural networks models can learn how to generate a probable sentence based on the user's statement along with a partial view of the dialogue history. While appealing to some extent, these approaches require huge training sets of general-purpose data and lack a principled way to intertwine language generation with information retrieval from back-end resources to fuel the dialogue with actualised and precise knowledge. As a practical alternative, in this paper, we present Lilia, a showcase for fast bootstrap of conversation-like dialogues based on a goal-oriented system. First, a comparison of goal-oriented and conversational system features is led, then a conversion process is described for the fast bootstrap of a new system, finalised with an on-line training of the system's main components. Lilia is dedicated to a chit-chat task, where speakers exchange viewpoints on a displayed image while trying collaboratively to derive its author's intention. Evaluations with user trials showed its efficiency in a realistic setup.

Keywords: Spoken dialogue systems · Chatbot · Goal-oriented dialogue system · On-line learning.

1 Introduction

While a new avenue of research on end-to-end deep-learning-based dialogue systems has shown promising results lately [18, 24, 27], the need of a huge quantity of data to efficiently train these models remains a major hindrance. In the reported studies, systems are typically trained with large corpora of movie subtitles or forum data, which are suitable for modelling long, open-domain dialogues. But then, systems' developments rely on a small set of reference datasets that may be unavailable for all languages (publicly available corpora are usually in English [25, 4]), or for all new domains of interest. Another difficulty is that they cannot handle entity matching between a knowledge source and utterances. Despite some recent propositions to extend the range of applications of the end-to-end neural-network-based framework to task-oriented systems [24, 10] , the way to connect the external information to inner representation remains fundamentally unsolved [23, 27].

As a consequence, classical modular architectures are still useful in many cases. They basically can be seen as a pipeline of modules processing the audio information from the user; downstream progressive treatments aim to first extract the content (speech recognition), then the meaning (semantic parsing, SP), to finally combine it with previous information (including grounding status) from the dialogue history (belief tracking). In this last module, a policy can decide from a dialogue state representation the next best action to perform according to some global criteria (generally dialogue length and success in reaching user's goal). This in-depth step of dialogue management (DM) can then supply the stream to convey the information back to the user: conversion of the dialogue manager action into utterances by the natural language generation (NLG) module followed by speech synthesis. The HIS architecture [26] offers such a setup, plunged into a global statistical framework accounting for the relationships between the data handled by the main modules of the system. Among other things it allows reinforcement learning of the DM policy. In this system some of the most sample-efficient learning algorithms had been implemented and tested [6], while on-line learning with direct interactions with the user had also been proposed [9]. Even more recently on-line learning has been generalised to the lower-level modules, SP and NLG, with protocols to control the cost of such operations during the system development (as in [8, 20, 15, 27, 16]).

HIS is meant to handle goal-oriented vocal interactions. It allows a system to exchange with users in order to address a particular need in a clearly identified field (make a hotel reservation, consult train timetables, troubleshooting, etc.). Goal-oriented dialogue systems require a database to be able to support domain specific tasks. In order to formulate system responses, entities of the database are matched with the information collected through the dialogue. The DM is responsible for making appropriate dialogue decisions according to the user goal and taking into account some uncertain information (e.g. speech recognition errors, misunderstood speech, etc.). The Partially Observable Markov Decision Process (POMDP) model [12] has been successfully employed in the Spoken Dialogue System (SDS) field [26, 22] as well as in the Human Robot Interaction (HRI) context [14] , due to its capacity to explicitly handle parts of the inherent uncertainty of the information which the system has to deal with (e.g. erroneous speech transcripts, falsely recognised gestures, etc.). In this setup, the agent maintains a distribution over possible dialogue states, referred to as the belief state in the literature, and interacts with its perceived environment using a reinforcement learning (RL) algorithm so as to maximise the expected cumulative discounted reward [21].

In this paper, we report on our investigations of the fast adaptation of such a system to handle conversation-like dialogues. Our underlying goal in this endeavour is to develop a system intended to be used in a neuroscience experiment. From inside an fMRI system, users will interact with a robotic platform, vocally powered by our system, which is live-recorded and displayed inside the head-antenna. Users discuss with the system about an image and they try jointly to elaborate on the message conveyed by the image (see Section 3 for further details). Considering that well-performing solutions can be used directly off-the-shelf for speech recognition and synthesis, the study focuses on adapting the spoken semantic parsing and dialogue management modules only.

The remainder of this paper is organised as follows. After presenting a comparison between a goal-oriented dialogue and a conversation in Section 2, we present some design guidelines, forming a recipe to convert the goal-oriented dialogue system to a conversational one in Section 3. Section 4 provides an experimental study with human evaluations of the proposed approach and we conclude in Section 5.

2 Comparison of Goal-oriented vs Conversational Agents

On the one hand, goal-oriented dialogue agents are designed for a few particular tasks and set up to have highly-focused interactions to get information from the user to help complete the task at stake, by helping her to reach a defined goal (such as making a reservation). On the other hand, conversational systems are designed to mimic the unstructured conversational or 'chats' characteristics of human interactions [11]. The review hereafter intends to outline the most important differences between the two situations.

Of course, one must be aware that most of natural human spoken interactions are in fact a composition of goal-driven and open-minded interleaved turns. The latter in this case generally play a role of social glue between speakers, as pointed out by conversational analysis studies (e.g. in [19]). So the presentation below is somewhat artificial and solely aims at making things clearer in the purpose of the implementation of an artificial interactive system.

The most obvious difference lies in the domain covered by the interactions. In principle, goal-oriented interactions suppose a limited single-domain backdrop. Nevertheless these domains have been largely extended in the recent years, and even some possibilities exist to design multi-domain applications (see for instance [7, 3]). On the contrary, conversational systems are supposed to have no limitation on the discussed topics. No such system has been built so far and this remains a research objective, mainly due to the limited understanding abilities of extant systems. It is worth mentioning here that a conversation can also happen in a restricted domain (such as banter about the weather forecast for instance). And then the distinction should be operated at other levels.

First of them, goal-oriented systems can be characterised by the existence of a back-end that the user wants to access to. It will generally be a database, but can be generalised to any knowledge source from which informational entities can be retrieved. During a conversation it is supposed that the user has no specific desire to know a particular piece of information. Even though it is not contradictory with getting to know things in a casual way, there is no incentive to do so. While conversing users are mainly interested in answers allowing them to pursue their own logic, some general knowledge is sufficient to produce responses that make sense in light of users' turns, most of the time. That is how some conversational systems could be built using a huge quantity of movie subtitles [17]. Not surprisingly, learning how to interact with a user based on movie excerpts does not end up with very coherent and purposeful reactions on behalf of the system, even when some contextual information is added [10]

Another major practical difference between goal-driven and chit-chat discussions lies in the timing. While goal-oriented systems are expected to reach the goal in the shortest possible time, it is almost the opposite for conversational ones. For these latter, the dialogue is supposed to go on as long as the speakers find some interest and motivation in the discussion (and they have available time to spend together). It arises a difficulty in using an automatic process to train such systems as one constraint guiding the learning progress (length penalty) is removed. Indeed, most of recent approaches to train DM by means of RL algorithms relied on two constraints: reach the goal (and collect a reward associated with it) and do it fast (and avoid rewards penalising each turn). Therefore, with only one remaining constraint, the kind of strategy is unclear at the end of the learning process.

Finally, a very important discrepancy between the two interaction types is the role-playing innuendo. In goal-oriented systems, the slave-to-master relationship between the user and the system is implicitly implemented, whereas when conversation is at stake, both speakers are expected to intervene at an equal level. So the conversational system becomes truly mix-initiative (in comparison to user or system-initiative systems), but this implies much more complex exchanges. More information can potentially be conveyed at each turn, with less prior expectation of what can be said. On top of all that, the conversation situation implies that the system might be able to participate in the argumentation about the discussed topics. The argumentative capacity, despite recent very promising works based on the use of game theory [2, 13], remains very difficult to implement in the current architecture. This is a point that our endeavours to convert a goal-oriented system to a conversational interface have not yet been able to address appropriately, as discussed in the next section.

3 Conversion from Goal-oriented to Conversational

In this section we present some principles behind the conversion of a goal-oriented system to its conversational version. The propositions are exemplified in the context of a new system, Lilia, intended to be used for neuroscience experiments in a near future.

The experiments presented in the paper concern a chit-chat dialogue system framed in a goal-oriented dialogue task. In this context, users discuss with the system about an image (out of a small predefined set of 6), and they tried jointly to discover the message conveyed by the image, as described in [5]. The discussion is expected to last only one minute at most. An example of a successful dialogue for the task is reported, from user trials, in Table 1, with the presented image. The dialogue is translated from French and the user inputs are given after automatic speech recognition (_asr) processing.

In order to use a goal-oriented system for such a task, the principle which has been followed is to construct, as the system's back-end, a database containing several hundreds of possible combinations of characteristics of the image. Each

Table 1. Example of a successful dialogue (translated from French). Automatic transcripts of the user's inputs are given.

S	Hello, my name is Lilia. Today, we've got to try to find this image's message. Maybe we could start by discussing its characteristics.
U_asr	*and a head-down apple*
S	And what do you feel about this fruit?
U_asr	*it looks like a superhero with eyes*
S	This apple looks like a hero with eyes. Concerning its look, we can say it is upside-down.
U_asr	*it seems appetising and nice-looking*
S	This apple with eyes really looks like a hero. Concerning its appearance, I would say nice looking.
U_asr	*image want to eat more fruits and natural products superheros mean something to kids*
S	I think that may be related to GMOs. In order to prove that they create strong fruits and vegetables, finally healthy.
U_asr	*you're right*
S	Yes, or more simply it is about recommending youth to eat more fruits in general?
U_asr	*yes may be, good bye*

of them is associated with a hypothesis about the conveyed message. The overall recommended process is summarised in algorithm 1 in its general terms, and its application to the development of Lilia is detailed hereafter.

During its interaction with the system, the user is expected to progressively provide elements about the image, which will make the system select a small subset of matching entity descriptions in the database. From this subset, it can pick other characteristics and present them as its opinion or ultimately select a pre-defined message to return as a plausible explanation of the image purpose. This would allow the user to speak rather freely about the image for several tens of seconds before arguing briefly about the message. Formally no argumentation is possible from the system's side, it can only propose canned messages. Yet by providing variants of surface forms for each of them, it is possible to simulate a progression in the system's idea of the message. For example, in the dialogue displayed in Table 1, the last two system turns are in fact issued from the same DM dialogue act ("inform(message=GMO)") but are converted to two different utterances which give the illusion to respond to each other. Altough this a very limited mechanism to mimic an argumentative capacity on behalf of the system, it appeared to work quite well during user trials, as the next section will show.

So a paramount starting point for designing the new system is to elaborate a dedicated new ontology. It should be built based not only on the expected topics but also on the targeted discussion structure. We illustrate this process for our 'image commentary' domain. The concepts chosen to describe an image have been elicited on the expectation of what a user could say about them. Here we ensure the ontology contains the elements to unroll the first part of

Algorithm 1 Design guidelines for conversation-like dialogues

1: Enumerate possible objects of discussion → **ontology, top slot and values**
2: Elaborate a (small) set of common characteristics → **ontology, leaf slots**
3: Enumerate slot values for each object → **ontology, flat list of slot/value pairs for each object**
4: Tailor ontology to enforce dialogue structure: tag a concluding slot (possible final message of the discussion), and tag several slots as compulsory (the message can be delivered only after users have provided them) → **ontology, structure tags**
5: Generate Cartesian product of all slot/value pairs per object → produce **backend DB**
6: Use ontology to bootstrap semantic parser: keyword spotting with values (or more elaborate, as for instance using ZSSP framework [8]) → **SP**
7: Use ontology to bootstrap a set of generation templates (concatenation of single-slot templates or composition of multi-slot templates) → **NLG**
8: Multi-criteria objective function: → **reward function** for DM RL training
- final step (e.g. informing of a particular final slot after exchanging at least several other slots, see ontology tags)
- length penalty, to preserve global coherence
9: Train system components: → **trained SP and DM policy**
- collect WoZ or human-human data first and batch train or
- direct online training

the conversation on exchanging impressions about image characteristics. The ontology has been kept simple and generic as it is mainly based on the following concepts:

– **Is** describes physical characteristics with the following values: “nice looking”, “rotten”, “upside-down”, etc.
– **Possesses** describes attributes of the fruit, such as: “arm”, “bumps”, etc.
– **Looks like** describes a resemblance of the fruit, with the following values: “human”, “batman”, etc.
– **Seems** describes an emotion or a feeling coming off the fruit: “sad”, “tired”, “appetising”, etc.
– **Color** describes the main colour of the fruit.

For the second part of conversation, delivering a message, it has been observed two sets of images: one with damaged poor-looking fruits with human characteristics (arms, legs, eyes) and another with fruit disguised as superheros looking rather strong (as the apple in Table 1). A dedicated message has been conceived for each group: first the author’s intention was to convince children that even poor-looking rotten fruits were healthy and good to eat, or fruits and vegetables in general are strong and healthy companions, as superheros are usually (for some versions of the message it has even been suggested that it could be a campaign in favour of GMO crops, see Table 1).

Those description concepts induce the system to discuss several characteristics of the image with the user, but their usage also presents some pitfalls. Firstly, when the system is discussing one concept, for example requesting about “Is”, and the user answers with a characteristic of a different concept, the system may keep repeating its request while the user thinks it has answered it. Secondly, the characteristics of a given concept do not necessarily exclude each other. For example, a same fruit can have both characteristics "is=nice looking" and "is=upside-down". To implement that in the goal-oriented system, the back-end database is built as the mere Cartesian product of all the values of the ontology's slots. In the previous case this will result in two distinct DB entities for the same fruit in the database, one having the "nice looking" characteristic, and the other having the "upside-down" one.

The SP module also has to be adapted to the new task. As our goal-oriented system relies on the use of an on-line trained SP module (such as in [8]) no further manual modifications have been necessary at this step. The ontology as described above is instantiated in the model, and each concept is associated with a set of values. In Lilia, 9 concepts are considered for a total of 51 values (so 5.7 values/slot on average). Only the concept of message has been specifically addressed. As the purpose of the dialogue system is to ultimately deliver a message, the message concept can only be requested by the user. Therefore all user inputs proposing a message are labelled as a request, whatever it is said about it, to drive the system to suggest its own message in return. For all concepts the openness of the system will derive from a loose match between surface forms and concept values (the opposite of what is generally required for goal-oriented systems). SP being trained on-line, see below, it was possible to provide the trainers with instructions on how to strive to connect their vocal inputs with the ontology elements: no need to be precise as long as it allows the system to unroll its entity matching process through the turns until the final delivery of the image's message.

On the side of the DM module, the goal-oriented dialogue system was designed to receive only one dialogue act for each user input. This act could carry several concepts (for example "inform(fruit=apple,seems=strong)"), but it could not inform and request at the same time. The most essential act was extracted from the SP outputs and it was the only one to be sent to the dialogue manager. In a conversational-like dialogue, the user is very likely to produce several acts in one sentence. To handle that, all the acts are sent to the dialogue manager as if they were multiple user turns, before the system is asked to respond. As the last user input act is used by the dialogue manager as a feature to choose the next answer, the acts are reordered to have the most important at the end. Here is the complete list of acts priority, from the most important to the least. First the acts which allow the user to request something to the system and expect an immediate answer, in this order: "help, "repeat", "restart", "request", "request alternatives", "request more", "confirm". Then the acts used by the user to inform the system, on which the system would have to bounce back: "negate","deny", "inform", "affirm", "acknowledge". Finally, pragmatic acts related to the overall dialogue management: "bye", "hello", "thank you".

To allow a fast development of the system, an online RL training approach has been retained for the DM. Several instructions have been given to the expert trainers to define its reward function (how she will penalise or compliment the system for its actions, with numerical values). A conversation, by definition, is not supposed to have a precise goal. However, to be able to train the system, we made explicit the notion of success of a dialogue in this case (associated to a strong positive reward). This is a key aspect of the conversion proposed here, to be able to tag a dialogue as successful or not. So it has been proposed to consider a dialogue objectively successful when a message has been said by the system and at least two description concepts have been discussed (no matter who introduced them in the dialogue). To handle difficult examples, the users are prompted to deem a dialogue failed whenever they notice anything they consider bad (too abnormal or unnatural).

This definition of success imposes a minimal dialogue length. In order to avoid unnecessary and redundant turns, a (-1) penalty reward is given at each turn during the DM policy training. And although a conversation has no time limit, generally speaking, the assumption is made that keeping a mechanism to favour the dialogues reaching their goal swiftly is relevant.

This is coherent with a specificity of the task which is that the system does not need to learn to end the dialogue. In final experiments, the dialogue will automatically be interrupted after 1 minute. In both on-line training and test phases, users were asked to end the dialogue themselves by saying bye as soon as it was successful, or when it had lasted too long already. So in a more general view this property can be preserved with an upper bound on the dialogue duration after which the system could decide to hang up.

Since the NLG module has a huge impact on user appreciation, we started with handcraft rules. Each possible dialogue act has one or a few sentence templates, for a total of roughly 80 seed templates in total. Adding different variations for a single act leads to reduce the impression of repetitions. The outputs have been specifically designed to induce the user behaviour. A small reminder of the goal is given at the start of the dialogue.

4 Evaluation

The evaluation of the converted system is presented in this section. In order to evaluate the interest of the on-line learning process, two complementary versions of the system are proposed in comparison. First, **handcraft** is a baseline version of the system without on-line learning; it uses the initial SP module (zero training) and a handcrafted (H) dialogue manager policy. Then, in order to effectively learn on-line the dialogue system, the system's developer needs to be able to both improve the SP and DM models. Therefore, an enhanced version of the system, referred to as **trained** hereafter, is obtained by replacing the initial SP module and the handcrafted dialogue manager policy by on-line learnt ones. The learning protocol proposed to achieve it, referred to as **on-line training** below, directly juxtaposes an adversarial bandit to learn the SP module and a Q-learner reinforcement learning approach to learn the dialogue manager policy following our prior work [16]. The knowledge base of the SP module as well as the DM policy are adapted after each dialogue turn.

In the experiments reported here a GUI interface has been used (a porting to the FurHat robot head platform [1] is planned for the next series in the fMRI context). The platform could rely on the I/O capacities of the Google Chrome web browser for automatic speech recognition and synthesis. Due to the cost of transcribing the user trials, no precise measure of the actual word error rate has been made; our estimation is less than 20% (with surprising variations depending on the period of the day during which the trials were carried out). The synthesis is of good quality, but cannot be used to add prosody information. So it can be perceived as a bit 'flat' every now and then, but not really disturbing, as most of the users noticed.

For **on-line training**, an expert user communicated with the system to train it. Using sample-efficient reinforcement learning algorithms allows us to converge pretty fast in terms of cumulated rewards and success rate. In our case the training session has been limited to 140 dialogues. Then a group of (mostly) naive users tested each model (48 dialogues each, so a total of 12 dialogues performed by each of our 8 users). At the end of each session, the users were asked to give a rating on a scale of 0 (worst) to 5 (best) to the understanding and generation perceived qualities of the system. The number of training dialogues, as well as the number of test sets for each configuration are recalled in Table 2, along with the results.

Table 2. Evaluation of the proposed approach with and without training

Model	**Train (#dial)**	**Test (#dial)**	**Success (%)**	**Avg cum. Reward**	**Sys. Underst. Rate**	**Sys. Gener. Rate**
handcraft	0	94	31	-1.7	1.3	4.1
on-line training	140	96	78	9.3	2.9	4.5

The difference in performance between handcraft and on-line training models (+47% absolute in success rate) shows the impact of the SP adaptation on the overall success of the conversation, along with a better understanding (1.3 for handcraft vs. 2.9 for on-line training). The average cumulated reward rate on the test is directly correlated to the success rate and comes in confirmation of the previous observations. Also, due to a well-tuned template-based generation system, the system generation rate is high (> 4) for all configurations.

From Table 3, it is possible to observe the gap in performance between the initial version of the SP module and after on-line training. For this evaluation a set of 100 utterances were randomly extracted from the user trials and their semantic annotation manually corrected. It was then possible to estimate the precision and recall of the SP outputs w.r.t. their references, and derive an overall F-measure. The measures were compared using or not the concept values in the scoring. It can be observed that after training, SP is more robust to value errors, as the gap of 5% with initial SP (65.5% vs 70.7%) is reduced to 3% (81% vs 84%). But more generally if the performance of the initial low-cost SP (65.5%) was well below standard for such system, the gap is filled after training where an 81% F-score is reached.

It is worth mentioning that in complementary experiments from our prior work [16] the results obtained after **on-line training** seem to suffer of great

Table 3. Semantic Parser module evaluation: initial vs post-on-line training

Model	Complete act			Without value		
	F-Score	Precision	Recall	F-Score	Precision	Recall
handcraft SP	65.5	60.0	72.1	70.7	65.0	77.6
online training SP	81.0	76.3	86.5	84.0	78.9	89.8

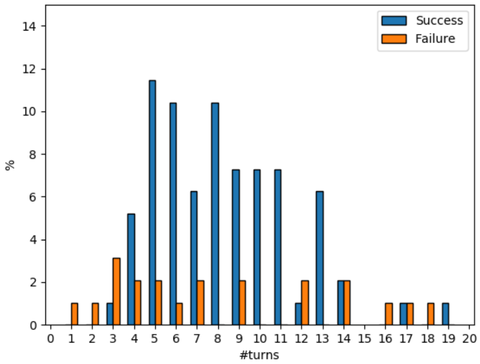

Fig. 1. Distribution of the dialogues w.r.t. the number of turns

variability, depending on the choices made by the expert training the system. The experts have a large margin of action in how they train their system: for instance, they can decide to locally reward only the correct actions (positively), or reversely, only the bad ones (negatively) or ideally, but more costly, both. Also they are free of the inputs used to train the system with: very simple to ensure a steep learning curve or more realistic to try to immediately reach the interesting regions of the DM policy state space. In any case it is worth noticing that the system performance has been shown to increase globally with time in all cases, and so a system can always be further improved to a certain level.

Some more detailed results are given in Table 4. The objective here was to determine if succeeded dialogues and failed ones have distinct features that would allow us to better handle and prevent failed cases in the future. For instance, it was hypothesised that failure could occur from more complex and long interactions from the user. But figures in Table 4 show that there is no such discrepancy between good and bad occurrences: average numbers of turns are very close (8.3 vs 8.1); the same statement applies to time durations (125s vs 130s), or the number of words or concepts by sentence, which are not different enough to give some clues for the reasons of failure.

Table 4. Comparison of successful and unsuccessful dialogues

Success	#dial	Avg #turns	Avg duration (seconds)	Avg #words by sentence	Avg #words by dialogue	Avg #concepts by sentence
success	75	8.3	124.9	7.1	55.4	2.2
failure	21	8.1	130.0	5.9	45.5	2.0
all	96	8.3	126.0	6.9	53.2	2.1

This tendency is further confirmed by looking at how succeeded and failed dialogues spread over the number of turns, as shown in the histograms of Figure 1. The two populations, represented in two distinct series, can be compared (while we did not re-normalise the percentage at each number of turns, to make obvious the difference in population size). It can be observed that success is pretty uniformly spread in $[4, 13]$ and failure alike, in a slightly larger interval $[3, 14]$, with small peaks in both cases (5 for success and 3 for failure). By the way, the targeted duration of the dialogues (60s) is on average doubled. Though departing from the instructions, it should be seen as a good point as it tends to show that users are willing to chat with the system, and are not expeditious as they could be if they had respected their guidelines giving a minute as an objective duration.

5 Conclusion

In this paper a conversion of a goal-oriented human-agent interaction system into a conversational system has been presented. After reviewing the main differences between the two types of interactions, some considerations to redesign a goal-oriented system have been proposed to handle conversation-like dialogues. This fast bootstrap of a goal-oriented system for conversation-liked dialogues is affordable in terms of development cost, and has shown an unexpected good level of performance. The user trials on a chit-chat task in French present a success rate as high as 78%, with very good perceptual ratings from the users (system's understanding and generation quality).
Acknowledgments This workshop has been partially supported by grants ANR-16-CONV-0002 (ILCB), ANR-11-LABX-0036 (BLRI).

References

1. Al Moubayed, S., Beskow, J., Skantze, G., Granström, B.: Furhat: A Back-Projected Human-Like Robot Head for Multiparty Human-Machine Interaction. In: Proceedings of the 2011 international conference on Cognitive Behavioural Systems, pp. 114–130. Springer Berlin Heidelberg (2012)
2. Barlier, M., Perolat, J., Laroche, R., Pietquin, O.: Human-machine dialogue as a stochastic game. In: SIGDIAL (Sep 2015)
3. Budzianowski, P., Ultes, S., Su, P.H., Mrksic, N., Wen, T.H., Casanueva, I., Rojas-Barahona, L., Gasic, M.: Sub-domain Modelling for Dialogue Management with Hierarchical Reinforcement Learning. arXiv.org (2017)
4. Casanueva, I., Budzianowski, P., Su, P.H., Mrkšić, N., Wen, T.H., Ultes, S., Rojas-Barahona, L., Young, S., Gašić, M.: A benchmarking environment for reinforcement learning based task oriented dialogue management. arXiv.org (nov 2017)
5. Chaminade, T.: An experimental approach to study the physiology of natural social interactions. Interaction Studies 18(2), 254–276 (2017)
6. Daubigney, L., Geist, M., Chandramohan, S., Pietquin, O.: A comprehensive reinforcement learning framework for dialogue management optimization. Selected Topics in Signal Processing 6(8), 891–902 (2012)

7. Ekeinhor-Komi, T., Bouraoui, J.L., Laroche, R., Lefèvre, F.: Towards a virtual personal assistant based on a user-defined portfolio of multi-domain vocal applications. In: 2016 IEEE SLT Workshop. pp. 106–113 (2016)
8. Ferreira, E., Jabaian, B., Lefèvre, F.: Online adaptative zero-shot learning spoken language understanding using word-embedding. In: IEEE ICASSP (2015)
9. Ferreira, E., Lefèvre, F.: Reinforcement-learning based dialogue system for human-robot interactions with socially-inspired rewards. Computer Speech & Language 34(1), 256–274 (2015)
10. Ghazvininejad, M., Brockett, C., Chang, M.W., Dolan, B., Gao, J., Yih, W.t., Galley, M.: A Knowledge-Grounded Neural Conversation Model. arXiv.org (2017)
11. Jurafsky, D., Martin, J.H.: Speech and Language Processing, chap. 29, pp. 201–213. Prentice-Hall, Inc., Dialogue Systems and Chatbots, 3 edn. (2017)
12. Kaelbling, L.P., Littman, M.L., Cassandra, A.R.: Planning and acting in partially observable stochastic domains. Artificial Intelligence 101(1-2), 99–134 (1998)
13. Lewis, M., Yarats, D., Dauphin, Y.N., Parikh, D., Batra, D.: Deal or No Deal? End-to-End Learning for Negotiation Dialogues. arXiv.org (2017)
14. Lucignano, L., Cutugno, F., Rossi, S., Finzi, A.: A dialogue system for multimodal human-robot interaction. In: ICMI (2013)
15. Riou, M., Jabaian, B., Huet, S., Lefèvre, F.: Online adaptation of an attention-based neural network for natural language generation. In: INTERSPEECH (2017)
16. Riou, M., Jabaian, B., Huet, S., Lefèvre, F.: Joint on-line learning of a zero-shot spoken semantic parser and a reinforcement learning dialogue manager. In: IEEE ICASSP (2019)
17. Serban, I.V., Sankar, C., Germain, M., Zhang, S., Lin, Z., Subramanian, S., Kim, T., Pieper, M., Chandar, S., Ke, N.R., Mudumba, S., de Brebisson, A., Sotelo, J.M.R., Suhubdy, D., Michalski, V., Nguyen, A., Pineau, J., Bengio, Y.: A Deep Reinforcement Learning Chatbot. arXiv.org (2017)
18. Serban, I.V., Sordoni, A., Bengio, Y., Courville, A., Pineau, J.: Building end-to-end dialogue systems using generative hierarchical neural network models. In: AAAI (2016)
19. Sidnell, J.: Conversation analysis : an introduction. Wiley-Blackwell (2010)
20. Su, P.H., Gasic, M., Mrksic, N., Rojas-Barahona, L., Ultes, S., Vandyke, D., Wen, T.H., Young, S.: On-line active reward learning for policy optimisation in spoken dialogue systems. In: ACL. pp. 2431–2441. Berlin, Germany (aug 2016)
21. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. IEEE Transactions on Neural Networks 9(5), 1054–1054 (1998)
22. Thomson, B., Young, S.: Bayesian update of dialogue state: A pomdp framework for spoken dialogue systems. Computer Speech and Language 24(4), 562–588 (2010)
23. Wen, T.H., Miao, Y., Blunsom, P., Young, S.: Latent Intention Dialogue Models. In: ICML 2017. p. 10 (2017)
24. Wen, T.H., Vandyke, D., Mrkšić, N., Gasic, M., Rojas Barahona, L.M., Su, P.H., Ultes, S., Young, S.: A network-based end-to-end trainable task-oriented dialogue system. In: ACL (2017)
25. Williams, J., Raux, A., Ramachandran, D., Black, A.: The dialog state tracking challenge. In: SIGDIAL (2013)
26. Young, S., Gašić, M., Keizer, S., Mairesse, F., Schatzmann, J., Thomson, B., Yu, K.: The hidden information state model: A practical framework for POMDP-based spoken dialogue management. Computer Speech and Language 24(2), 150–174 (2010)
27. Zhao, T., Eskenazi, M.: Zero-Shot Dialog Generation with Cross-Domain Latent Actions. arXiv.org (may 2018)