A Planning Strategy For Dialogue Management in Healthcare

Milene Santos Teixeira[1,2], Mauro Dragoni[2], and Claudio Eccher[2]

[1] University of Trento, Italy
m.santosteixeira@unitn.it
[2] Fondazione Bruno Kessler, Italy
{dragoni,cleccher}@fbk.eu

Abstract. Planning strategies aim to support an agent in achieving a specific goal. Similarly, dialogue-based strategies are thought for (i) equipping intelligent systems with data-acquisition capabilities, and (ii) supporting users by providing, for example, task orientation. In the digital health domain, the interactions between physicians and patients have the objective of classifying the patients current condition to, subsequently, give him/her new instructions. This work aims to develop a non-deterministic planning based approach for human-machine dialogue management in the health domain. The approach is divided in two parts: the first part focuses on slot-filling dialogues for acquiring information about the patient in order to classify him/her with respect to clusters. The second part concerns planning for the management of a task-oriented dialogue to give advice to patients with the intention of giving some guidance on the patient's treatment. Both parts are supported by a knowledge-based back-end performing reasoning every time new data are provided by users. For demonstrating the complexity of this task, we provide a sample scenario based on the asthma domain showing how, even with few variables, the management of a whole conversation is challenging.

Keywords: Healthcare · Semantic Technologies · Conversational Agents · Planning

1 Introduction

Planning is an important component of rational behaviour, it is part of the human intelligence and reasoning. It commonly precedes acting; we plan whenever we face a situation that is complex or that may offer some high risk or cost that can be aggravated by acting without a plan. In Artificial Intelligence (AI), planning [6] is a process that, given an initial state, searches a sequence of actions that leads to the goal. To achieve this goal, it explores all possible paths by anticipating the outputs of the available actions. In other words, by comprehending the consequence of executing an action, the planner selects and structures these actions in some rational order aiming at achieving a final pre-stated objective.

Dialogue managers (DM) also deal with the problem of action selection. Different approaches have been applied to DM [4] and planning is a promising alternative since, like in a conversation, it aims to achieve a goal. In planning for human-machine dialogue, each utterance will be treated as an action and, by anticipating the outcome of each of these actions, a planner is capable of identifying the states that could be achieved and of choosing the best path that leads to the goal of the dialogue. For example, a health assistant agent that conducts a dialogue with a patient facing an emergency could benefit from planning to optimize the information retrieval and prioritize the use of questions that will identify the type of situation as fast as possible.

The development of methods that implement automated planning to structure and manipulate effective human-machine dialogue is still in its early stages, but it has gained substantial attention in recent years [1, 8, 5]. Several works [9, 2, 7] have identified that planning for dialogue may become challenging due to the huge number of paths that a dialogue can take. In addition, dialogues can be complex since they may have non-deterministic outcomes, the goal may change during the dialogue and user' answers may be partial or even misunderstood by the natural language processor.

In this paper, we propose a novel approach combining reasoning and planning for supporting the effective and efficient management of dialogues. The synergy of these two techniques allows to dynamically update the behavior of conversational agents based on the data provided by users. The reasoner is responsible for inferring the most suitable status of a user (or patient). This activity is performed by exploiting not only the user data and the integrated conceptual model, but also the proper resources of the Linked Open Data (LOD) cloud. On the other hand, the planner is in charge of generating the interactions for supporting a multi-turn conversation with users in order to acquire the missing information enabling the classification of the users' status. This aspect advances the state of the art of conversational agents in healthcare from two perspectives. First, the planner allows to dynamically update the conversation, hence, the system is able to acquire information in a most efficient way. Second, the reasoner enables the access to external knowledge resources that, combined with information provided by users, allow to improve the inference of the users' status.

Considering that domain-independence has always been one of the main concerns in planning [12, 3, 3], for both parts our approach is domain configurable: it expects to receive domain knowledge for the topic of the dialogue, but a plan can be automatically generated by any domain-independent planning engine. The human effort to build a plan is reasonable, since it requires the knowledge of an expert to identify the variables and constraints related to health, but does not require the developer to understand the planning generation process.

The remainder of the paper starts with a brief presentation of the background of our research (Section 2). Then we present our proposed approach (Section 3) followed by the description about how the reasoning process supports the planning activities (Section 4). As an example, we provide a scenario (Section 5) showing the complexity of performing an effective management of conversations. Finally, we discuss our intentions for future work (Section 6).

2 Background

We introduce below some basic notions concerning the background inspired the development of the proposed work. Due to space limit, we are not able to discuss and cite in detail all contribution. Hence, the reader is invited to read the main references provided below for having more details about the mentioned topics.

Automated Planning In AI, automated planning [6] explores the process of planning in order to comprehend and computationally model it. Automated planning is intended for systems that aim to reproduce the human reasoning, presenting an autonomous behaviour. Whenever verified that planning is required for the problem, in order to minimize risks or costs, it should be made before starting to execute any action.

In planning we consider that the world has different *states*. A state can be changed to a new one by the so called actions, which have *preconditions* that must be met to be applicable to that state; and present *effects* that describe what changes in the world after its application. Planning can be viewed as a search problem that describes the sequence of actions that must be applied in the world in order to achieve a goal. As result, a plan is a path that takes an initial state to a goal state.

Dialogue Managers A Dialogue Manager (DM) [10] is responsible for the flow of the conversation in a dialogue system. The DM communicates with the other parts of the dialogue system to (i) receive the input from the user and (ii) to provide the output, which corresponds to the response given by the system in the dialogue. Depending on the chosen design for the DM, the natural language for both input and output can either be processed inside the DM or through an external natural language processing (NLP) module. This is because the focus of the DM is not the NLP, but the strategy that will be used to manage the conversation and decide what to say next.

3 Proposed approach

In this work we propose a plan based approach exploiting knowledge for dialogue management of health systems used to monitor and support patients. Our approach is domain configurable and can be applied to any topic related to health. It consists of two parts: (i) information acquiring and classification of the patient and (ii) orientation.

3.1 Information Acquiring and Classification of the Patient

The first part of our work aims to generate a non-deterministic plan for managing a slot-filling dialogue to retrieve the information that influences the classification of the current health condition (status) of the patient. The generated plan must not only be satisfactory (achieve the goal), it must be as short as possible (optimal) with the aim of detecting critical situations as fast as possible. We provide an architecture for the specification of a planning domain and, from that, an instance of a planning problem can be easily generated. The components necessary to specify the planning domain are represented in Figure 1 and are explained in detail below.

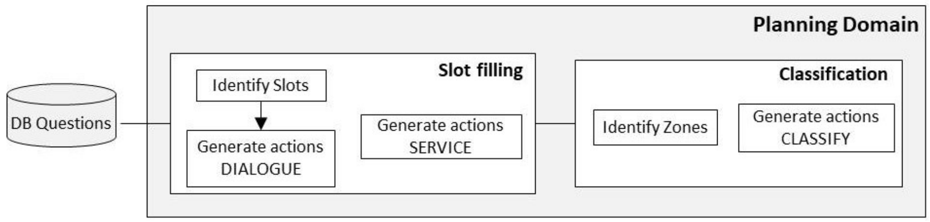

Fig. 1. Proposed architecture to specify the planning domain.

The first step is the identification of the set of slots to be filled through the dialogue with the patient. A slot corresponds to an attribute (e.g. a symptom) whose value is required to classify the patient's health condition. This value is initially fixed as "*unknown*" and the further acceptable values should be identified with the support of expert knowledge. After the specification of this set, each slot is converted to an action that is understandable by a classical planner. This action is identified by a type (*DIALOGUE*) and, basically, corresponds to a question that will be made with the aim of obtaining the value for the slot. We expect open answers for the dialogue, since they reproduce closely a human-human dialogue and do not limit or influence the user's answer or opinion. Consequently, when asking a question (executing the action), it is not possible to know what will be the patient's answer and, therefore, the actions of type *DIALOGUE* hold non-deterministic effects. These effects correspond to the set of acceptable values previously defined for the slot and that are mapped to our model by an external natural language understanding (NLU) component or inferred through the use of a reasoner since the slot to be filled cannot be directly mapped to the content of users' answer but it is the result of a logic analysis of data packages. Details about the interaction between the reasoner and the planner can be found in Section 4. Instead, concerning the use of an NLU component, it is anyway mandatory for extracting from natural language text the intents that the patient wants to communicate to the system as such intents can be directly mapped to slots or provided as input to the reasoner. The investigation of an effective NLU component is out scope of the paper. In our work, we rely on existing NLU strategies.

The actions of type *DIALOGUE* presented in the plan do not contain the actual content of the dialogue. This content could either be generated through natural language generation (NLG) or it can be retrieved from a database. In our work, we are currently using a database containing questions that should lead to the answers that fill out each slot. Considering that a non satisfactory answer can be obtained for a given slot and that repeating an already asked question could lead to a non-natural dialogue, we encourage the storage of several questions associated with a single slot. Consequently, to support the decision on which question to ask, each question is associated with a heuristic value that indicates its relevance or priority. Therefore, during the plan execution, whenever an action of type DIALOGUE is called, a method identifies the type of slot and retrieves from the database the still-non-asked question with the highest heuristic for that slot.

The slot filling dialogue may require further actions that do not correspond to questions. In our approach, these actions are identified by the type *SERVICE* and can be generated to manage the conversation (e.g. setting some constraint). Both types of actions *DIALOGUE* and *SERVICE* were borrowed from [3].

The classification of the patient's status must also be defined in the planning domain according to a set of mutually exclusive zones. A zone can be seen as a label that identifies a status that a patient may be found in and it is composed by a set of slots-values that must hold for this zone to be recognized. Similarly to the process that converted slots to actions, each zone will become an action in the planning domain. An action of type *CLASSIFY* classifies the patient in a zone and, since this classification is our goal, it has as effect a predicate called *GOAL_ACHIEVED*. Consequently, this will be the last action executed in the plan. Finally, the planning problem generation becomes trivial:

- Initial state: all values of slots are set to "*unknown*";
- Goal state: *GOAL_ACHIEVED*

3.2 Orientation

Based on the classification made by the synergistic use of reasoning and planning, the next step consists in the generation of a further plan for managing a task-oriented dialogue able to provide directions to the patient, or to communicate with an external system, in order to act and change the current condition that the patient was found to be in.

This aspect is part of the future work and it presents interesting research challenges to be addressed like the generation of natural language text for explaining the results of the reasoning process and the management of the interaction with two further actors: the healthcare organization that is in charge of managing possible emergencies of its patients and the physician that is responsible for the patient's treatments.

4 The Role of Reasoning

Reasoning plays an important role in inferring the values of attributes exploited for classifying patient's status. If we consider the management of conversations and data acquisition sessions within the healthcare domain, values of attributes can be the result of the analysis of more complex and structured information and not extracted by single question/answer interaction. Let us consider, as example, the monitoring of parameters associated with chronic diseases like the glycemic index for people affected by diabetes or the number and type of cough episodes within a specific time-span for people affected by asthma.

The use of a knowledge base designed for specific domains allows to define classes modeling risky and undesired situations that can be inferred from the analysis of users' data. The use of reasoners enables the detection of such situations and triggers specific actions for the planner and, in turn, for the conversational agent. This kind of approach significantly advances the state of the art of dialogue management, especially within the healthcare domain where aspects like user profiling, tracking, environmental data acquisition, etc. play a crucial role in supporting the interaction with patients.

Our solution goes into this direction by providing the theoretical background necessary for developing a new generation of systems able to manage complex conversations within the healthcare domain for supporting patients better in an effective and efficient way.

Figure 2 shows how the reasoner can support a classic NLU component during the slot-filling activity. The figure refers to a single interaction during a dialog.

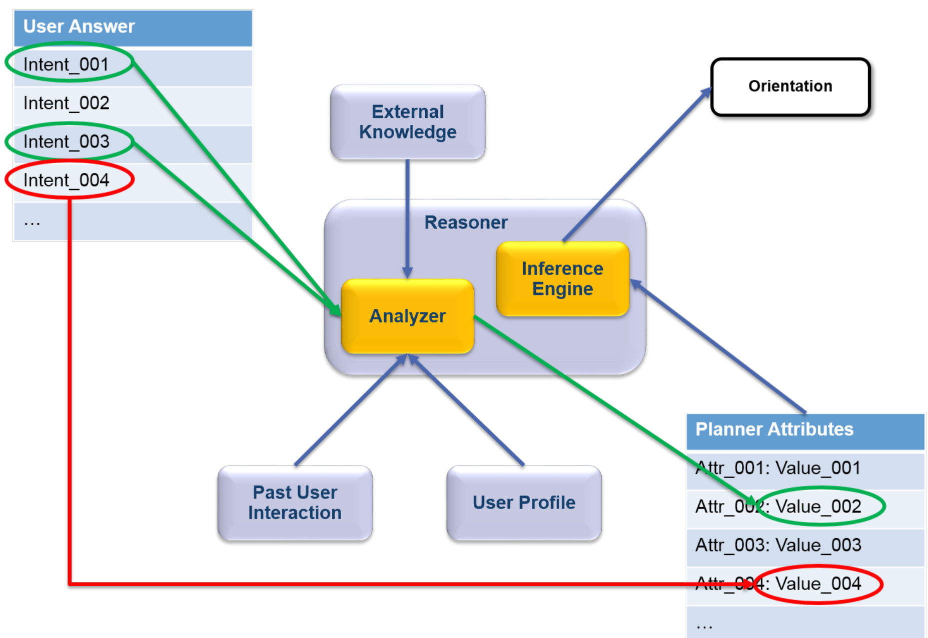

Fig. 2. Representation of the reasoning support to the slot-fill process.

When a user sends an answer to the system, the NLU component processes the natural language text and provides a list of intents that are detected. The intents that the NLU component is able to detect are part of the knowledge provided by the domain experts. Indeed, for every intent the system needs to know what to do, i.e. to fill a slot or to invoke the reasoner.

The reasoner is composed by two modules: the Analyzer and the Inference Engine. The Analyzer is invoked when specific intents are detected within a user's answer. Here, the detected intent is merged with three other kinds of information:

– User Profiles: they contain personal information of the user, e.g. the clinical history stored into his/her Personal Health Record.
– Past User Interactions: the register of all interactions that the user had with the system. Such information is relevant for stream reasoning tasks, recidivity detection, and inference over historical data.
– External Sources: all external knowledge bases, like the LOD Cloud, connected with the system that can be exploited for both inference and slot-filling purposes.

The output of the Analyzer is used for filling the slots of the attributes used by the planner. Once the slots have been filled, the status of planner's variables is sent back to the Inference Engine of the reasoner that tries to infer if the user can be classified within a precise status or not. If the classification is positive, the orientation component is invoked in order to conclude the dialogue with the user. Otherwise, the planner is invoked for updating the dialogue workflow based on the slots that have been filled and for running the subsequent interaction.

5 Sample Scenario

In this Section, we present a simple scenario showing the complexity of managing effective dialogues supporting the self-management of a chronic disease. We chose the asthma domain for two reasons. First, this domain needed the management of conversations that are not limited to single sessions but that require a multi-turn conversation due to the necessity of acquiring information about the different type of symptoms held by the patient as well as possible recent treatments the patient did. Second, the size of domain is limited enough for performing "debugging" operations for both the reasoning and planning activities in order to verify the effectiveness of our solution.

We introduce the example of a chatbot that dialogues with an asthma patient to understand his current condition and give him some orientation. A plan is built to manage this dialogue and retrieve which symptoms the patient hold. To build a very simple scenario, we distinguish only 2 symptoms (slots) that are common for patients with asthma: chest constriction and fatigue. Each slot may hold true, false or *unknown* (default) values and will be converted to an action of type DIALOGUE (a1_d and a2_d, respectively), whose actual content is mapped to a database of questions (e.g. *Have you had episodes of coughing recently*?). Every DIALOGUE action has a non-deterministic effect, since we cannot know in advance which will be the answer given by the patient.

The symptoms can be classified by the actions of type CLASSIFY in 2 different clusters/zones (orange a3_c and red a4_c), for which different orientations are given to the patient. However, to keep this example simple, we consider that the goal is achieved when the classification of a zone is made.

We have defined that the dialogue is system-initiated and the question for the initial state (when all values are unknown) is a0_d, which corresponds to the question "*How are you feeling?*". This question also has a non-deterministic outcome since the patient may inform none, a single symptom or several symptoms (up to 2 in this scenario). This implies that this single question, may result in several different states, more precisely, 2^3.

After generating the planning domain and problem instance, we used PRP planner [11] to generate a strong cyclic plan. The generated plan is illustrated in Figure 3. We can appreciate the complexity of the generated dialogue by considering only two symptoms. The real asthma scenario on which we are working on for our pilot study contains 10 symptoms. Hence, the size of the plan graph dramatically increases as well as its computation complexity.

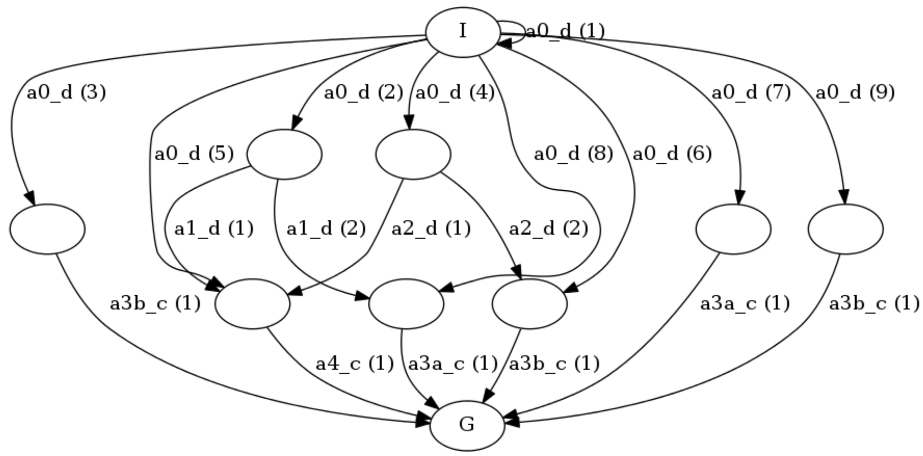

Fig. 3. Generated dialogue plan for the asthma scenario. Suffix _d corresponds to DIALOGUE actions; suffix _c corresponds to CLASSIFY actions

6 Discussion and future work

Our proposed approach facilitates how to reason, plan, and converse in the healthcare domain. It provides a high abstraction level to generate dialogues for different topics, requiring only the knowledge in the topic.

In dialogue systems for the healthcare domain, making the right question at the right moment is a relevant challenge. This is because the patient may be in an emergence state that should be detected and confirmed by the system as fast as possible to execute the right action. In our approach, whenever the system pre-detects a given zone (state), it must try to identify if the patient is in an even more critical zone. Proven that this is not the case (if the state does not hold), the pre-detected zone that identifies that the patient may be in the most critical condition, has priority to be confirmed (or disregarded). If no other slot that influences this zone can be confirmed, the system tries to confirm the previous one. For example, when a symptom that is known to influence a cluster is acknowledged, instead of keep verifying random symptoms, the dialogue should focus on confirming the other symptoms that are crucial to determine that the patient is in this cluster or in a more critical one. So far, this was implemented by associating constraints to each slot defined to the dialogue. But we are exploring new strategies to improve the confirmation of a pre-recognized zone in order to avoid false positives.

The adoption of planning techniques for managing this context is not simple: it has a human cost for modeling the domain and it is a time-consuming process. A relevant challenge that we identified in planning for dialogue is the unreasonable number of states that are generated due to the many possible outcomes (non-determinism) of the dialogue. We intend to find a strategy to minimize this issue and optimize the plan-generation performance, possibly, by discarding the automatically generated alternatives that are computationally true, but unlikely to happen in real life. We also aim to explore the use of open answers by allowing our model to manipulate answers with full, partial or no information at all. These types of answers will be used to improve the heuristic value in the relation slot-question in the database (Section 3.1) and also to relate a slot to the other (e.g. when a question leads to an answer that fills more than one slot), facilitating the decision on which question should be made next.

Finally, with the intention of contributing to planning for dialogue as a whole, we intend to investigate how the characteristics in the health domain can contribute to improve algorithms or methodologies currently used in planning for dialogue in general. From this, we expect that our approach can be applied in dialogue managers for any other domain that requires some expert knowledge for giving orientation, rather than the health domain.

References

1. Bercher, P., Biundo, S., Geier, T., Hoernle, T., Nothdurft, F., Richter, F., Schattenberg, B.: Plan, repair, execute, explainhow planning helps to assemble your home theater. In: Twenty-Fourth International Conference on Automated Planning and Scheduling (2014)
2. Black, E., Coles, A., Bernardini, S.: Automated planning of simple persuasion dialogues. In: International Workshop on Computational Logic and Multi-Agent Systems. pp. 87–104. Springer (2014)
3. Botea, A., Muise, C., Agarwal, S., Alkan, O., Bajgar, O., Daly, E., Kishimoto, A., Lastras, L., Marinescu, R., Ondrej, J., et al.: Generating dialogue agents via automated planning. arXiv preprint arXiv:1902.00771 (2019)
4. Burgan, D.: Dialogue systems and dialogue management. Tech. rep., DST Group Edinburgh Edinburgh SA Australia (2016)
5. Foster, M.E., Petrick, R.P.: Separating representation, reasoning, and implementation for interaction management: Lessons from automated planning. In: Dialogues with Social Robots, pp. 93–107. Springer (2017)
6. Ghallab, M., Nau, D., Traverso, P.: Automated Planning: theory and practice. Elsevier (2004)
7. Kominis, F., Geffner, H.: Multiagent online planning with nested beliefs and dialogue. In: Twenty-Seventh International Conference on Automated Planning and Scheduling (2017)
8. Lee, K., Lee, Y.S., Nam, Y.: A model of fsm-based planner and dialogue supporting system for emergency call services. The Journal of Supercomputing **74**(9), 4603–4612 (2018)
9. Lison, P.: Towards online planning for dialogue management with rich domain knowledge. In: Natural Interaction with Robots, Knowbots and Smartphones, pp. 111–123. Springer (2014)
10. LuperFoy, S., Loehr, D., Duff, D., Miller, K., Reeder, F., Harper, L.: An architecture for dialogue management, context tracking, and pragmatic adaptation in spoken dialogue systems. In: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2. pp. 794–801. Association for Computational Linguistics (1998)
11. Muise, C.J., McIlraith, S.A., Beck, C.: Improved non-deterministic planning by exploiting state relevance. In: Twenty-Second International Conference on Automated Planning and Scheduling (2012)
12. Petrick, R.P., Foster, M.E.: Using general-purpose planning for action selection in human-robot interaction. In: 2016 AAAI Fall Symposium Series (2016)