RADAR-X: An Interactive Interface Pairing Contrastive Explanations with Revised Plan Suggestions

Valmeekam Karthik, Sarath Sreedharan, Sailik Sengupta, Subbarao Kambhampati
CIDSE, Arizona State University, Tempe, AZ 85281 USA
{kvalmeek, ssreedh3, sailiks, rao}@asu.edu

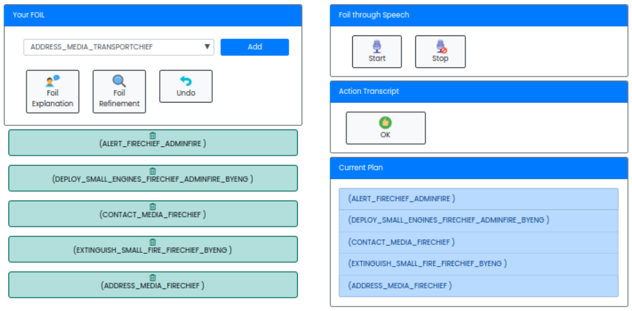
Abstract

Automated Planning techniques can be leveraged to build effective decision support systems that assist and cooperate with the human-in-the-loop. Such systems must provide intuitive explanations when the suggestions made by these systems seem inexplicable to the human. In this regard, we consider scenarios where the user questions the system's suggestion by providing alternatives (referred to as foils). In response, we empower existing decision support technologies to engage in an interactive explanatory dialogue with the user and provide contrastive explanations based on user-specified foils to reach a consensus on proposed decisions. To provide contrastive explanations, we adapt existing techniques in Explainable AI Planning (XAIP). Furthermore, we use this dialog to elicit the user's latent preferences and propose three modes of interaction that use these preferences to provide revised plan suggestions. Finally, we showcase a decision support system that provides all these capabilities.

Introduction

Decision support systems powered by automated planning techniques have been shown to aid humans-in-the-loop in making faster and better decisions (Grover et al. 2020). In scenarios where the expert user is held responsible for the final plan, such systems need to support the user's requirement for explanations if the suggestions made by the system appear inexplicable to the user. While previous works on decision support systems (Grover et al. 2020; Mishra et al. 2019) leverage technologies developed in Explainable AI Planning (XAIP) (Chakraborti et al. 2017; Sreedharan, Kambhampati et al. 2018), the participation of the user in explanatory dialogue is limited; RADAR (Grover et al. 2020) does not allow the user to ask for explanations based on specific queries. This can result in the explanations generated being verbose, making them incomprehensible to the decision-maker. To avoid such situations, the system should let the user drive the dialogue and provide explanations based on the user's query.

In this paper, we propose RADAR−X, an extension of the RADAR system, that supports interactive contrastive explanations (Miller 2019) and uses it as the main vehicle for the interaction between the user and the system. Specifically, we allow explanation generation that caters to specific alternatives by the human (referred to as foils). Further, we view foils as a specification of the user's latent preferences and use them to revise plan suggestions. We discuss two technical challenges that we have to address to design the functionalities of the system (shown in Figure 1). First, we describe updates made to the model-reconciliation framework (Chakraborti et al. 2017) to support generation of constrastive explanations. We look at cases where the foils are specified as partial plans (Kambhampati, Knoblock, and Yang 1995). Second, we consider the case of proactive preference elicitation based on the specified foils that represent the user's latent preference. We look at three different interaction strategies to elicit user preferences and refine plan suggestions.



(ALERT_FIRECHIEF_ADMINFIRE)
(DEPLOY_SMALL_ENGINES_FIRECHIEF_ADMINFIRE_BYENG)
(CONTACT_MEDIA_FIRECHIEF)
(EXTINGUISH_SMALL_FIRE_FIRECHIEF_BYENG)
(ADDRESS_MEDIA_FIRECHIEF)

Figure 1: RADAR−X supports specification of foils, contrastive explanations, and interaction strategies for preference elicitation.

RADAR−X

Demo Video Using the fire-fighting scenario proposed in (Grover et al. 2020), we illustrate the use cases and the functionalities supported by RADAR−X in a demo video. We assume that the system has a model of the task represented as a classical planning problem ($\mathcal{M}^R = \langle \mathcal{D}^R, I^R, G^R \rangle$) that may be different from the human's model ($\mathcal{M}^H = \langle \mathcal{D}^H, I^H, G^H \rangle$) but $\mathcal{M}^H$ is known to the system R beforehand. In this section, we describe the techniques developed for (1) providing contrastive explanations and (2) engaging in proactive preference elicitation via three different interaction mechanisms. The demo video can be found at https://bit.ly/2Uzhciq.

Supporting Contrastive Explanations

A contrastive explanation is generally seen as an answer to questions of the form "Why P and not Q?" (Miller 2019), where P is the fact being explained and Q is the foil or the alternative expected by the explainee (or the user). In decision support systems, a natural way such explanations could arise are cases where the system suggests a plan (the fact being explained) and the user wants to know why the plan they were expecting (the foil) was not chosen. We focus on scenarios where the mismatch between the suggested plan and user's expectation, arise due to model mismatch (Chakraborti et al. 2017). The user's foil represents a set of actions and ordering constraints over these actions; it can be thought of as specifying a set of alternate plans, where every plan in the set includes the specified actions and meets the ordering constraints. The need for explanation arises when the specified foil (1) cannot be part of a (valid) plan in the planner's model ($\mathcal{M}^R$) or (2) is sub-optimal or costlier than the optimal plan suggested by the system. We consider the former case and, further, allow users to raise additional foils after each explanation thereby making the explanation a multi-step procedure. In order to give the explanation, RADAR-X searches in the space of models, starting from the human model ($\mathcal{M}^H$), to find a particular model where the given foil cannot be realized (i.e there exists no valid plan in the model that satisfies the partial plan).[1] This is a modification of the Minimally Complete Explanations Search presented in (Chakraborti et al. 2017). The model difference between the found model and the initial human model is presented as the explanation and can be viewed as the correction that needs to be made in the human's model for refuting the suggested foil.

Proactive Preference Elicitation - Suggesting Plans

Even though the human's model is updated and the human understands that the given foil is invalid, it can still be seen as an indication of some unspecified preferences of the user. Foils can be thus used as a way to identify plans that are closer to the user's expectations. In RADAR-X, we identify such plans using three strategies.

Closest plan approach: In this approach, we look at generating the *closest plan* to the specified foil which implies using the largest part of the foil in the revised plan. For this, we revisit the plan-recognition-as-planning methodology presented in (Ramírez and Geffner 2009) and construct a simple compilation that encodes the partial foils as soft constraints and imposes penalties if any of them are violated when coming up with a plan. This is similar in spirit to (Sohrabi, Riabov, and Udrea 2016). Once the plan suggestion is generated, the user can either accept the plan or engage in recurring interactions to reach a final plan that they prefer.

Conflict sets approach: Even though the plan generated using the above compilation utilizes the largest part of the foil, the actions may have different importance to the user; hence, a planner may choose to use parts of the foil that are less important to the user. Thus, to reach the final plan that the user prefers, they might have to engage in recurring interactions thereby increasing the amount of effort the user has to put in to make sure that the planner generates a plan of his/her liking. A simple attempt to reduce the cognitive load on the user would be to provide all possible sets of conflicting actions in the specified foil and ask the user to resolve them. To find such conflict sets, we employ a systematic breadth-first search in the space of subsets of the foil (similar to that of the Systematic Strengthening (SysS) approach in (Eifler et al. 2020)). We compile each subset into a planning problem (using (Ramírez and Geffner 2009)) and check if the corresponding problem is unsolvable.[2] Subsets corresponding to unsolvable problems are presented to the user to resolve them by removing an action from the set. Using these preferences, a plan suggestion is presented to the user.

Plausible sets approach: In this approach, we do a similar search in the space of subsets and present to the user all the maximal valid subsets of the foil as options to choose from. Maximal valid subsets can be considered as subsets which contain the maximum number of actions from the foil and a plan can be generated using *all* of the actions present in the subset. To find such sets, we use an idea similar to that of Systematic Weakening (SysW) mentioned in (Eifler et al. 2020). We compile each subset into a planning problem (using (Ramírez and Geffner 2009)) and try to generate a plan. An unsolvability test (similar to that of the previous approach) is done before generating the plan to discard unsolvable subsets. In the case of successful plan generation, the corresponding subset is deemed to be valid. Note that subsets that are already part of a valid subset are not checked as we aim to present the maximal plausible sets. Once all the valid subsets are found, they are presented to elicit the preference of the user and then, based on the preference, a plan is suggested.

Further details: A detailed description of the techniques behind RADAR-X can be found in (Valmeekam et al. 2020)

Acknowledgements

The research is supported in part by ONR grants N00014-16-1-2892, N00014-18-1-2442, N00014-18-1-2840, N00014-19-1-2119, AFOSR grant FA9550-18-1-0067, DARPA SAIL-ON grant W911NF-19-2-0006.

[1] We convert the goal condition test for the model space search into a compiled planning problem which is only solvable if there exists a valid completion of the foil in the model. To speed up this test we can rely on faster unsolvability tests that are sound but not complete. For example, in this work, we used h^m based tests in the initial states.

[2] If we relax the requirement of obtaining optimal conflict sets, we can rely on faster unsolvability tests that are sound but not complete; similar to the one mentioned previously.

References

Chakraborti, T.; Sreedharan, S.; Zhang, Y.; and Kambhampati, S. 2017. Plan explanations as model reconciliation: Moving beyond explanation as soliloquy. In *Proc. IJCAI*.

Eifler, R.; Cashmore, M.; Hoffmann, J.; Magazzeni, D.; and Steinmetz, M. 2020. A New Approach to Plan-Space Explanation: Analyzing Plan-Property Dependencies in Oversubscription Planning. In *AAAI*, 9818–9826.

Grover, S.; Sengupta, S.; Chakraborti, T.; Mishra, A. P.; and Kambhampati, S. 2020. RADAR: automated task planning for proactive decision support. *Human–Computer Interaction* 1–26.

Kambhampati, S.; Knoblock, C. A.; and Yang, Q. 1995. Planning as refinement search: A unified framework for evaluating design tradeoffs in partial-order planning. *Artificial Intelligence* .

Miller, T. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence* .

Mishra, A. P.; Sengupta, S.; Sreedharan, S.; Chakraborti, T.; and Kambhampati, S. 2019. Cap: A decision support system for crew scheduling using automated planning. *Naturalistic Decision Making* .

Ramírez, M.; and Geffner, H. 2009. Plan recognition as planning. In *Twenty-First International Joint Conference on Artificial Intelligence*.

Sohrabi, S.; Riabov, A. V.; and Udrea, O. 2016. Plan Recognition as Planning Revisited. In *IJCAI*, 3258–3264.

Sreedharan, S.; Kambhampati, S.; et al. 2018. Handling model uncertainty and multiplicity in explanations via model reconciliation. In *ICAPS*.

Valmeekam, K.; Sreedharan, S.; Sengupta, S.; and Kambhampati, S. 2020. RADAR-X: An Interactive Interface Pairing Contrastive Explanations with Revised Plan Suggestions. In *ICAPS Workshop on Explainable AI Planning (XAIP)*.