Knowledge Representation for Explainability in Collaborative Robotics and Adaptation

Alberto Olivares-Alarcos[1], Sergi Foix[1] and Guillem Alenyà[1]

[1]*Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens i Artigas 4-6, 08028 Barcelona, Spain*

Abstract

In the near future, autonomous robots are going to be used in a large diversity of contexts, interacting and/or collaborating with humans, who will add uncertainty to the task and cause re-planning and online adaptations to the execution of robots' plans. Hence, trustworthy robots must be able to store and retrieve relevant knowledge about their collaborations and adaptations. Furthermore, they shall also use that knowledge to generate explanations for human collaborators. A reasonable approach is first to represent the domain knowledge in triples using an ontology, and then generate natural language explanations from the stored knowledge. In this article, we propose ARE-OCRA, an algorithm that generates explanations about target queries, which are answered by a knowledge base built using an Ontology for Collaborative Robotics and Adaptation (OCRA). The algorithm first queries the knowledge base to retrieve the set of relevant triples that would answer the queries. Then, it generates the explanation in natural language using the triples. We also present the implementation of the core algorithm's routine: construct explanation, which generates the explanations from a set of given triples. We consider three different levels of abstraction, being able to generate explanations for different uses and preferences. This is different from most of the literature works that use ontologies, which only provide a single type of explanation. The least abstract level, the set of triples, is intended for ontology experts and debugging, while the second level, aggregated triples, is inspired by other literature baselines. Finally, the third level of abstraction, which combines the triples' knowledge and the natural language definitions of the ontological terms, is our novel contribution. We showcase the performance of the implementation in a collaborative robotic scenario, showing the generated explanations about the set of OCRA's competency questions. This work is a step forward to explainable agency in collaborative scenarios where robots adapt their plans.

Keywords

explainability, explainable agency, ontology, collaborative robotics, robot plan adaptation

1. Introduction

Throughout the next decades, research and industry are expected to experience several transformations towards autonomous robots that operate in a large spectrum of environments and tasks. This includes scenarios where robots interact and/or collaborate with humans, who would add uncertainty and constraints to the environment. The development of applications where humans and robots closely collaborate, triggers the appearance of several issues such as those related to trustworthiness between the partners. Hence, autonomous collaborative robots

DAO-XAI'21: International Workshop on Data meets Applied Ontologies, September 18–19, 2021, Bratislava, Slovakia

aolivares@iri.upc.edu (A. Olivares-Alarcos); sfoix@iri.upc.edu (S. Foix); galenya@iri.upc.edu (G. Alenyà)
https://aolivaresalarcos.com (A. Olivares-Alarcos)
0000-0002-7733-7715 (A. Olivares-Alarcos); 0000-0001-9249-6696 (S. Foix); 0000-0002-6018-154X (G. Alenyà)

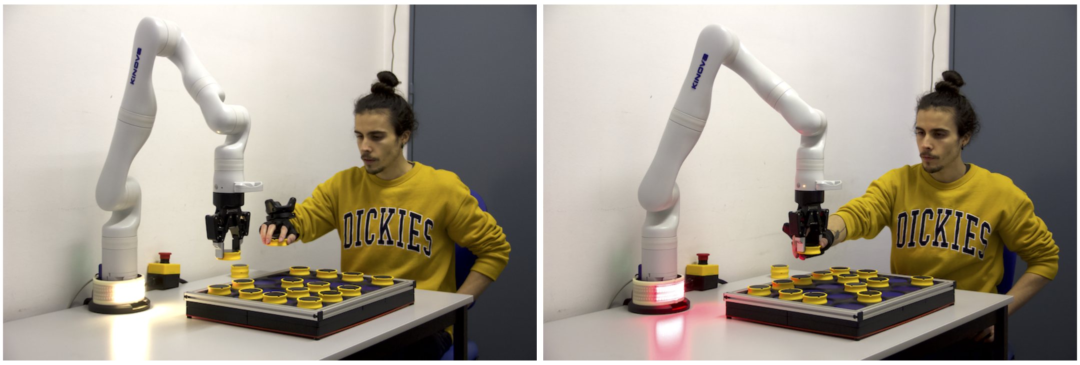
CEUR Workshop Proceedings (CEUR-WS.org)

shall, among others, be able to store and retrieve knowledge about their experiences to explain their decisions and actions. For instance, knowledge about how their tasks' requirements (e.g. safety) and the changes in their environment affect their plan adaptations. Furthermore, each human collaborator might prefer a different type of explanation and robots should be able to provide several types.

Nowadays, there is a need for trustworthy intelligent agents, specially due to the growing trend of using 'black-box' machine learning algorithms. Aligned with this idea, the European General Data Protection Regulation (GDPR) law [1] has considered the right to explanations. Hence, research on eXplainable Artificial Intelligence (XAI) [2] has recently gained significant momentum. Indeed, there are several works on interpreting the results of 'black-box' machine learning mechanisms such as deep neural networks [3]. Furthermore, it is also possible to find other efforts towards explainable agency (i.e. explaining the behavior of goal-driven agents and robots) [4], and explainable automated planning and decision-making [5].

Langley et al. [6], discussed the need for three elements of explainable agency: a representation of the domain knowledge, a way to store the knowledge, and the ability to access and retrieve the knowledge to generate explanations. Knowledge representation formalisms such as ontologies, are commonly used to store and retrieve knowledge in the robotics domain. Indeed, the 1872–2015 IEEE Standard Ontologies for Robotics and Automation [7] presented a core ontology for robotics and automation, which is currently being extended to other robotics' subdomains [8]. Furthermore, ontologies have been widely used for autonomous robotics during the last years [9]. Out of the robotics domain, a few works have used ontologies and RDF triples to generate natural language (NL) explanations [10, 11, 12, 13, 14]. In other works ontologies helped with improving the human understanding of global post-hoc explanations, presented in the form of decision trees [15, 16]. Although inspiring, in none of those works can we find methods that are able to generate different types of explanations, a desirable functionality that has already been considered for the verbalization of robot's plans [17]. Furthermore, to the best of our knowledge, ontologies have not been used for explainable agency in robotic domains yet.

In this article, we explore how storing robots' experiences in a knowledge base, might help to generate human readable explanations about robots' collaborations and plan adaptations. The relevant knowledge is formally represented with OCRA, an Ontology for Collaborative Robotics and Adaptation [18]. Here we present ARE-OCRA, an algorithm to generate explanations about robots' collaborations and plan adaptations using the retrieved facts from the knowledge base. We implement the core algorithm's routine that provides different explanations depending on the level of abstraction, from robot formal knowledge to more human readable formats. The first level reports the set of relevant triples that would answer the target query or queries. This could be used by ontology experts to debug the reasoning system. The second level, inspired by other literature baselines, produces a NL sentence joining the knowledge from the triples with aggregation rules commonly used in NL generation. Note that even though it is inspired by other works, we implemented our own solution due to the lack of working implementations. Furthermore, ours actually includes more aggregation rules than most of the literature approaches. Finally, the third level of abstraction is our novel contribution. It extracts the relevant entities from the triples and inserts them in the available natural language definition of the ontology entities. The implementation is applied to a collaborative robotics scenario, in which a human and a robot share the execution of a task (see Fig. 1). The contributions of this



(a)

(b)

Figure 1: Example of a collaborative task where a human and a robot simultaneously fill a tray with tokens. The human would require explanations about the collaboration (e.g. risk and types) and the robot's plan adaptations. (a) Indirectly physical collaboration, the human and the robot move close to each other without contact. (b) Directly physical collaboration, the human and the robot exchange forces.

work are:

- the design of an algorithm to generate explanations for collaborative robotics and adaptation by means of an ontology;
- an implementation of the core routine: *construct explanation*. It uses a set of relevant triples that are needed to answer the target queries, and generates a natural language explanation in three different levels of abstraction;
- and a qualitative validation in different situations extracted from a real case: a complete collaborative task in which a human and a robot, closely interacting, fill a tray with tokens.

2. ARE-OCRA: Algorithm for Robot Explanation with an Ontology for Collaborative Robotics and Adaptation

Given a human and a robot collaboratively executing a task, we can represent and store the knowledge about their collaboration and adaptations using the ontology OCRA. The algorithm ARE-OCRA (see Alg 1) generates the target explanations about the human-robot task's execution. ARE-OCRA first gets all the relevant instances in our queries (see line 2). They will be instances of the two main event sub-classes of interest in our work: `Collaborations` or `Plan adaptations`. Second, the algorithm extracts and selects a set of relevant and needed triples to explain some target competency questions (line 4). Third, ARE-OCRA uses the set of triples to generate the final explanation in NL (line 5). In this work, we present an implementation of the main routine: *construct explanation*. Hence, we assume that in the knowledge base there is only one target instance (e.g. a `Collaboration`), and that the set of triples has already been generated by querying the knowledge base.

```
Algorithm 1: ARE-OCRA
  Input: Target competency questions (q), abstraction level (a), natural language
         definitions (nl), ontology properties and their inverse (ont_prop)
  Output: Robot's explanation (exp)
1 $exp \leftarrow \emptyset$
2 $target\_instances \leftarrow$ GetTargetEventInstances($q$)
3 foreach $ei \in target\_instances$ do
4     $triples \leftarrow$ GetTriplesFromKnowledgeBase(ei, q)
5     $ei\_exp \leftarrow$ ConstructExplanation($triples, a, nl, ont\_prop$)
6     $exp \leftarrow exp + ei\_exp$
7 end
  Result: Robot's explanation (exp)
```

The *construct explanation* routine (see Alg. 2) is able to generate three different explanations for the same set of triples, depending on the desired level of abstraction.

```
Algorithm 2: Construct explanation routine.
   Input: Relevant and needed set of triples to answer the queries (triples), abstraction
          level (a), natural language definitions (nl), ontology relationships (properties in
          OWL) and their inverse (ont_prop)
   Output: Sentence explaining a set of triples (explanation)
1  $explanation \leftarrow \emptyset$
2  if $a == 1$ then
3      $explanation \leftarrow$ TriplesListToSentenceFirstLevel(triples)
4  else if $a == 2$ then
5      $mod\_triples \leftarrow$ CastTriples($triples$)
6      $mod\_triples \leftarrow$ ClusterTriples($mod\_triples$)
7      $mod\_triples \leftarrow$ OrderTriples($mod\_triples$)
8      $explanation \leftarrow$ GroupTriplesIntoASentence($mod\_triples$)
9  else
10     $target\_nl\_id \leftarrow$ SelectTargetNLDefinitionID($triples$)
11     $target\_nl \leftarrow$ SelectTargetNLDefinition($nl, target\_nl\_id$)
12     $tags \leftarrow$ ExtractTagsFromNLDefinition($nl, target\_nl$)
13     $tags\_knowledge \leftarrow$ GetKnowledgeAboutTags($tags, triples, ont\_prop$)
14     $initial\_sentence \leftarrow$ SubstituteTagsByKnowledge($target\_nl, tags\_knowledge$)
15     $target\_instance \leftarrow$ SelectOntologicalTargetInstance($triples$)
16     $explanation \leftarrow$ AddKnowledgeAboutClass($target\_nl\_id, target\_instance, initial\_sentence$)
17 end
   Result: Sentence explaining a set of triples (explanation)
```

In the *construct explanation* routine, the first level just processes the triples and concatenates them (see line 2). In the second level of abstraction, the implemented routine joins the triples using commonly natural language generation rules [19] (line 4). This level represents the baseline and it is inspired by other works from the literature [11, 12]. Note that until this point, we are generating natural language from the formal axioms of the ontology. Finally, the third level is our novel contribution and it generates the most abstract explanation (line 9). We use the NL definitions of the ontology terms and combine them with the knowledge from the triples. These definitions are distributed together with OCRA, and capture the same notion that is represented in the logical axioms of a term, but in an more human friendly way.

In this maximum level of abstraction, we first choose the NL definition depending on the triples' knowledge (lines 10-11). The triples are prepared so that we can detect which is the main term in the explanation. There is a triple indicating that an entity `is individual of` one of the OCRA's classes (e.g. `Plan adaptation`). Second, we use pre-defined tags to know where to insert the relevant knowledge from the triples (line 12). In the NL definitions, after the mention to the relevant classes appearing in the axioms, we find the tags. Looking into the triples and using the tags, we extract the knowledge that corresponds to each tag (line 13), and we substitute the tag by it (line 14). For instance, in an explanation about a `Plan adaptation`, the NL def. has a tag after the mention to the initial plan: 'is worse plan than'. In the triples, we could find the instance corresponding to the initial robot's plan by using the tag. At this point, we already have the NL definition filled with instances from the triples. Finally, we extract from the triples the instance that is individual of the target ontological entity (e.g. `Plan adaptation`). The final explanation starts with a sentence referring to this instance-class relationship (lines 14-15): 'High risk plan adaptation' is an instance of 'Plan adaptation', followed by the NL definition obtained before.

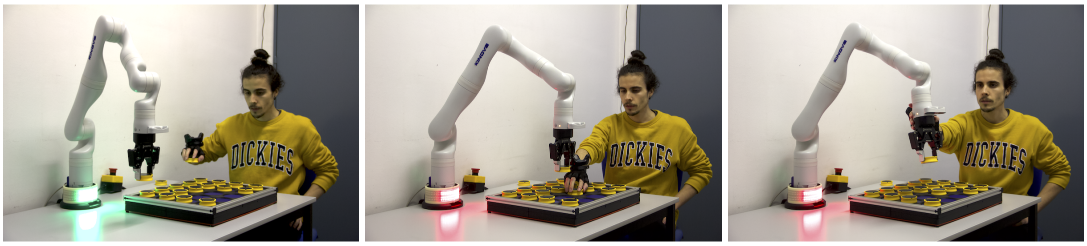
3. Validation: Collaboratively Filling a Tray

In the collaborative task depicted in Fig. 1, there are several situations from which a human might require an explanation. For instance, we can find different risks or types of collaborations, and the robot might adapt to different unexpected situations such a high risk of collision. In this work, we have focused on providing explanations to the competency questions of the ontology OCRA, which are presented in our previous work [18]. An OWL DL formalization of OCRA can be found at the additional material[1]. The competency questions are the set of queries that fix the scope of an ontology. Some examples are:

- **Collaboration questions:** Which and how many collaborations are running now? Which is the plan of a collaboration? Which is the goal of a collaborative plan?
- **Plan adaptation questions:** Which and how many plan adaptations are running now? Which is/are the agent/s participating in the plan adaptation? Why is an adaptation of an agent's plan happening? Which is the plan before and after an adaptation?

We have made available an open source implementation of the routine *construct explanation* together with some use cases with explanations about: the risk and location of a collaboration, collaboration types, and a plan adaptation[1]. In addition, we present one of those cases in detail in this document: a plan adaptation triggered by a predicted high risk of collision (see Fig. 2).

[1] www.iri.upc.edu/groups/perception/ARE-OCRA



(a) (b) (c)

Figure 2: Robot's plan adaptation to a situation of high risk of collision: (a) First the robot executes its initial plan, filling a compartment; (b) then the robot detects a situation of high risk and stops, (c) Finally, the human interacts with the robot while it executes its new plan: stop and remain compliant until a human command is received.

In the proposed plan adaptation, the robot is initially moving towards one of the tray's compartments, to place a token on it. However, the robot detects a potential risky situation of collision with the human, and decides to stop. From that moment on, the robot remains in admittance mode, thus compliant, until the human gives the command to resume its motion. A user might want to know why the robot has stopped, and/or which where the initial and final plans. Having stored the knowledge about the adaptation using OCRA, and extracting the triples from it, our method would generate an explanation in natural language. We are using Knowrob [20, 21] to store and retrieve the knowledge of the collaboration. Hence, we have already tested that we can assert and query the knowledge about the target competency questions. However, recall that in this article we focus on the part of our algorithm in which we generate the explanation using a set of already available triples. In Listings 1 and 2 we can see the explanation that our method would generate for abstraction levels 2 and 3, respectively. The result of the first level is depicted in the additional material [1].

Listing 1: Generated explanation for all competency questions about the plan adaptation 'High Risk Plan Adaptation' with abstraction level 2.

```
'HighRiskPlanAdaptation' isIndividualOf PlanAdaptation and hasParticipant '
KinovaGen3_0'. 'PlaceTokenOnCompartment9' isWorsePlanThan '
StopAndRemainCompliantUntilHumanCommand' and hasComponent '
TrayFullOfTokensUnderSafetyConditions'. 'CollisionRiskIsHigh' isPostconditionOf '
ExecutionOfPlaceTokenOnCompartment9'. 'ExecutionOfPlaceTokenOnCompartment9'
executesPlan 'PlaceTokenOnCompartment9'. '
ExecutionOfStopAndRemainCompliantUntilHumanCommand' executesPlan '
StopAndRemainCompliantUntilHumanCommand'. 'StopAndRemainCompliantUntilHumanCommand
' hasComponent 'TrayFullOfTokensUnderSafetyConditions'.
```

Listing 2: Generated explanation for all competency questions about the plan adaptation 'High Risk Plan Adaptation' with abstraction level 3.

```
'HighRiskPlanAdaptation' is an individual of PlanAdaptation, an Event in which an
Agent ('KinovaGen3_0'), due to its evaluation of the current or expected future
state ('CollisionRiskIsHigh'), changes its current Plan ('PlaceTokenOnCompartment9
') while executing it, into a new Plan ('StopAndRemainCompliantUntilHumanCommand')
, in order to continuously pursue the achievement of the 'plans Goal ('
TrayFullOfTokensUnderSafetyConditions').
```

4. Conclusion

In this work, we have proposed an algorithm (ARE-OCRA) to generate explanations for collaborative robotics and adaptation utilizing an ontology (OCRA). The main routine of the algorithm has already been implemented: *construct explanation.* It uses a set of relevant triples to answer the target queries to be explained, generating a natural language explanation in three different levels of abstraction for different final users. We showcase the performance of the implemented routine in several situations extracted from a realistic collaborative task. Our work enhances the explainability of robots in collaborative situations in which they adapt their plans to unexpected situations. In the future, we first want to finish the implementation of the whole algorithm, also extracting the triples from the knowledge base. We would like to expand the explanation space, considering other parameters such as specificity and locality [17]. Furthermore, it is also interesting to store not only a volatile knowledge base but a whole episodic memory for long-term collaborative explanations. Finally, we also plan to evaluate the different types of explanation with a user study.

References

[1] P. Carey, Data protection: a practical guide to UK and EU law, Oxford University Press, Inc., 2018.

[2] D. Gunning, D. Aha, Darpa's explainable artificial intelligence (xai) program, AI Magazine 40 (2019) 44–58.

[3] Q.-s. Zhang, S.-c. Zhu, Visual interpretability for deep learning: a survey, Frontiers of Information Technology & Electronic Engineering 19 (2018) 27–39.

[4] S. Anjomshoae, A. Najjar, D. Calvaresi, K. Främling, Explainable agents and robots: Results from a systematic literature review, in: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 2019, p. 1078–1088.

[5] T. Chakraborti, S. Sreedharan, S. Kambhampati, The emerging landscape of explainable automated planning decision making, in: C. Bessiere (Ed.), Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, International Joint Conferences on Artificial Intelligence Organization, 2020, pp. 4803–4811. Survey track.

[6] P. Langley, B. Meadows, M. Sridharan, D. Choi, Explainable agency for intelligent autonomous systems, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI'17, AAAI Press, 2017, p. 4762–4763.

[7] C. Schlenoff, E. Prestes, R. Madhavan, P. Goncalves, H. Li, S. Balakirsky, T. Kramer, E. Migueláñez, An ieee standard ontology for robotics and automation, in: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 1337–1342.

[8] S. R. Fiorini, J. Bermejo-Alonso, P. Gonçalves, E. Pignaton de Freitas, A. Olivares Alarcos, J. I. Olszewska, E. Prestes, C. Schlenoff, S. V. Ragavan, S. Redfield, B. Spencer, H. Li, A suite of ontologies for robotics and automation [industrial activities], IEEE Robotics Automation Magazine 24 (2017) 8–11.

[9] A. Olivares-Alarcos, D. Beßler, A. Khamis, P. Goncalves, M. K. Habib, J. Bermejo-Alonso, M. Barreto, M. Diab, J. Rosell, J. Quintas, J. Olszewska, H. Nakawala, E. Pignaton, A. Gyrard, S. Borgo, G. Alenyà, M. Beetz, H. Li, A review and comparison of ontology-based approaches to robot autonomy, The Knowledge Engineering Review 34 (2019) e29.

[10] K. Bontcheva, B. Davis, Natural Language Generation from Ontologies, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 113–127.

[11] I. Androutsopoulos, G. Lampouras, D. Galanis, Generating natural language descriptions from owl ontologies: the naturalowl system, Journal of Artificial Intelligence Research 48 (2013) 671–715.

[12] A.-C. Ngonga Ngomo, D. Moussallem, L. Bühmann, A holistic natural language generation framework for the semantic web, in: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), INCOMA Ltd., Varna, Bulgaria, 2019.

[13] O. Domingo, D. Bergés, R. Cantenys, R. Creus, J. A. Fonollosa, Enhancing sequence-to-sequence modelling for rdf triples to natural text, in: Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+)., 2020, pp. 40–47.

[14] G. Lapalme, Rdfjsrealb: a symbolic approach for generating text from rdf triples, in: Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+), 2020, pp. 144–153.

[15] R. Confalonieri, T. Weyde, T. R. Besold, F. M. d. P. Martín, Trepan reloaded: A knowledge-driven approach to explaining artificial neural networks (2020).

[16] R. Confalonieri, T. Weyde, T. R. Besold, F. Moscoso del Prado Martín, Using ontologies to enhance human understandability of global post-hoc explanations of black-box models, Artificial Intelligence 296 (2021) 103471.

[17] S. Rosenthal, S. P. Selvaraj, M. M. Veloso, Verbalization: Narration of autonomous robot experience., in: IJCAI, volume 16, 2016, pp. 862–868.

[18] A. Olivares-Alarcos, S. Foix, G. Alenyà, Knowledge Representation for Collaborative Robotics and Adaptation, in: Workshop on Unlocking the potential of human-robot collaboration for industrial applications, at 2021 IEEE International Conference on Robotics and Automation (ICRA), 2021.

[19] H. Dalianis, Aggregation in natural language generation, Computational Intelligence 15 (1999) 384–414.

[20] M. Tenorth, M. Beetz, Knowrob – knowledge processing for autonomous personal robots, in: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2009, pp. 4261–4266.

[21] M. Beetz, D. Beßler, A. Haidu, M. Pomarlan, A. K. Bozcuoğlu, G. Bartels, Know rob 2.0 — a 2nd generation knowledge processing framework for cognition-enabled robotic agents, in: 2018 IEEE International Conference on Robotics and Automation (ICRA), 2018, pp. 512–519.