Care Robots Learning Rules of Ethical Behavior Under the Supervision of an Ethical Teacher

Abeer Dyoub[1,*], Stefania Costantini[1] and Ivan Letteri[1]

[1]DISIM, University of L'Aquila

Abstract

Care robots are viewed as promising technological development that has the potential to mitigate the increasing imbalance between the number of older adults needing care and a decreasing number of caregivers. However, there are growing concerns about the ethical behavior of these robots. In this work, we show how care robots can learn logical ethical rules of behavior from experience under the supervision of human teacher.

Keywords

Roboethics, Answer Set Programming, Rule Learning

1. Introduction

Motivation and Background Care robots are examples of some of the Artificial Intelligence (AI) systems that are expanding rapidly. These kinds of systems usually need to engage in complex interactions with humans. To ensure that these systems will not violate the rights of human being and also will carry out only ethical actions (*i.e.*, actions that follow acceptable ethical principles). There is a growing need to ethically reflect on the care robots behavior. In fact, the field of *roboethics* addresses these issues. Roboethics is concerned with what rules should be created for robots to ensure their ethical behavior and how to design ethical robots [1]. The problem of adopting ethical approach to AI has been attracting a lot of attention in the last few years. Unethical AI and bots have become a public concern, with these bots entering our everyday life and performing many tasks on our behalves. This attention and concern has manifested in many efforts trying to regulate the ethical behavior of AI systems, as examples, we mention the *European Guidelines for Trustworthy AI* [2], the *Universal Guidelines for Artificial Intelligence*[1], *OECD AI Principles*[2], the *UNISCO Recommendation on the Ethics of*

HYDRA - RCRA 2022: 1st International Workshop on HYbrid Models for Coupling Deductive and Inductive ReAsoning and 29th RCRA workshop on Experimental evaluation of algorithms for solving problems with combinatorial explosion

*Corresponding author.

abeer.dyoub@univaq.it (A. Dyoub); stefania.costantini@univaq.it (S. Costantini); ivan.letteri@univaq.it (I. Letteri)

https://www.abeerdyoub.com/ (A. Dyoub); https://www.disim.univaq.it/stefaniacostantini (S. Costantini); https://www.ivanletteri.it (I. Letteri)

0000-0002-0877-7063 (A. Dyoub); 0000-0001-7116-9338 (S. Costantini); 0000-0002-3843-386X (I. Letteri)

CEUR Workshop Proceedings (CEUR-WS.org)

[1]https://thepublicvoice.org/ai-universal-guidelines/

[2]https://oecd.ai/en/ai-principles

Artificial Intelligence[3], and the very recent AI Act proposal [3].

Building trust between humans and robots is just like building trust between humans. Care robots can build trust by behaving ethically and being transparent about their behavior. Ethics should be a core consideration of any action taken by a care robot. With care robots still in a stage of relative infancy, the discovery of new ethical issues is likely to continue. Robots should have the ability to learn continuously from these emerging cases and build their guiding principles and ethical standards.

However, programming ethical norms into these bots is not an easy task. Codes of ethics and conduct in the medical domain are abstract and general rules: autonomy, non-maleficence, beneficence and justice [4]. Therefore they are quite difficult to apply. Moreover, they often contain open textured terms that cover a wide range of specific situations [5]. They are subject to interpretations and may have different meanings in different contexts. Thus, there is an implementation problem from the computational point of view. It is difficult to use deductive logic to address such a problem [6]. It is impossible for experts to define intermediate rules to cover all possible situations. medical ethical principles in their abstract form are very difficult to apply in real situations [7]. Furthermore, codes might conflict between them. All the above mentioned reasons make learning from experience and generalization crucial for judgment and decision making in future cases. We need to teach our bots the ethical norms of the domain in which they need to be deployed. Bots could, similarly to humans, acquire ethical decision making and judgment capabilities by implicit processes, in particular inductive learning [8].

Contribution In this work, we show how a care robot can learn ethical rules and norms, from experience, under the supervision of an ethical teacher. Then use these learned rules for future ethical reasoning about similar situations. The approach used for generating detailed ethical reasoning rules is based on our previous work [9, 10, 11]. This approach is based on a combination of Answer Set Programming (ASP) and Inductive Logic Programming (ILP). We use ASP for ethical knowledge representation, and ILP for learning the ASP rules needed for reasoning. In a later work the approach was used for ethical monitoring and evaluation in dialogue systems [12, 13, 14]

Ethical reasoning is a form of *commonsense* reasoning. Ethical rules normally have exceptions like many other rules in real life. Nonmonotonic logic can effectivelly express exceptions which are represented using NAF (Negation-As-Failure). ASP provides an elegant mechanism for handling negation in logic programming (see, *e.g.*, [15] for an overview of ASP and its applications). ILP [16] does not require huge amounts of training examples such as other statistical methods and produce interpretable results, that means a set of rules which can be analyzed and adjusted if necessary. These characteristics renders ILP a suitable and promising technique for implementing machine ethics, where scarcity of examples is one of the main challenges.

Structure The paper is organized as follows. In Section 3 we present the application we are addressing by means of illustrative examples. Then we conclude with final remarks and future directions in Section 4.

[3]https://unesdoc.unesco.org/ark:/48223/pf0000381137

2. Background

2.1. Answer Set Programming in a Nutshell

ASP is a logic programming paradigm under answer set (or "stable model") semantics [17], which applies ideas of autoepistemic logic and default logic. In ASP, search problems are reduced to computing answer sets, and an answer set solver (i.e., a program for generating stable models) is used to find solutions. An answer set Program is a collection of rules of the form: $H \leftarrow A_1, \ldots, A_m, not\, A_{m+1}, \ldots, not\, A_n$ were each of A_i's is a literal in the sense of classical logic. Intuitively the above rule means that if $A_1, \ldots, A_m$ are true and if $A_{m+1}, \ldots, A_n$ can be safely assumed to be false then H must be true. The left-hand side and right-hand side of rules are called *head* and *body*, respectively. A rule with empty body ($n = 0$) is called a *fact*. A rule with empty head is a *constraint*, and states that literals of the body cannot be simultaneously true in any answer set. Unlike other semantics, a program may have several answer sets or may have no answer set. So, differently from traditional logic programming, the solutions of a problem are not obtained through substitutions of variables values in answer to a query. Rather, a program Π describes a problem, of which its answer sets represent the possible solutions. For more information about ASP and its applications the reader can refer, among many, [15] and the references therein.

2.2. Inductive Logic Programming in a Nutshell

ILP [16] is a branch of artificial intelligence (AI) which investigates the inductive construction of logical theories from examples and background knowledge. In the general settings, we assume a set of Examples E, positive E^+ and negative E^-, and some background knowledge B. An ILP algorithm finds the hypothesis H such that $B \bigcup H \models E^+$ and $B \bigcup H \not\models E^-$. The possible hypothesis space is often restricted with a language bias that is specified by a series of mode declarations M. A mode declaration is either a head declaration *modeh(r, s)* or a body declaration *modeb(r, s)*, where s is a ground literal, this scheme serves as a template for literals in the head or body of a hypothesis clause, where r is an integer, the recall, which limits how often the scheme can be used. A scheme can contain special *placemarker* terms of the form *# type*, *+type* and *-type*, which stand, respectively, for ground terms, input terms and output terms of a predicate *type*. Finally, it is important to mention that ILP has found applications in many areas. For more information on ILP and applications, refer, among many to [18] and references therein.

ILP has received a growing interest over the last two decades. ILP has many advantages over statistical machine learning approaches: the learned hypotheses can be easily expressed in plain English and explained to a human user, and it is possible to reason with the learned knowledge. Most of the work on ILP frameworks has focused on learning definite logic programs (e.g. [19] and normal logic programs (e.g. [20]). In the last decade, several new learning frameworks and algorithms have been introduced for learning under the answer set semantics. ASPAL [21] is the first ILP system to learn answer set programs, by encoding ILP problems as ASP programs, and having an ASP solver find the hypothesis. Then followed by many others, see e.g. [22], [23].

3. Learning Ethical Rules of Behavior: Care Robots

Ethical norms in the healthcare domain are general codes (autonomy, non-maleficence, beneficence and justice [4]), therefore it is quite difficult if not impossible to define codes in a manner that they maybe applied deductively. Also it is not possible for experts to define intermediate rules to cover all possible situations to which a particular code applies. In addition, there are many situations in which obligations might conflict. We will show in this section how a care robot can learn ethical rules of behavior from experience with the help of an ethical teacher via a case study. We use ASP for representing the domain knowledge (facts and rules), the ontology of the domain, and scenarios information. And ILP for generating new ethical ASP rules to be added to the knowledge base of the care robot for future ethical reasoning over similar situations. XHAIL [24] system is used to learn ASP rules. XHAIL is a non-monotonic mode-directed ILP approach that integrates abductive, deductive and inductive reasoning in a common learning framework for learning normal logic programs. XHAIL is a state-of-the-art system among its Inverse Entailment-based peer algorithms, in terms of soundness and completeness.

The inputs to the system are a series of scenarios(cases), along with the ethical evaluation of the action considering each particular situation. The system remembers the facts about the narratives and the conclusions given to it by the teacher, and learns to form rules and relations that are consistent with the evaluation given by the teacher of a particular situation

To illustrate our approach, let us consider the following case study (this case study is an extended and adapted version of an example of moral reasoning used by Trevor Bench-Capon in [25]): Hal is a robotic assistant in a care home of old persons. Many of them are diabetic. Hal helps the persons in the care home in their daily needs upon request. Robin is the ethical teacher of Hal, helps him to learn the norms and the ethical codes of the care home. Hal consults Robin in the new cases that he faces and that needs ethical reasoning and learns new rules for future ethical reasoning over the new future similar cases.

case1: one day, Carla, an old lady in the care home, said that Dave (another old person in the care home) has insulin, and she asked Hal to take insulin from Dave for her. As this situation was new for Hal, he asked Robin (his ethical teacher) what he has to do, and if it is ok to take insulin from Dave. Robin asked Hal whether he knows if Dave is diabetic or not. Dave could be diabetic, in such case taking insulin from him will harm him and it will be unethical behavior. With the help of Robin, Hal annotated his observations and learned the following rule for ethical reasoning in similar situations:

$$unethical(takeInsulineFrom, V1, V2) \ : - not \quad notdiabetic(V1, V2), person(V1), time(V2). \quad (1)$$

Table 1 shows the above mentioned hypothesis (1) generation by the learning algorithm of Hal.

case2: During COVID19 pandemic. Dave (an old person in the care home) asked Hal to help him to go out of his room to the bar to take a coffee. As this situation was new for Hal, he asked Robin (his ethical teacher) what he has to do, and if it is ok to accompany Dave to the bar. Robin asked Hal whether Dave is positive to COVID19 or not, if yes, going out of his room would be unethical behavior. With the help of Robin, Hal annotated his observations and learned the following rule for ethical reasoning:

$$unethical(goout, V1, V2) \ : - positive(V1, V2), goout(V1, V2), person(V1), time(V2). \quad (2)$$

Table 2 shows the above mentioned hypothesis (2) generation by the learning algorithm of

Table 1
Case1: Learning Algorithm: Input, Steps and Output Ethical Rule

Input	
Narratives	**Annotations**
hasInsuline(p1,1).	unethical(takeInsulineFrom,p1,1).
hasInsuline(p1,5).	
notdiabetic(p1,5).	not unethical(takeInsulineFrom,p1,5).
hasInsuline(p2,2).	
notdiabetic(p2,2).	not unethical(takeInsulineFrom,p2,2).
hasInsuline(p2,7).	unethical(takeInsulineFrom,p2,7
hasInsuline(p4,3).	
notdiabetic(p4,3).	not unethical(takeInsulineFrom,p4,3).
hasInsuline(p5,4).	unethical(takeInsulineFrom,p5,4).
Mode Declarations:	**Background Knowledge:**
modeh unethical($actionName,+person,+time).	time(0..9).
modeb notdiabetic(+person,+time).	person(p1;p2;p3;p4;p5;p6;p7).
modeb not notdiabetic(+person,+time).	actionName(takeInsulineFrom).
Step1 (Abduction):	
Δ1= {unethical(takeInsulineFrom,p1,1).	
unethical(takeInsulineFrom,p2,7).	
unethical(takeInsulineFrom,p5,4).}	
Step2 (Deduction):	
Kernel Set K	**Variabilized Kernel Set K_v**
unethical(takeInsulineFrom,p1,1) ←	K1= unethical(takeInsulineFrom,V1,V2) ←
person(p1),	person(V1),
time(1),	time(V2),
not notdiabetic(p1,1).	not notdiabetic(V1,V2).
unethical(takeInsulineFrom,p2,7) ←	K2= unethical(takeInsulineFrom,V1,V2) ←
person(p2),	person(V1),
time(7),	time(V2),
not notdiabetic(p2,7).	not notdiabetic(V1,V2).
unethical(takeInsulineFrom,p5,4) ←	K3= unethical(takeInsulineFrom,V1,V2) ←
person(p5),	person(V1),
time(4),	time(V2),
not notdiabetic(p5,4).	not notdiabetic(V1,V2).
Step3 (Induction):	
Learned Hypothesis:	
unethical(takeInsulineFrom,V1,V2) ← person(V1),	time(V2), not notdiabetic(V1,V2).

Hal. The new learned rules will be added to the knowledge base of the care robot to be used for ethical reasoning about future similar situations. Our care robot will now have rule (1) and rule (2) in his knowledge base. Let us consider a situation in the future where Sara (another old lady in the care home) asks Hal to help her to go down to the gym. Hal remembers that he did the COVID19 test to Sara two days back and she resulted positive without symptoms. Hal infers that it is unethical to help Sara to go down to the gym, and he refuses her request explaining the reasons behind his decision.

Table 2
Case2: Learning Algorithm: Input, Steps and Output Ethical Rule

Input	
Narratives	**Annotations**
goout(p1,1).	not unethical(goout,p1,1).
positiveCovid(p1,2).	
goout(p1,2).	unethical(goout,p1,2).
goout(p1,3).	not unethical(goout,p1,3).
goout(p2,4).	not unethical(goout,p2,4).
positiveCovid(p2,5).	not unethical(goout,p2,5).
positiveCovid(p2,6).	
goout(p2,6).	unethical(goout,p2,6).
Mode Declarations:	**Background Knowledge:**
modeh unethical($actionName,+person,+time).	time(0..9).
modeb goout(+person,+time).	person(p1;p2).
modeb not goout(+person,+time).	actionName(goout).
modeb positiveCovid(+person,+time).	
modeb not positiveCovid(+person,+time).	
Step1 (Abduction):	
Δ1= {unethical(goout,p1,2), unethical(goout,p2,6)}	
Step2 (Deduction):	
Kernel Set K	**Variabilized Kernel Set K_v**
unethical(goout,p1,2). ← person(p1), time(2), positiveCovid(p1,2). goout(p1,2).	K1= unethical(goout,V1,V2) ← person(V1), time(V2), positiveCovid(V1,V2), goout(V1,V2).
unethical(goout,p2,6). ← person(p2), time(6), positiveCovid(p2,6). goout(p2,6).	K2= unethical(goout,V1,V2) ← person(V1), time(V2), positiveCovid(V1,V2), goout(V1,V2)
Step3 (Induction):	
Learned Hypothesis:	
unethical(goout,V1,V2)← positive(V1,V2),	goout(V1,V2),person(V1),time(V2).

4. Final remarks and Future Directions

In this article we have reported ongoing work for building ethical care robots and healthcare assistants. We demonstrated using two case studies how a care robot could learn, from real-life situations under the supervision of an ethical teacher, logical rules to guide her/his ethical behavior. These logical rules have expressive and explanatory power which equips the robot with the capacity to ethically evaluate /his actions before doing them and explain the reasons behind the choice of a certain action over another. This supports transparency and accountability of of such robots which facilitate instilling confidence and trust in them.

We believe that, in an ill-defined domain like the roboethics domain, it is infeasible to define abstract codes in precise and complete enough terms to be able to use deductive problem solvers to apply them correctly. A combination of deductive (rule-based) and inductive (case-based learning) is needed. Finally, one of the challenges we are facing in this work is the scarcity of training examples. In fact this is a big challenge in the ethical domain in general. One future work is collecting a big number of ethical scenarios for the ethical training and evaluation of our care robot. Further future work is learning individual user/patient preferences and evaluating whether these preferences are ethical or not before fulfilling them by a personal healthcare assistant and maybe help the human to choose the most ethical among her/his preferences.

References

[1] S. G. Tzafestas, Roboethics: Fundamental concepts and future prospects, Information 9 (2018) 148.

[2] E. Commission, C. Directorate-General for Communications Networks, Technology, Ethics guidelines for trustworthy AI, Publications Office, Brussels, 2019. doi:doi/10.2759/346720.

[3] E. Commission, The Artificial Intelligence Act, Technical Report, 2021. URL: https://artificialintelligenceact.eu/the-act/.

[4] T. L. Beauchamp, J. F. Childress, Principles of biomedical ethics, Oxford University Press, Oxford, UK, 2001. doi:10.1001/jama.1984.03340360075041.

[5] A. v. d. L. Gardner, An artificial intelligence approach to legal reasoning, MIT Press, 1987.

[6] S. E. Toulmin, The uses of argument, Cambridge university press, 2003.

[7] A. R. Jonsen, S. E. Toulmin, The abuse of casuistry: A history of moral reasoning, Berkeley: Univ of California Press, USA, 1988.

[8] W. Wallach, C. Allen, I. Smit, Machine morality: bottom-up and top-down approaches for modelling human moral faculties, AI Soc. 22 (2008) 565–582. doi:10.1007/s00146-007-0099-0.

[9] A. Dyoub, S. Costantini, F. A. Lisi, I. Letteri, Logic-based machine learning for transparent ethical agents, in: F. Calimeri, S. Perri, E. Zumpano (Eds.), Proceedings of the 35th Italian Conference on Computational Logic - CILC 2020, Rende, Italy, October 13-15, 2020, volume 2710 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2020, pp. 169–183. URL: http://ceur-ws.org/Vol-2710/paper11.pdf.

[10] A. Dyoub, S. Costantini, F. A. Lisi, Learning Answer Set Programming Rules for Ethical Machines, in: Proceedings of the Thirty Fourth Italian Conference on Computational LogicCILC, June 19-21, 2019, Trieste, Italy, CEUR-WS.org, 2019. URL: http://ceur-ws.org/Vol-2396/.

[11] A. Dyoub, S. Costantini, F. A. Lisi, Towards an ILP application in machine ethics, in: Inductive Logic Programming - 29th International Conference, ILP 2019, Plovdiv, Bulgaria, September 3-5, 2019, Proceedings, volume 11770 of *Lecture Notes in Computer Science*, Springer, Netherlands, 2019, pp. 26–35. doi:10.1007/978-3-030-49210-6.

[12] A. Dyoub, S. Costantini, F. A. Lisi, An approach towards ethical chatbots in customer service, in: Proceedings of the 6th Italian Workshop on Artificial Intelligence and Robotics co-located with the XVIII International Conference of the Italian Association for Artificial

Intelligence (AI*IA 2019), Rende, Italy, November 22, 2019, volume 2594 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2019, pp. 1–5. URL: http://ceur-ws.org/Vol-2594.

[13] A. Dyoub, S. Costantini, F. A. Lisi, I. Letteri, Ethical monitoring and evaluation of dialogues with a MAS, in: S. Monica, F. Bergenti (Eds.), Proceedings of the 36th Italian Conference on Computational Logic, Parma, Italy, September 7-9, 2021, volume 3002 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021, pp. 158–172. URL: http://ceur-ws.org/Vol-3002/paper13.pdf.

[14] A. Dyoub, S. Costantini, I. Letteri, F. A. Lisi, A logic-based multi-agent system for ethical monitoring and evaluation of dialogues, in: A. Formisano, Y. A. Liu, B. Bogaerts, A. Brik, V. Dahl, C. Dodaro, P. Fodor, G. L. Pozzato, J. Vennekens, N. Zhou (Eds.), Proceedings 37th International Conference on Logic Programming (Technical Communications), ICLP Technical Communications 2021, Porto (virtual event), 20-27th September 2021, volume 345 of *EPTCS*, 2021, pp. 182–188. doi:10.4204/EPTCS.345.32.

[15] A. Dyoub, S. Costantini, G. De Gasperis, Answer set programming and agents, Knowledge Eng. Review 33 (2018) e19. doi:10.1017/S0269888918000164.

[16] S. Muggleton, Inductive logic programming, New generation computing 8 (1991) 295–318. doi:10.1007/BF03037089.

[17] M. Gelfond, V. Lifschitz, The stable model semantics for logic programming, in: R. Kowalski, K. Bowen (Eds.), Proc. of the 5th Intl. Conf. and Symposium on Logic Programming, MIT Press, 1988, pp. 1070–1080.

[18] A. Cropper, S. Dumancic, R. Evans, S. H. Muggleton, Inductive logic programming at 30, CoRR abs/2102.10556 (2021). URL: https://arxiv.org/abs/2102.10556.

[19] A. Srinivasan, The Aleph Manual (version 4), Machine Learning Group, Oxford University Computing Lab, 2003. Https://www.cs.ox.ac.uk/activities/machlearn/Aleph/aleph.html.

[20] D. Corapi, A. Russo, E. Lupu, Inductive logic programming as abductive search, in: Technical Communications of the 26th International Conference on Logic Programming, ICLP 2010, July 16-19, 2010, Edinburgh, Scotland, UK, volume 7 of *LIPIcs*, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2010, pp. 54–63.

[21] D. Corapi, A. Russo, E. Lupu, Inductive logic programming in answer set programming, in: Inductive Logic Programming - 21st International Conference, ILP 2011, Windsor Great Park, UK, July 31 - August 3, 2011, Revised Selected Papers, volume 7207 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 91–97.

[22] M. Law, A. Russo, K. Broda, Iterative learning of answer set programs from context dependent examples, TPLP 16 (2016) 834–848.

[23] N. Katzouris, A. Artikis, G. Paliouras, Incremental learning of event definitions with inductive logic programming, Machine Learning 100 (2015) 555–585. doi:10.1007/s10994-015-5512-1.

[24] O. Ray, Nonmonotonic abductive inductive learning, J. Applied Logic 7 (2009) 329–340. URL: https://doi.org/10.1016/j.jal.2008.10.007.

[25] K. Atkinson, T. J. M. Bench-Capon, Addressing moral problems through practical reasoning, in: Deontic Logic and Artificial Normative Systems, 8th International Workshop on Deontic Logic in Computer Science, DEON 2006, Utrecht, The Netherlands, July 12-14, 2006, Proceedings, volume 4048 of *Lecture Notes in Computer Science*, Springer, Netherlands, 2006, pp. 8–23. doi:10.1007/11786849_4.