Agentified Argumentative Learning

Emanuele De Angelis[1,*,†], Maurizio Proietti[1,†] and Francesca Toni[2,†]

[1]IASI-CNR, Rome, Italy
[2]Imperial College London, UK

Abstract

Argumentative learning amounts to integrating argumentative reasoning into forms of machine learning from examples. Amongst several approaches, *ABA Learning* is a form of argumentative learning that, given a background knowledge, and positive and negative examples, derives an Assumption-Based Argumentation (ABA) framework. The learnt ABA frameworks can be deployed to make run-time inference about previously unseen examples, even after having seen very few positive and negative examples. This inference is determined by (non-)acceptance of examples in extensions of the ABA frameworks. However, it may be impossible to determine definite (non-)acceptance when the learnt ABA frameworks admit no or several extensions. In this paper, we explore how this behaviour can be managed by "agentifying" ABA learning. This agentification amounts to leveraging the use of rules in non-flat ABA frameworks, representing denial integrity constraints, towards definite conclusions. Specifically, agentified ABA Learning can identify actions in the external environment aimed at generating observations for expanding the original ABA frameworks so that they admit extensions and at choosing amongst the extensions of (expanded) ABA frameworks.

Keywords

Assumption-based Argumentation, Agentic AI, Symbolic Learning

1. Introduction

Argumentative learning amounts to integrating argumentative reasoning into forms of machine learning from examples [1]. ABA Learning [2, 3, 4] is a recent approach to argumentative learning, generating Assumption-based Argumentation (ABA) frameworks [5, 6, 7, 8] from (possibly very few) *positive and negative examples* of *learnable predicates*, given an initial ABA framework serving as a *background knowledge*. The ABA frameworks generated by ABA Learning can be deployed to make run-time inference about previously unseen examples, by determining whether these examples are accepted or not in extensions (such as stable extensions [5, 6, 7, 8]) of the frameworks expanded with information about the unseen examples. However, especially when the (positive and negative) examples seen during training are very few, the learnt ABA frameworks may admit no or several extensions, thus leading to the inability to determine definite (non-)acceptance of the new examples. For illustration (see Section 3 for a formalisation with the help of the well-known *Nixon diamond* example [9]), the learnt ABA framework may include (non-defeasible) rules that quakers are pacifists and republicans are militarists, as well as background knowledge that pacifists tend to vote against war and militarists tend to vote for war, but individuals cannot vote for both (here votes are assumptions, each being the contrary of the other). Suppose then that, at inference time, we are interested in determining whether a previously unseen individual $nixon$, who is both quaker and republican, is pacifist or militarist: the learnt ABA framework admits no stable extension (as both voting assumptions need to be accepted, but they are in conflict), and, if modified so that the learnt rules become defeasible, we get two different ABA frameworks accepting conflicting claims about $nixon$. So no conclusion can be drawn about $nixon$.

In this paper we explore how this behaviour can be managed by "agentifying" ABA learning. Like in standard autonomous agent systems [10], ABA Learning agents are able to decide actions to be performed in the external environment (possibly including humans, data repositories and/or other

AI[3] *2025: 9th Workshop on Advances in Argumentation in Artificial Intelligence, September 13, 2025, Rende, Italy*
*Corresponding author.
†These authors contributed equally.
emauele.deangelis@iasi.cnr.it (E. De Angelis); maurizio.proietti@iasi.cnr.it (M. Proietti); ft@ic.ac.uk (F. Toni)
0000-0002-7319-8439 (E. De Angelis); 0000-0003-3835-4931 (M. Proietti); 0000-0001-8194-1459 (F. Toni)

agents) and draw observations from this environment as concerns the actions' outcomes. The actions consist in consulting an external source (human, agent or data repository) in the environment; the observations amount to learning the outcome of the consultation. To continue the $nixon$ illustration (again, see Section 3 for details), an ABA Learning agent will decide to check how $nixon$ voted in the past and, upon observing that he voted for war, extend the learnt ABA framework to obtain a single stable extension where $nixon$ is militarist. The action results from the need to "satisfy" the original background knowledge (that pacifists/militarist tend to vote against/for war but individuals cannot vote for both). This behaviour is aided by the presence, in the ABA frameworks, of rules with "actionable"assumptions (e.g. votes) in the head. Thus, our agentification of ABA Learning needs to extend it beyond *flat* ABA frameworks (without assumptions in the head of rules) that have been the focus of existing approaches to date [2, 3, 4].

This paper introduces the general vision of agentified ABA Learning (Section 4), after providing the core background (Section 2) and formalising the earlier illustration (Section 3). It concludes by discussing directions for future work (Section 5).

Related work Our use of rules with assumptions in the head towards actions aimed at choosing amongst extensions of (expanded) ABA frameworks is reminiscent of enforcement of integrity constraints seen as agents' goals, e.g. in the spirit of [11, 12]. Indeed, non-flat ABA rules can be equivalently understood as denials, under stable extensions [13]. Works on argumentative agents exist (e.g. towards persuasion [14]), including recent ones where agents are based on Large Language Models (e.g. for explainability [15]). Unlike these works, we see agentification as a way to support a form of active learning.

2. Background

An *ABA framework* [5, 6, 7, 8] is a tuple $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{} \rangle$ such that

- $\langle \mathcal{L}, \mathcal{R} \rangle$ is a deductive system, where $\mathcal{L}$ is a *language* and $\mathcal{R}$ is a set of *(inference) rules* of the form $s_0 \leftarrow s_1, \ldots, s_m$ ($m \geq 0, s_i \in \mathcal{L}$, for $1 \leq i \leq m$);
- $\mathcal{A} \subseteq \mathcal{L}$ is a (non-empty) set of *assumptions*;[1]
- $\overline{}$ is a *total mapping* from $\mathcal{A}$ into $\mathcal{L}$, where $\overline{a}$ is the *contrary* of a, for $a \in \mathcal{A}$.

Given a rule $s_0 \leftarrow s_1, \ldots, s_m$, s_0 is the *head* and $s_1, \ldots, s_m$ is the *body*; if $m = 0$ then the body is said to be *empty* (represented as $s_0 \leftarrow$) and the rule is called a *fact*. Elements of $\mathcal{L}$ can be any sentences, but in this paper we focus on ABA frameworks where $\mathcal{L}$ is a finite set of ground atoms. However, we will use *schemata* for rules, assumptions and contraries, using variables, similarly to logic programs, to represent compactly all instances over some underlying universe. We will also use equalities of the form $t_1 = t_2$, where t_1, t_2 are ground terms, and we assume that, for all ground terms t, the fact $t = t \leftarrow$ is in $\mathcal{R}$. In particular, we will feel free to write a fact $p(t) \leftarrow$, with t a (tuple of) terms, as $p(X) \leftarrow X = t$, with X a (tuple of) variables. Unlike other works [2, 3, 4], in this paper ABA frameworks are not required to be *flat*, and thus assumptions can be heads of rules. As customary, we leave $\mathcal{L}$ implicit, and use $\langle \mathcal{R}, \mathcal{A}, \overline{} \rangle$ to stand for $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{} \rangle$.

The semantics of an ABA framework is given in terms of sets of assumptions, called *extensions* (for a formal definition see, e.g., [5, 8, 6]). A set of assumptions $S \subseteq \mathcal{A}$ *attacks* an assumption $\alpha \in \mathcal{A}$ iff there is a finite deduction (i.e., an *argument*) from $S' \subseteq S$ to $\overline{\alpha}$ using rules in $\mathcal{R}$; a set of assumptions $S_1 \subseteq \mathcal{A}$ *attacks* a set of assumptions $S_2 \subseteq \mathcal{A}$ iff S_1 attacks some $\alpha \in S_2$. Then, a set of assumptions $S \subseteq \mathcal{A}$ is *stable* iff S is *conflict-free* (i.e., S does not attack itself), *closed* (i.e., there is no $\alpha \notin S$ such that there is an argument, i.e., a deduction, from some $S' \subseteq S$ to α using rules in $\mathcal{R}$), and S attacks all $\alpha \notin S$. An ABA framework is *satisfiable* iff it admits at least one stable extension. A claim $s \in \mathcal{L}$ is *accepted* in a stable extension Δ of an ABA framework F iff there is an argument from Δ to s using rules of F.

[1] The non-emptiness requirement can always be satisfied [7].

Example 1. *Consider the following non-flat ABA framework $\langle \mathcal{R}, \mathcal{A}, \overline{} \rangle$, where:*[2]

$\mathcal{R} = \{\beta(X) \leftarrow \alpha(X),\ p(1) \leftarrow,\ q(2) \leftarrow,\ r(3) \leftarrow\}$;
$\mathcal{A} = \{\alpha(X),\ \beta(X)\}$;
$\overline{\alpha(X)} = q(X), \quad \overline{\beta(X)} = r(X)$;

for $X \in \{1, 2, 3\}$. A stable extension is given by $\{\alpha(1), \beta(1), \beta(2)\}$. Instead, for example, $\{\alpha(1)\}$ is not stable as it is not closed, $\{\beta(3)\}$ is not stable as it is not conflict-free, and $\{\alpha(1), \beta(1)\}$ is not stable as it does not attack $\beta(2)$.

ABA Learning [2, 3, 4] is a method that, given background knowledge, in the form of a satisfiable ABA framework $F = \langle \mathcal{R}, \mathcal{A}, \overline{} \rangle$, positive examples $\mathcal{E}^+ \subseteq \mathcal{L}$, and negative examples $\mathcal{E}^- \subseteq \mathcal{L}$, derives an ABA framework $F' = \langle \mathcal{R}', \mathcal{A}', \overline{}' \rangle$, with $\mathcal{R} \subseteq \mathcal{R}'$, $\mathcal{A} \subseteq \mathcal{A}'$, $\overline{} \subseteq \overline{}'$, such that 1) F' admits a stable extension Δ, 2) all positive examples are accepted in Δ, and 3) no negative example is accepted in Δ.[3]

ABA Learning makes use of *transformation rules*, including the following ones[4] (1) *rote learning*, which, given a positive example $p(a)$, introduces a new rule $p(X) \leftarrow X = a$, (2) *folding*, which, given rules $H \leftarrow B, C$ and $K \leftarrow B$, derives the new rule $H \leftarrow K, C$, and (3) *assumption introduction*, which, given rule $H \leftarrow B$, introduces an assumption α, with contrary $\overline{\alpha}$, and derives the new rule $H \leftarrow B, \alpha$. As mentioned above, we can freely introduce equalities, which can then occur in rule premises (e.g., in B, C). Thus, for instance, if $\mathcal{R}$ contains two facts $p(a) \leftarrow$ and $q(a) \leftarrow$, we can rewrite them into $p(X) \leftarrow X = a$ and $q(X) \leftarrow X = a$, and by folding, we can derive $p(X) \leftarrow q(X)$. In previous work [2, 3, 4], we have presented various learning algorithms, in the case of flat ABA frameworks, based on these transformation rules. We will see examples of their application in our non-flat setting the next section.

3. ABA Learning through Action and Observation

We illustrate our idea of an ABA Learning agent that proactively interacts with the environment, by means of an example – a variant of the well-known Nixon diamond problem proposed in the field of non-monotonic reasoning [9]. By considering non-flat ABA frameworks, we are able to represent background knowledge in a richer way, also through certain types of denials that enforce a form of integrity constraints. For instance, the following ABA framework $\langle \mathcal{R}, \mathcal{A}, \overline{} \rangle$ represents that individuals a, b are quakers and individual c is a republican and the general information that pacifists vote against war, while militarists vote for war.

$$\begin{aligned} \mathcal{R} = \{ &\ \mathit{quaker}(a) \leftarrow, \quad \mathit{quaker}(b) \leftarrow, \quad \mathit{republican}(c) \leftarrow, \\ &\ \mathit{voted}(X, \mathit{against\_war}) \leftarrow \mathit{pacifist}(X), \quad \mathit{voted}(X, \mathit{pro\_war}) \leftarrow \mathit{militarist}(X) \ \} \end{aligned}$$
$$\mathcal{A} = \{\ \mathit{voted}(X, \mathit{pro\_war}), \quad \mathit{voted}(X, \mathit{against\_war})\ \}$$
$$\overline{\mathit{voted}(X, \mathit{against\_war})} = \mathit{voted}(X, \mathit{pro\_war}) \qquad \overline{\mathit{voted}(X, \mathit{pro\_war})} = \mathit{voted}(X, \mathit{against\_war})$$

This ABA framework is non-flat because the heads of the last two rule schemata in $\mathcal{R}$ are assumptions, $\mathit{voted}(X, \mathit{against\_war})$ and $\mathit{voted}(X, \mathit{pro\_war})$, which are one the contrary of the other. Suppose now that we want to learn an explicit definition of the concepts $\mathit{pacifist}$ and $\mathit{militarist}$ from the background knowledge and the following positive and negative examples:

$$\mathcal{E}^+ = \{\ \mathit{pacifist}(a), \mathit{pacifist}(b), \mathit{militarist}(c)\ \} \quad \mathcal{E}^- = \{\ \mathit{militarist}(a), \mathit{militarist}(b), \mathit{pacifist}(c)\ \}$$

This ABA Learning problem can be solved by applying the transformation rules presented in the previous section, according to one of the ABA Learning algorithms[5] presented in recent work [3, 4, 16].

---

[2]As in [2, 3, 4], we give components of ABA frameworks as schemata, with variables in capital letters implicitly universally quantified with scope the schemata in which they occur.
[3]Note that this definition, originally given for flat ABA frameworks only, naturally extends to our setting, by adopting the notion of acceptance given earlier.
[4]Here we present only the instances of the rules that are sufficient to present the example in the next section. For more extended versions, we refer to previous work [2, 3, 4].
[5]The specific algorithm is not relevant in our development.

By rote learning, we introduce a new (ABA) rule:[6]

ρ_1. $\mathit{pacifist}(X) \leftarrow X = a$.

Then, we rewrite the rule '$quaker(a) \leftarrow$' as '$quaker(X) \leftarrow X = a$' and, by folding, we replace $X = a$ in ρ_1 by its consequence $quaker(X)$. Hence, we get a generalised rule:

ρ_2. $\mathit{pacifist}(X) \leftarrow quaker(X)$.

Similarly, we can learn the rule:

ρ_3. $militarist(X) \leftarrow republican(X)$.

Thus, we have learnt an ABA framework, where quakers are pacifists and republicans are not.

Suppose now that we get a new observation in the form of a new individual, e.g., $nixon$, who is both a quaker and a republican. We can add the new information to the rules of the background knowledge, thereby getting the new set of rules

$\mathcal{R}' = \mathcal{R} \cup \{\rho_2, \rho_3,\ quaker(nixon) \leftarrow,\ republican(nixon) \leftarrow\}$.

Unfortunately, the ABA framework with rules $\mathcal{R}'$ admits no stable extensions, as any closed extension is not conflict-free. Indeed, on one hand, from $quaker(nixon)$, by ρ_2, we get $\mathit{pacifist}(nixon)$, and therefore, by $voted(X, against\_war) \leftarrow \mathit{pacifist}(X)$, we conclude $voted(nixon, against\_war)$. On the other hand, from $republican(nixon)$, by ρ_3, we get $militarist(nixon)$, and therefore, by $voted(X, pro\_war) \leftarrow militarist(X)$, we conclude $voted(nixon, pro\_war)$, which is the contrary of $voted(nixon, against\_war)$. Thus, we can conclude neither $\mathit{pacifist}(nixon)$ nor $militarist(nixon)$.

We now show how to use ABA Learning, together with an act of learning from an external source, to derive a new ABA framework F'' that admits a stable extension, where either $\mathit{pacifist}(nixon)$ or $militarist(nixon)$ is accepted. First of all, the rules learnt thus far are rendered defeasible by applying the assumption introduction transformation and deriving the new rules:

ρ_4. $\mathit{pacifist}(X) \leftarrow quaker(X), normal\_quaker(X)$
ρ_5. $militarist(X) \leftarrow republican(X), normal\_republican(X)$

where $normal\_quaker(X)$ and $normal\_republican(X)$ are new assumptions with contraries

$\overline{normal\_quaker(X)} = abnormal\_quaker(X)$ and
$\overline{normal\_republican(X)} = abnormal\_republican(X)$,

respectively. Now, by rote learning, we may add one of the two rules:

ρ_6. $abnormal\_quaker(X) \leftarrow X = nixon$
ρ_7. $abnormal\_republican(X) \leftarrow X = nixon$

each of which disallowing the deduction of conflicting claims about $nixon$. By folding, from ρ_6 and ρ_7, respectively, we get:

ρ_8. $abnormal\_quaker(X) \leftarrow republican(X)$
ρ_9. $abnormal\_republican(X) \leftarrow quaker(X)$

The two derived ABA frameworks have rules, respectively:

$\mathcal{R} \cup \{\rho_4, \rho_5, \rho_8,\ quaker(nixon) \leftarrow,\ republican(nixon) \leftarrow\}$.
$\mathcal{R} \cup \{\rho_4, \rho_5, \rho_9,\ quaker(nixon) \leftarrow,\ republican(nixon) \leftarrow\}$.

They admit two distinct stable extensions: one accepting *militarist*$(nixon)$ (and hence $voted(nixon, pro\_war)$) and the other accepting *pacifist*$(nixon)$ (and hence $voted(nixon, against\_war)$). To decide which atom between $voted(nixon, pro\_war)$ and $voted(nixon, against\_war)$ should be accepted, we assume that we can issue an action and consult the past voting record for $nixon$. In general, we assume that (some of) the assumptions are *actionable*,that is, they correspond to actions resulting in the acquisition (or rejection) as valid facts of the background knowledge. Suppose that, in our example, the inquiry of $nixon$'s record established $voted(nixon, pro\_war)$. Then, by rote learning, we add the rule:

ρ_{10}. $voted(X, Y) \leftarrow X = nixon, Y = pro\_war$.

[6]We assign identifiers ρ_i to rules for ease of reference.

This last step can be seen as the result of an active learning step. By doing so, we obtain an ABA framework F'' with $\mathcal{R}'' = \mathcal{R} \cup \{\rho_4, \rho_5, \rho_8, \rho_{10}, quaker(nixon) \leftarrow, republican(nixon) \leftarrow\}$, $\mathcal{A}'' = \mathcal{A} \cup \{abnormal\_quaker(X), abnormal\_republican(X)\}$, and the contrary mapping extended to the new assumptions as indicated above. F'' has a single stable extension, which accepts the conclusion $militarist(nixon)$.

4. Vision

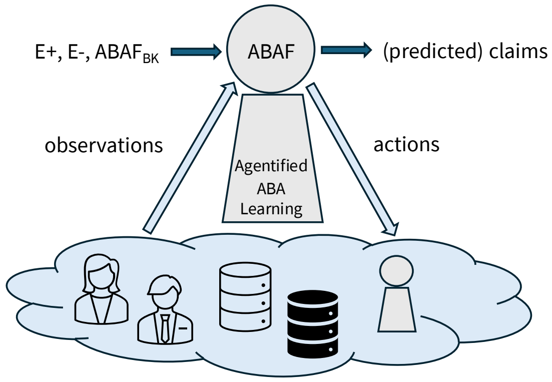

Figure 1: Agentified ABA Learning reasons with the learnt ABAF to determine actions (assumptions) to be performed in the environment and observes the outcomes of those actions to update its ABAF.

Figure 1 summarizes the approach to agentified ABA Learning that we propose in this paper. Conventional ABA Learning, in its original formulation [2, 3, 4, 16], is depicted in the upper part of the figure: it takes positive and negative examples and a background knowledge to generate an ABA framework (ABAF, for short) from which the acceptability of claims, with respect to a given semantics, can be predicted. This process disregards the interaction with the external environment. Agentified ABA Learning aims to enhance this simple schema, and enable ABA Learning to empower agent interactions with the external environment. This may possibly include other agents, humans as well as data repositories, as depicted in the bottom part of Figure 1. We advocate two main novelties: 1) the reliance on non-flat ABAF (whereas originally ABA Learning only considered flat ABAF); and 2) the treatment of some assumptions as actions to be executed in the environment. These actions are autonomously identified to guarantee predictions for previously unseen cases and result in the addition of rules with assumptions as heads to the learnt ABAF, by adapting the same ABA learning process.

5. Conclusions

We have proposed a novel vision for argumentative agents that can learn from examples while autonomously deciding on actions to be executed in their environment to generate targeted expansions of their knowledge. These argumentative agents are supported by an enhancement of ABA Learning, leveraging on non-flat ABA frameworks.

Much future work is ahead of us. First, we need to formally define the enhanced ABA Learning algorithm, catering in particular non-flat ABA frameworks. Second, there is substantial work to be done to realise our approach. To this extent, we plan to use the recent understanding of non-flat ABA frameworks as denial integrity constraints [13] to extend the existing ASP-based implementations of conventional ABA Learning [3, 4]. Third, we plan to explore the use of the enhanced ABA Learning in different types of environment (with humans and/or data repositories and/or other agents), which will require different approaches to render assumptions actionable. For example, if the environment amounts to a human user, then actionability will amount to a conversation with the user that may benefit from a Large Language Model. Lastly, it would be interesting to relate agentified ABA Learning to argumentative forms of *contestable AI* [17, 18], given that the interactions with the environment can be interpreted as contestations that need redressing via argumentative learning.

Acknowledgments

We thank support from the Royal Society, UK (IEC\R2\222045). Toni was partially funded by the ERC (grant agreement No. 101020934) and by J.P. Morgan and the RAEng, UK, under the Research Chairs Fellowships scheme (RCSRF2021\11\45). De Angelis and Proietti were supported by the MUR PRIN 2022

Project DOMAIN funded by the EU – NextGenerationEU (2022TSYYKJ, CUP B53D23013220006, PNRR, M4.C2.1.1), by the PNRR MUR project PE0000013-FAIR (CUP B53C22003630006), and by the INdAM - GNCS Project *Argomentazione Computazionale per apprendimento automatico e modellazione di sistemi intelligenti* (CUP E53C24001950001). De Angelis and Proietti are members of the INdAM-GNCS research group.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] A. Rago, K. Cyras, J. Mumford, O. Cocarascu, Argumentation and machine learning, CoRR abs/2410.23724 (2024). doi:10.48550/ARXIV.2410.23724.

[2] M. Proietti, F. Toni, Learning assumption-based argumentation frameworks, in: ILP 2022, LNCS 13779, Springer, 2024, pp. 100–116. doi:10.1007/978-3-031-55630-2_8.

[3] E. De Angelis, M. Proietti, F. Toni, Learning brave assumption-based argumentation frameworks via ASP, in: ECAI 2024, volume 392 of *FAIA*, IOS Press, 2024, pp. 3445–3452. doi:10.3233/FAIA240896.

[4] E. De Angelis, M. Proietti, F. Toni, Greedy ABA learning for case-based reasoning, in: AAMAS 2025, 2025, p. 556–564.

[5] A. Bondarenko, P. M. Dung, R. A. Kowalski, F. Toni, An abstract, argumentation-theoretic approach to default reasoning, Artif. Intell. 93 (1997) 63–101. doi:10.1016/S0004-3702(97)00015-5.

[6] P. Dung, R. Kowalski, F. Toni, Assumption-based argumentation, in: Argumentation in AI, Springer, 2009, pp. 199–218. doi:10.1007/978-0-387-98197-0_10.

[7] F. Toni, A tutorial on assumption-based argumentation, Argument & Computation 5 (2014) 89–117. doi:10.1080/19462166.2013.869878.

[8] K. Cyras, X. Fan, C. Schulz, F. Toni, Assumption-based argumentation: Disputes, explanations, preferences, FLAP 4 (2017).

[9] R. Reiter, G. Criscuolo, On interacting defaults, in: P. J. Hayes (Ed.), Proceedings of IJCAI '81, William Kaufmann, 1981, pp. 270–276.

[10] M. J. Wooldridge, Introduction to multiagent systems, Wiley, 2002.

[11] R. A. Kowalski, K. Satoh, Obligation as optimal goal satisfaction, J. Philos. Log. 47 (2018) 579–609. doi:10.1007/S10992-017-9440-3.

[12] R. A. Kowalski, F. Sadri, Towards a unified agent architecture that combines rationality with reactivity, in: D. Pedreschi, C. Zaniolo (Eds.), Logic in Databases, International Workshop LID'96, San Miniato, Italy, July 1-2, 1996, Proceedings, volume 1154 of *Lecture Notes in Computer Science*, Springer, 1996, pp. 137–149. doi:10.1007/BFB0031739.

[13] A. Rapberger, M. Ulbricht, F. Toni, On the correspondence of non-flat assumption-based argumentation and logic programming with negation as failure in the head, in: NMR 2024, volume 3835 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2024, pp. 112–121.

[14] A. Rosenfeld, S. Kraus, Strategical argumentative agent for human persuasion, in: ECAI–PAIS 2016, volume 285 of *FAIA*, IOS Press, 2016, pp. 320–328. doi:10.3233/978-1-61499-672-9-320.

[15] S. Hong, L. Xiao, X. Zhang, J. Chen, Argmed-agents: Explainable clinical decision reasoning with LLM disscusion via argumentation schemes, in: BIBM 2024, IEEE, 2024, pp. 5486–5493. doi:10.1109/BIBM62325.2024.10822109.

[16] E. De Angelis, M. Proietti, F. Toni, ABA learning via ASP, in: ICLP 2023, volume 385 of *EPTCS*, 2023, pp. 1–8. doi:10.4204/EPTCS.385.1.

[17] F. Leofante, H. Ayoobi, A. Dejl, G. Freedman, D. Gorur, J. Jiang, G. Paulino-Passos, A. Rago, A. Rapberger, F. Russo, X. Yin, D. Zhang, F. Toni, Contestable AI Needs Computational Argumentation, in: KR 2024, 2024, pp. 888–896. doi:10.24963/kr.2024/83.

[18] V. Dignum, L. Michael, J. C. Nieves, M. Slavkovik, J. Suarez, A. Theodorou, Contesting black-box AI decisions, in: AAMAS 2025, 2025, p. 2854–2858.