February 2024

Using Default Logic to Create Adaptable User Models for Behavior Support Agents

Johanna WOLFF [a,1] Victor DE BOER [b]Dirk HEYLEN [a]and
M. Birna VAN RIEMSDIJK [a]

[a] *University of Twente*
[b] *Vrije Universiteit Amsterdam*

Abstract. Behavior support agents can assist a user in reaching their goals by suggesting suitable actions. In order for these agents to be effective, the agent's advice should be personalized to the user's needs and preferences. However, the way context influences the user, the internal state of the user and the user's desired behavior are all subject to change while the agent is in use. If the agent is not able to adapt to these changes, this can lead to a misalignment between the user and the agent. By making the reasoning of the agent explicit, we can allow the user to directly interact with the agent's user model in order to resolve possible misalignments. We propose to use ordered default logic to reason about the user model as its defeasible nature is inherently well suited to model behavior patterns and routines which may have exceptions dependent on the context. We then analyze different misalignment scenarios and describe how we can use various belief revision techniques to update the agent's user model and resolve these misalignments.

Keywords. Default Logic, Non-Monotonic Logics, Belief Revision, Human-Machine Alignment, User Modeling, Behavior Support Agents

1. Introduction

Artificial assistants which are designed to help their users change their behavior and adopt new routines [1] are becoming increasingly popular. These agents are most effective when they are personalized to the user's needs and preferences [2,3]. Beyond that, artificial support agents are increasingly expected to work as a team together with the human user [4]. However, the way context influences the user, the internal state of the user and the user's desired behavior are all subject to change while the agent is in use. The agent needs to be able to adapt to these changes in order to support the user over a longer period of time [3]. This ensures not only the effectiveness of the agent but also that the user remains in control of how they use the technology [4,5].

Machine learning techniques can be used to achieve a high degree of personalization [6], but these data-driven approaches can also make it difficult to update or influence the knowledge base directly when the user changes their behavior or preferences. This is because relevant concepts are often not explicitly represented, making it unclear how to input new information and which effect this may have on the agents output [7].

[1] Corresponding Author

To allow the user to interact with the reasoning of the agent directly and make necessary changes in case of a misalignment, we use knowledge-based methods to represent this explicitly. In particular, we propose a way to express the user model of a behavior support agent in ordered default logic (Section 2). As default logic uses defeasible reasoning, this allows us to draw tentative conclusions based on incomplete information about the world. The default rules are used to represent the agent's beliefs about the user, in particular about their behavior patterns and routines (Section 3). By including a preference relation on these defaults we can represent the user's preferences and priorities.

If the user feels like the agent is not providing optimal suggestions, this needs to be resolved by adapting the agent's reasoning. We analyze different misalignment scenarios based on [8] and show that these can each be resolved by performing an appropriate belief revision update on the default logic (Section 4).

Example 1 *Throughout this paper we will illustrate our work using an example of a simple scheduling agent which helps the user find time to exercise. By taking the user's schedule, goals, routines and preferences into consideration, the agent attempts to find the best possible suggestion. The agent is also able to process information about certain contexts, and otherwise relies on the user to add additional information which is needed.*

2. Preliminaries

Default logic was first introduced in [9] as a way to reason with beliefs which may need to be rejected when additional information is obtained. The logic is characterized by the introduction of default rules of the form

$$\frac{\text{Prerequisite} \quad : \text{Justification}}{\text{Consequent}}\ \delta$$

which express that if the prerequisite is given and there is no proof that the justification is false, then we infer the consequent.

In this paper we use the framework introduced in [10], which includes an ordering on the default rules of a theory. A theory of this ordered default logic can be translated into standard default logic, allowing for an implementation in theorem provers for standard default logic. An ordered default theory has the form $T = (W, D, <)$ where W is a set of proposition logic formulas, D is a set of default rules and $<$ is a strict partial order on D. The sentences in W describe our, possibly incomplete, knowledge of the world, while the default rules in D allow us to derive additional information based on our beliefs. For two default rules δ_1, δ_2, we take $\delta_1 < \delta_2$ to mean that δ_1 can only be applied after δ_2 has either been applied or blocked. A default rule can be blocked either because the prerequisite cannot be proven or the negation of the justification has been proven.

A minimal set of sentences E which contains W and is deductively closed regarding both the default rules in D and standard logical inference is called an extension of the theory T. In [10] formal requirements are given which enforce the existence of a consistent extension, the detailed proof is taken from [11]. The technical details are outside the scope of this paper, but necessary to ensure the usability of the framework.

The logic of belief revision is concerned with the formalization of different change operators on a set of beliefs. In particular, these operations should be in line with relevant rationality postulates and result in a consistent set of beliefs. One of most dominant theories of belief revision is the AGM model [12]. We will be using the following operators based on [13].

Operator	Effect
$W \div \varphi$	Completely removes the sentence ϖ from W
$W * \varphi$	Completely removes $\neg\varphi$ from W and adds φ
$D \div \delta$	Removes the default rule δ from D
$D + \delta$	Adds the default rule δ to D
$T +_1 \varphi$	Ensures that there is at least one extension of T which contains φ

By using the translation of ordered default logic into standard default logic, we can apply the belief revision operators introduced here to our framework. When updating the default rules in D, there also needs to be corresponding update which adjusts the ordering $<$ accordingly. In the following we consider these changes to be included in the updates.

3. Using Ordered Default Logic for User Modeling

We propose to use ordered default logic as described in Section 2, to reason about the agent's user model and determine which advice the agent should present to the user. The input of the agent's reasoning process is a theory of ordered default logic $T = (W, D, <)$ which describes the user model and the knowledge of the world, the output is a set of extensions E of this theory which determine the advice that the agent presents to the user.

The user model of our framework is based on the preference-based reasoning for BDI agent systems introduced in [14]. Specifically, the user model will include the user's goals, the possible actions that can be taken to achieve these goals, behavior patterns and the preferences that the user has regarding these. Additionally, we include knowledge and beliefs about the world, which allow us to reason about the context that the other concept can appear in. In the following, we detail how each of these concepts can be represented in a theory of ordered default logic.

Example 2 *In Table 1 we show how the agent from Example 1 can be represented using our framework. We only look at the schedule for one day, and only differentiate between morning and afternoon, abbreviated as Morn and Aftn respectively. We use the predicates $Plan(t, a)$ to express that an action a is scheduled at time t and $Friend(t)$ to express that a friend is available at time t. Together, this information in Table 1 forms the theory $T = (W, D, <)$ with the extensions E_1, E_2 and E_3. For readability we have only listed the sentences which are relevant for the agent's advice.*

Knowledge and Beliefs about the World The knowledge of the agent is represented as statements in W and includes the axioms and definitions that are needed to express the user model as well as information about the context. We may also require certain safety requirements to be included in the agent's knowledge base. All the information in W will be included in every extension of T, so all advice the agent can give will be consistent with this. We can include beliefs b about the world by introducing a default rule $\frac{c\,:\,b}{b}$, where c describes the context that this belief is valid in.

Concept	Example	Formalization	In
Knowledge of the World	Only one action can be scheduled at the same time	$\mathrm{Plan}(\mathrm{t},\mathrm{a}_1) \to \neg\mathrm{Plan}(\mathrm{t},\mathrm{a}_2)$	W
	Each action can only be scheduled once a day	$\mathrm{Plan}(\mathrm{t}_1,\mathrm{a}) \to \neg\mathrm{Plan}(\mathrm{t}_2,\mathrm{a})$	W
Beliefs about the World	Unless stated otherwise, we assume the friend does not have time	$\frac{:\neg\mathrm{Friend}(t)}{\neg\mathrm{Friend}(t)}$	D
Possible Actions	The user can use the morning to jog, go to the gym or read and use the afternoon to jog, read or get coffee.	$\mathrm{Plan}(\mathrm{Morn},\mathrm{Jog})$, $\mathrm{Plan}(\mathrm{Morn},\mathrm{Gym})$, $\mathrm{Plan}(\mathrm{Morn},\mathrm{Read})$, $\mathrm{Plan}(\mathrm{Aftn},\mathrm{Jog})$, $\mathrm{Plan}(\mathrm{Aftn},\mathrm{Read})$, $\mathrm{Plan}(\mathrm{Aftn},\mathrm{Coffee})$	
Behavior patterns	Jogging and reading are possible in any context, going to the gym is only considered in the morning	$\frac{:\mathrm{Plan}(\mathrm{Morn},\mathrm{Jog})}{\mathrm{Plan}(\mathrm{Morn},\mathrm{Jog})}\delta_1$ $\frac{:\mathrm{Plan}(\mathrm{Morn},\mathrm{Gym})}{\mathrm{Plan}(\mathrm{Morn},\mathrm{Gym})}\delta_2$ $\frac{:\mathrm{Plan}(\mathrm{Morn},\mathrm{Read})}{\mathrm{Plan}(\mathrm{Morn},\mathrm{Read})}\delta_3$ $\frac{:\mathrm{Plan}(\mathrm{Aftn},\mathrm{Jog})}{\mathrm{Plan}(\mathrm{Aftn},\mathrm{Jog})}\delta_4$ $\frac{:\mathrm{Plan}(\mathrm{Aftn},\mathrm{Read})}{\mathrm{Plan}(\mathrm{Aftn},\mathrm{Read})}\delta_5$	D
	Going for a coffee with a friend is only possible if the friend is available	$\frac{\mathrm{Friend}(\mathrm{Aftn})\ :\mathrm{Plan}(\mathrm{Aftn},\mathrm{Coffee})}{\mathrm{Plan}(\mathrm{Aftn},\mathrm{Coffee})}\delta_6$	D
Preferences	In the morning jogging is preferred over the gym	$\delta_2 < \delta_1$	$<$
	In the afternoon coffee with a friend is preferred over reading	$\delta_5 < \delta_6$	$<$
Goals	Exercising once a day	ExerciseOnce	W
	The goal requires at least one type of exercise	$\neg(\mathrm{Plan}(\mathrm{Morn},\mathrm{Jog}) \lor \mathrm{Plan}(\mathrm{Morn},\mathrm{Gym}) \lor \mathrm{Plan}(\mathrm{Aftn},\mathrm{Jog})) \to \neg\mathrm{ExerciseOnce}$	W
Advice	Possible Schedules	$E_1 = \{\mathrm{Plan}(\mathrm{Morn},\mathrm{Jog}),\mathrm{Plan}(\mathrm{Aftn},\mathrm{Read})\}$ $E_2 = \{\mathrm{Plan}(\mathrm{Morn},\mathrm{Gym}),\mathrm{Plan}(\mathrm{Aftn},\mathrm{Jog})\}$ $E_3 = \{\mathrm{Plan}(\mathrm{Morn},\mathrm{Read}),\mathrm{Plan}(\mathrm{Aftn},\mathrm{Jog})\}$	

Table 1. Example of a User Model in Ordered Default Logic

Possible Actions The possible actions of the user need to be expressible as a statement in the language of the knowledge base W. The actions which are contained in the extension E will constitute the advice of the agent.

Behavior Patterns Each behavior pattern is a combination of a context and the action that is taken in this context. It will generally not be possible for the user to follow all of these behaviors simultaneously. Instead, we regard the set of patterns to be a contextualized collection of the user's possible actions. Each behavior pattern is formalized as a default rule $\frac{c\,:\,a}{a}$ which contains a description of the context C as the prerequisite and the action a as the consequence and justification.

Preferences In our user model we consider the user's preferences on behavior patterns. These preferences are represented using the ordering $<$ on the default rules. As noted in Section 2, the ordering $<$ must fulfill certain requirements to ensure that we can find a consistent extension of the initial theory. This will be an important challenge when populating the user model, but is out of the scope of this paper.

Goals We take goals to be concrete, desirable and collectively achievable outcomes that the user intends to work towards and we require the agents advice to lead to these goals being reached. We include each goal in the knowledge base W of our theory $T = (W, D, <)$ as a sentence g. This means that every extension of this theory must contain, and be consistent with the assertion that the goal has been reached. As every goal is reachable, there are formulas $p_1, \dots, p_n$ which represent the possible plans to achieve the goal. Each plan is a conjunction of actions which results in the goal g being achieved. We include a statement $\neg(p_1 \vee \cdots \vee p_n) \rightarrow \neg g$ in W which infers that the goal is not reached if none of the corresponding plans have been executed in an extension. If this occurs, the extension is inconsistent and will not be considered when providing advice to the user.

Advice The agent gives the user advice of which actions to perform. These suggestions are based on the action sentences which are contained in an extension E of the theory T. If there are multiple consistent extensions, the agents needs a way to choose from these.

4. Human-Agent Realignment via Updates

Our motivation for using default logic to represent the agent's user model was that this allows the user to interact with and adapt the agents reasoning process directly if the agent's advice does not match the needs of the user. In the following, we refer to these situations as misalignment scenarios.

The three causes for these misalignments that are differentiated in [8] are the reasoning process of the agent being wrong, the agent's user model being wrong, or something having changed in such a way that the agent needs to adapt to the new situation. The last case is further divided according to the concepts that could change, namely the context the user is in, the user's internal state, and the user's desired behavior. An overview of the different misalignment scenarios, including an example of what this scenario could be for the example agent we have introduced in Table 1, can be found in Table 2.

For the purpose of this work, we assume that the agent can accurately determine which misalignment scenario is causing the mismatch between between the agent's support and the user's expectations by communicating with the user. An example of how to design such an interaction between the agent and the user can be found in [8].

In order to resolve each of the identified misalignment scenarios, we now give explicit realignment updates. An overview of these updates is also included in Table 2.

Example 3 *In Table 3 we present which realignment updates correspond to the misalignment scenarios from Table 2 and the result of this update.*

Cause	#	Scenario	Example	Realignment Update
Incorrect World Model	1	Incorrect Knowledge	An action can be scheduled multiple times a day	$W \div \varphi$ or $W * \varphi$
	2	Incorrect Beliefs	Going to the gym is not possible	$D \div \delta$ or $D + \delta$
Incorrect User Model	3	Incorrect Goals	The user plans to read every Saturday	$W \div \{g, P\}$ or $W * \{g, P\}$
	4	Incorrect Preferences	In the morning the user prefers going the gym rather than jogging	$< \div \delta < \delta'$ or $W * \delta < \delta'$
Change in Context	5	Incorrect Context	A friend is available for coffee	$W * c$
	6	New Context	The user does not want to go jogging when it is raining	$W * \{c,\ D\} + \frac{c\,:\,\varphi}{\varphi}$
Change in the User's Internal State	7	Certain Change	The user is too sick to exercise	$W * \varphi$
	8	Add Possibility	The user wants to know whether jogging can be avoided	$T +_1 \varphi$
Change in the User's Desired Behavior	9	Change of Goals	The user plans to read every Saturday	$W \div \varphi$ / $W * \varphi$
	10	Incorrect Preferences	In the morning the user prefers going the gym rather than jogging	$W \div \varphi$ or $W * \varphi$

Table 2. Types of Misalignment Scenarios and Corresponding Realignment Updates

Incorrect World Model We understand misalignments due to the agent's reasoning being wrong to manifest as mistakes in the knowledge and beliefs of the agent about the world. If the agent has incorrect knowledge of the world, this requires an update on the knowledge base W. The operators $\div$ and $*$ can be used on the the set W to either remove incorrect information or update new knowledge as introduced in 2.

If the beliefs in the agent's world model are incorrect, this can be resolved by updating the set of default rules D. For a wrong belief, we first need to identify the default rule δ that this belief corresponds to and then remove it using the contraction operator $\div$ on D. If a belief is missing it can be added to the world model as a default rule δ using the expansion operator $+$. As mentioned in Section 2, these updates also entail additional updates that are necessary for the ordering $<$.

Incorrect User Model Misalignments of the user model can refer to the information the agent has about the user's goals or preferences. When updating the goals, this also need to include changes to the corresponding plans $p_1, \dots, p_n$. If a goal g is removed, this results in the new knowledge base $W' = W \div \{g, P\}$, where $P = \neg(p_1 \vee \dots \vee p_n) \to \neg g$. If a new goal g' is added then we obtain the new knowledge base $W' = W * \{g, P\}$.

The user's preferences between behavior patterns are expressed in the ordering $<$. While we have not directly introduced update operators on this ordering, using the translation given in Section 2, the ordering is contained in the knowledge base W. This means

#	Update	Result
1	$W * \{\neg(\mathrm{Plan(t,a_1)} \rightarrow \neg\mathrm{Plan(t,a_2)})\}$	$E'_1 = \{\mathrm{Plan(Morn,Jog)}, \mathrm{Plan(Aftn,Jog)}\}$ $E'_2 = \{\mathrm{Plan(Morn,Jog)}, \mathrm{Plan(Aftn,Read)}\}$ $E'_3 = \{\mathrm{Plan(Morn,Read)}, \mathrm{Plan(Aftn,Jog)}\}$
2	$D \div \frac{: \mathrm{Plan(Morn,Gym)}}{\mathrm{Plan(Morn,Gym)}}$	$E'_1 = \{\mathrm{Plan(Morn,Jog)}, \mathrm{Plan(Aftn,Read)}\}$ $E'_2 = \{\mathrm{Plan(Morn,Read)}, \mathrm{Plan(Aftn,Jog)}\}$
3	$W * \{\mathrm{ReadAtLeastOnce},$ $\neg(\mathrm{Plan(Morn,Read)} \vee \mathrm{Plan(Aftn,Read)}) \rightarrow$ $\neg\,\mathrm{ReadOnce}\}$	$E'_1 = \{\mathrm{Plan(Morn,Jog)}, \mathrm{Plan(Aftn,Read)}\}$ $E'_2 = \{\mathrm{Plan(Morn,Read)}, \mathrm{Plan(Aftn,Jog)}\}$
4	$< *(\delta_1 < \delta_2)$	$E'_1 = \{\mathrm{Plan(Morn,Gym)}, \mathrm{Plan(Aftn,Read)}\}$ $E'_2 = \{\mathrm{Plan(Morn,Gym)}, \mathrm{Plan(Aftn,Jog)}\}$ $E'_3 = \{\mathrm{Plan(Morn,Read)}, \mathrm{Plan(Aftn,Jog)}\}$
5	$W * \{\mathrm{Friend(Aftn)}\}$	$E'_1 = \{\mathrm{Plan(Morn,Jog)}, \mathrm{Plan(Aftn,Coffee)}\}$ $E'_2 = \{\mathrm{Plan(Morn,Read)}, \mathrm{Plan(Aftn,Jog)}\}$
6	$W * \{Rain\}$ $D + \frac{Rain : \neg(\mathrm{Plan(t,Jog)})}{\neg(\mathrm{Plan(t,Jog)})}\delta_c$ $\delta_1 < \delta_c;\ \delta_4 < \delta_c$	$E'_1 = \{\mathrm{Plan(Morn,Gym)}, \mathrm{Plan(Aftn,Read)}\}$
7	$W * \{\neg\mathrm{Plan(Morn,Jog)} \wedge \neg\mathrm{Plan(Morn,Gym)} \wedge$ $\neg\mathrm{Plan(Aftn,Jog)}\}$	$E'_1 = \{\mathrm{Plan(Morn,Read)}\}$ $E'_2 = \{\mathrm{Plan(Aftn,Read)}\}$
8	$T +_1 \{\neg\mathrm{Plan(Morn,\ Jog)} \wedge \neg\mathrm{Plan(Aftn,Jog)}\}$	$E'_1 = \{\mathrm{Plan(Morn,Jog)}, \mathrm{Plan(Aftn,Read)}\}$ $E'_2 = \{\mathrm{Plan(Morn,Gym)}, \mathrm{Plan(Aftn,Jog)}\}$ $E'_3 = \{\mathrm{Plan(Morn,Read)}, \mathrm{Plan(Aftn,Jog)}\}$ $E'_4 = \{\mathrm{Plan(Morn,Gym)}, \mathrm{Plan(Aftn,Read)}\}$

Table 3. Realignment Updates corresponding to Examples # 1 - 8 in Table 2

that we can use the update operators defined for W to update the user's preferences. We use $< \div\delta < \delta'$ to remove and $< *\delta < \delta'$ to include a new preference.

Context Changes Context changes refer to a change of the situation which the user is in. Misalignments of this type can occur in two different ways. Firstly, it is possible that the agent is wrong about the context the user is currently in but generally knows which support the user requires in this situation. In this case the context information needs to be updated in the knowledge base W in order to fix the incorrect world model.

Secondly, the agent has no knowledge of the context c yet, therefore the context is not recognized and there is no information about how this context should be handled. We resolve this by expressing the behavior patterns which contain the new context as default rules of the form $\frac{c\,:\,\varphi}{\varphi}$ which we include in D using the expansion operator. We also need to ensure that this default rule is placed in the ordering $<$ correctly.

Internal State Changes The user's internal state includes any emotional, physical or mental factors which may lead to the user wanting different support from the agent. We distinguish between different levels of commitment that the user has towards these updates. If the user is completely certain that a specific part of the agent's advice should be changed, then the agent should be able enforce this. This can be done by updating the knowledge base W to include the sentence expressing the action, or negation of the action using the revision operator $*$. However, this update may lead to the removal of sentences which are used to express or regulate the goals of the user.

If the user would prefer different support φ but is still open to accepting the original suggestion, then this possibility should be introduced without being enforced. We can ensure that after the update there is an extension which aligns with the user's changes, but this extension may not be the optimal extension according to the user's goals and desires. This update is achieved by using the operator $+_1$ on the theory T, which preserves all previous extensions but adds at least one which contains the advice expressed in φ.

Desired Behavior Changes We do not treat this case separately but refer to the case of an incorrect user model.

5. Discussion

Our goal was to create a user model in a way which allows the user to interact with and adapt the agents reasoning process directly. We approached this by representing each part of the user model explicitly in a theory of ordered default logic and presented the different revisions that can be used for this.

The ordered default logic we have used can be translated into standard default logic, which means that existing theorem provers for default logic can be used to implement our framework. However, before this can be used in a behavior support agent there are still a number of issues to resolve. Most importantly, there needs to be a control mechanism to ensure that all changes that are made to the agent's reasoning maintain the effectivity of the agent and the safety of the user.

We have already mentioned the need for a dialogue which can be used to determine the cause of the misalignment from the perspective of the user [8]. However, for these interactions to be effective, the agent should also be able to communicate the information it has available and what its advice is based on. We therefore need to find ways to explain the agent's reasoning in ways that are understandable to the user, such as the work in [15]. This may for example include expanding the logic to keep track of the inference steps which were taken to arrive at each conclusion.

Further work can also be done in expanding the user model of the agent to include additional concepts such as temporal aspects, probabilistic aspects, values, or norms as seen for example in [16,17,18,19]. Finding ways to reduce the complexity of the framework and ensuring that it can be scaled for more complex situations will be necessary before implementing a realistic agent.

Lastly, while we have proposed this framework as an alternative to data-driven approaches for the purpose of adaptable and explainable reasoning, we do not view these methods as mutually exclusive. Data-driven approaches can be especially useful for recognizing behavior patterns and learning about the preferences of the user. Eventually we hope to combine the strengths of both approaches and find ways to include information which was obtained from data-driven approaches within our logical framework.

Acknowledgements

This research was partly funded by the Hybrid Intelligence Center, a 10-year programme funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organisation for Scientific Research, `https://hybrid-intelligence-centre.nl`, grant number 024.004.022.

References

[1] Oinas-Kukkonen H. Behavior Change Support Systems: A Research Model and Agenda. In: Ploug T, Hasle P, Oinas-Kukkonen H, editors. Persuasive Technology. Berlin, Heidelberg: Springer Berlin Heidelberg; 2010. p. 4-14.

[2] Albers N, Neerincx MA, Brinkman WP. Persuading to Prepare for Quitting Smoking with a Virtual Coach: Using States and User Characteristics to Predict Behavior. In: Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems. AAMAS '23. International Foundation for Autonomous Agents and Multiagent Systems; 2023. p. 717–726.

[3] van Riemsdijk MB, Jonker CM, Lesser V. Creating Socially Adaptive Electronic Partners: Interaction, Reasoning and Ethical Challenges. In: Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems. AAMAS '15. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems; 2015. p. 1201–1206.

[4] Akata Z, Balliet D, de Rijke M, Dignum F, Dignum V, Eiben G, et al. A Research Agenda for Hybrid Intelligence: Augmenting Human Intellect With Collaborative, Adaptive, Responsible, and Explainable Artificial Intelligence. Computer. 2020;53(8):18-28.

[5] Sundar SS. Rise of Machine Agency: A Framework for Studying the Psychology of Human–AI Interaction (HAII). Journal of Computer-Mediated Communication. 2020 01;25(1):74-88.

[6] Goldenberg D, Kofman K, Albert J, Mizrachi S, Horowitz A, Teinemaa I. Personalization in Practice: Methods and Applications. In: Proceedings of the 14th ACM International Conference on Web Search and Data Mining. WSDM '21. New York, NY, USA: Association for Computing Machinery; 2021. p. 1123–1126. Available from: https://doi.org/10.1145/3437963.3441657.

[7] De Laat PB. Algorithmic decision-making based on machine learning from big data: can transparency restore accountability? Philosophy & technology. 2018;31(4):525-41.

[8] Chen PY, Tielman M, Heylen D, Jonker C, Riemsdijk M. Acquiring Semantic Knowledge for User Model Updates via Human-Agent Alignment Dialogues: An Exploratory Focus Group Study. In: HHAI 2023: Augmenting Human Intellect - Proceedings of the 2nd International Conference on Hybrid Human-Artificial Intelligence. IOS Press; 2023. p. 93-108.

[9] Reiter R. A logic for default reasoning. Artificial Intelligence. 1980;13(1):81-132. Special Issue on Non-Monotonic Logic. Available from: https://www.sciencedirect.com/science/article/pii/0004370280900144.

[10] Delgrande JP, Schaub T. Expressing preferences in default logic. Artificial Intelligence. 2000;123(1):41-87. Available from: https://www.sciencedirect.com/science/article/pii/S0004370200000497.

[11] Papadimitriou CH, Sideri M. Default theories that always have extensions. Artificial Intelligence. 1994;69(1):347-57. Available from: https://www.sciencedirect.com/science/article/pii/0004370294900876.

[12] Alchourrón CE, Gärdenfors P, Makinson D. On the Logic of Theory Change: Partial Meet Contraction and Revision Functions. The Journal of Symbolic Logic. 1985;50(2):510-30. Available from: http://www.jstor.org/stable/2274239.

[13] Antoniou G. On the dynamics of default reasoning. International Journal of Intelligent Systems. 2002;17(12):1143-55. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/int.10065.

[14] Visser S, Thangarajah J, Harland J, Dignum F. Preference-based reasoning in BDI agent systems. Autonomous Agents and Multi-Agent Systems. 2016 mar;30(2):291–330. Available from: https://doi.org/10.1007/s10458-015-9288-2.

[15] Winikoff M, Sidorenko G, Dignum V, Dignum F. Why bad coffee? Explaining BDI agent behaviour with valuings. Artificial Intelligence. 2021;300:103554.

[16] van Riemsdijk MB, Jonker CM, Lesser V. Creating Socially Adaptive Electronic Partners: Interaction, Reasoning and Ethical Challenges. In: Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems. AAMAS '15. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems; 2015. p. 1201–1206.

[17] Pasotti P. Representing human habits: towards a habit support agent. In: COIN++@ECAI2016; 2016. p. 29-36.

[18] Kließ MS, Jonker CM, van Riemsdijk MB. A Temporal Logic for Modelling Activities of Daily Living. In: Alechina N, Nørvåg K, Penczek W, editors. 25th International Symposium on Temporal Representation and Reasoning (TIME 2018). vol. 120 of Leibniz International Proceedings in Informat-

ics (LIPIcs). Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik; 2018. p. 17:1-17:15. Available from: `http://drops.dagstuhl.de/opus/volltexte/2018/9782`.

[19] Tielman M, Jonker C, Riemsdijk M. What should I do? Deriving norms from actions,values and context. In: MRC@ IJCAI. CEUR Workshop Proceedings; Vol. 2134). CEUR-WS. http://ceur-ws.org/Vol-2134/; 2018. p. 35-40.