



HAL
open science

Enhancing human-AI collaboration in Opportunistic Software Composition (OCE)

Kévin Delcourt, Sylvie Trouilhet, Jean-Paul Arcangeli, Françoise Adreit

► **To cite this version:**

Kévin Delcourt, Sylvie Trouilhet, Jean-Paul Arcangeli, Françoise Adreit. Enhancing human-AI collaboration in Opportunistic Software Composition (OCE). IRIT/RR-2024-01-FR, IRIT : Institut de recherche en informatique de Toulouse. 2024. hal-04476871

HAL Id: hal-04476871

<https://hal.science/hal-04476871v1>

Submitted on 25 Feb 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License



Institut de Recherche
en Informatique de Toulouse



UNIVERSITÉ
TOULOUSE III
PAUL SABATIER

Enhancing human-AI collaboration in Opportunistic Software Composition (OCE)

IRIT/RR-2024-01-FR

Février 2024

*Kevin DELCOURT,
Sylvie TROUILHET,
Jean-Paul ARCANGELI,
Françoise ADREIT,*

Institut de Recherche en Informatique de Toulouse (IRIT)

Abstract

Our focus is on improving the Opportunistic Composition Engine's (OCE) human-AI interaction. To accomplish this goal, we identify design solutions and explore how they could be applied to AmI and OCE. We have organized this study using the guidelines of Amershi et al., which offer a comprehensive framework for answering all of our research questions. For each guideline, we discuss its significance and how it can enhance human interaction with an IML application like OCE.

Keywords

Human-AI Interaction, Interactive Machine Learning, Ambient Intelligence, Reinforcement Learning, Software Composition

I Introduction

Our primary objective is to enhance the human-AI interaction within the Opportunistic Composition Engine (OCE) [16]. To achieve this aim, we propose design solutions tailored to the context of OCE. Our approach is structured around the guidelines presented by Amershi et al. [2], providing a comprehensive framework to address our research questions systematically. Each guideline is examined for its significance and potential contributions to improving human-AI interaction. Table 1 synthesizes the results of our study. There, S_{i-j} represents the proposed solution numbered j for the guideline G_i .

	Human-AI guidelines	Design solutions
Initially	G1. Make clear what the system can do.	$S1-1$. Textual explanations. $S1-2$. Demonstrations.
	G2. Make clear how well the system can do what it can do.	$S2-1$. Textual explanations. $S2-2$. Demonstrations on what to do when the engine is wrong.
	GA. Make clear how the system works.	$SA-1$. Textual explanations. $SA-2$. Demonstrations.
During interaction	G3. Time services based on context.	$S3-1$. New assembly at every change in the environment. $S3-2$. Request input when the human is attentive [13].
	G4. Show contextually relevant information.	$S4-1$. Display of the available components. $S4-2$. Automatically detected human activity.
	GB. Provide interpretable output to the user.	$SB-1$. Propose alternative views for an assembly [15].
	G5. Match relevant social norms.	Not relevant for OCE.
	G6. Mitigate social biases.	Not relevant for OCE.
	When wrong	G7. Support efficient invocation.
G8. Support efficient dismissal.		$S8-1$. Ability to reject an assembly. $S8-2$. Ability to ignore an assembly. $S8-3$. Ability to undo an assembly.
G9. Support efficient correction.		$S9-1$. Ability to modify an assembly. $S9-2$. Highlights to assist correction. $S9-3$. An intuitive correction interface [7].
G10. Scope services when in doubt.		$S10-1$. Human choice between components. $S10-2$. Multiple assemblies. $S10-3$. Notification of uncertainty.
G11. Make clear why the system did what it did.		$S11-1$. Available agents' knowledge. $S11-2$. Agent's decision explanation. $S11-3$. Collective decision explanation.
G12. Remember recent interactions.		$S12-1$. Give more weight to the most recent interaction. $S12-2$. Ability to redeploy the most recent assemblies.
G13. Learn from user behavior.		$S13-1$. Decisions based on past interactions. $S13-2$. Human error rate [1]. $S13-3$. Complex interaction data [17].
Over time	G14. Update and adapt cautiously.	$S14-1$. Reinforcement learning limits disruptive changes. $S14-2$. Less disruptive exploration strategies [17].
	G15. Encourage granular feedback.	$S15-1$. Assembly level feedback. $S15-2$. Component level feedback. $S15-3$. Connection level feedback. $S15-4$. Service level feedback.
	G16. Convey the consequences of user actions.	$S16-1$. Textual notifications. $S16-2$. Pre-visualization of an action's consequences [14].
	G17. Provide global controls.	$S17-1$. Editable engine parameters. $S17-2$. Optionally editable parameters [5]. $S17-3$. Automatic parameter tuning.
	G18. Notify user about changes.	Not relevant for OCE.

Table 1: Synthesis of the proposed design solutions for each guideline for Human-AI interaction [2]. GA and GB were added for this analysis

Amershi et al.'s framework is open for extension, and we have incorporated two additional guidelines, GA and GB.

- *GA. Make clear how the system works.* This guideline emphasizes the importance of transparency regarding the AI's inner workings in order to gain trust from the human. While this

may only be relevant for experienced users in complex applications, providing a simplified explanation of the inner workings can be beneficial for the interaction. As this guideline pertains to information that should be made available to the human at the beginning of the interaction, we have placed it in the “Initially” category.

- *GB. Provide interpretable output to the user.* This guideline emphasizes the need to ensure that the AI’s output is understandable by the human. It is particularly relevant in complex use cases like Opportunistic Composition, whereas it may not be necessary in simpler scenarios such as content recommendation. This guideline pertains to information that is distilled during the interaction, and therefore we have put it in the “During Interaction” category.

II Initially

These guidelines provide recommendations for the design of the beginning of the human-AI interaction, emphasizing the importance of managing the user’s expectations and building trust in the intelligent service from the start.

G1. Make clear what the system can do. While the main purpose of an IML application, such as image classification, may be easy to understand, explaining the full range of capabilities to the user, such as data labeling or parameter tuning, can be more challenging.

This is particularly true in the case of an Aml application designed to assist humans in a pervasive environment like OCE. In this context, textual information (*S1-1*) or demonstrations (*S1-2*) may be the best options for explaining the application’s capabilities, *i.e.* building and proposing component assemblies to the human user. Over time and with use, the user should gain a better understanding of how the application works.

G2. Make clear how well the system can do what it can do .

The IML application should provide information to the human about the likelihood of approximations or mistakes occurring during the process. Managing human expectations is crucial to avoid over reliance on the AI system and prevent users from discontinuing use due to surprise or disappointment [12].

In the case of reinforcement learning applications like OCE, mistakes are an essential part of the trial and error learning process, so providing clear and understandable warnings (*S2-1*) about this can initially help. Additionally, explaining what to do when the AI system produces incorrect output (*S2-2*), as discussed in Section IV, can further assist in managing expectations and maintaining user trust. Regarding OCE, it is crucial to communicate to the human that the assemblies proposed by OCE are built from the components currently available and the system’s past compositions.

GA. Make clear how the system works. This guideline emphasizes the importance of transparency in the AI system’s inner workings, which can help build trust with human users. Especially in complex applications, users want to know how the AI arrived at a particular decision or recommendation. Providing a simplified version of the inner workings can enhance the interaction, although it may only be necessary for experienced users.

In the context of OCE, there is currently no available explanation of how the engine works. Additionally, it may be difficult to demonstrate the improvement of the learner’s knowledge over time,

as this is a gradual process that occurs over an extended period. To address this need for transparency, one potential solution is to provide information on the engine’s learning process. Textual explanations (*SA-1*) or targeted demonstrations (*SA-2*) should be considered.

III During interaction

These guidelines provide recommendations for designing the interaction itself. They primarily focus on managing the workload assigned to humans and addressing their social expectations.

G3. Time services based on context. Determining the appropriate moments to involve the human in the learning process is critical for a positive user experience. Disruptive interruptions can be detrimental to the user’s ability to complete tasks, leading them to disregard important information.

The OCE prototype currently generates a new assembly for every component that appears or disappears in the environment (*S3-1*), which aligns with this guideline. However, it may be beneficial to consider the user’s current focus and tailor the interactions accordingly (*S3-2*), as suggested by [13]. Nonetheless, detecting a user’s focus in an open Aml context can be more challenging than in a traditional desktop application.

G4. Show contextually relevant information. IML applications should display information relevant to the human’s current task or environment, to avoid overwhelming them with unnecessary information.

In the case of OCE, available components in the human’s environment are considered part of their context and are relevant to display (*S4-1*). However, taking into account additional information about the human’s context, such as their location or activity (*S4-2*), could further improve the relevance of the proposed assemblies. For example, if the human is in a classroom setting, it would be inappropriate to suggest applications for their favorite streaming service. However, automatically detecting the human’s context poses a significant challenge, especially for a generic application like OCE. Moreover, it should be noted that, fundamentally, OCE’s learning does not rely on learning patterns of behavior [4], activity detection or, more broadly, context identification.

GB. Provide interpretable output to the user. This recommendation is not included in the guidelines of Amershi et al. and the ability for users to intuitively understand the output of AI systems is not addressed by any other guideline. We believe this recommendation is crucial for Aml applications like OCE because such an understanding is essential for effective interaction with the AI system.

OCE’s first available view is a component diagram, which may not be suitable for end-users which are not trained in programming. In [15], authors proposed to address this issue by using model transformation to provide alternative views that fit users’ comprehension capabilities (*SB-1*), as shown in Figure 1.

G5. Match relevant social norms. The interaction between the AI and human should be sensitive to the human’s social and cultural context to manage their expectations. For example, a voice assistant should address the human in a semi-formal manner, as it’s commonly expected in professional settings [2].

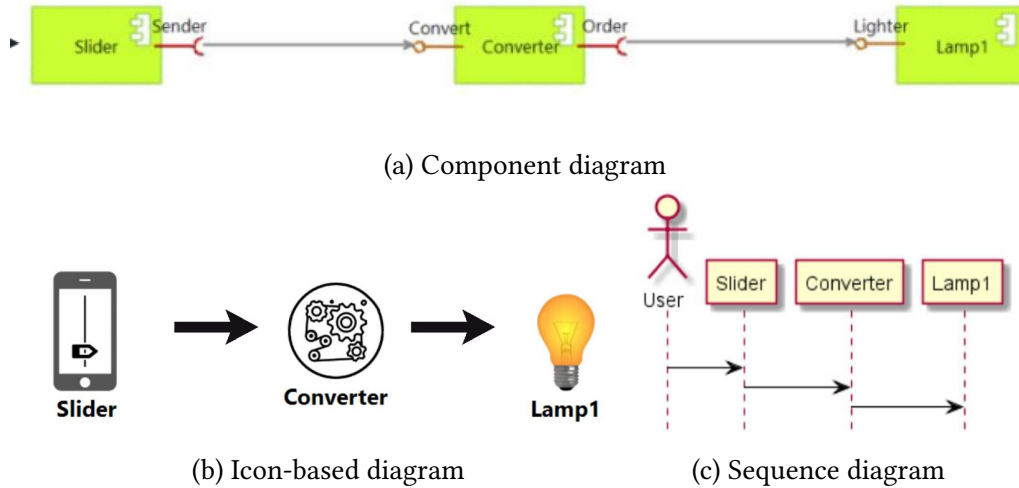


Figure 1: Possible views for an assembly in OCE

However, in the case of OCE and ambient applications, it’s challenging to determine which social norms should be followed or avoided, as these applications may target a broad audience with diverse cultural backgrounds. At the current time, this guideline is not a priority to Opportunistic Composition.

G6. Mitigate social biases. The AI system should be free of any undesirable stereotypes or biases, which can arise during training and manifest in the model, particularly in applications related to race or gender [9].

At present, OCE does not collect user data to identify common profiles, so there are no unfair biases of this nature that could emerge. Learning is only based on the available components in the human’s environment and the human’s actions within the application. While we acknowledge that biases and stereotypes can be present in any AI system, this particular guideline is not applicable to OCE as it stands. In the future, mitigating biases may become a concern if techniques such as learning by transfer, as seen in Teachable Machine [5], are implemented.

IV When wrong

The following guidelines provide support for managing mistakes made by the learning service during human-AI interaction, when the output does not meet the expectations of the human. In most cases, the learning service tries to tune its service according to the preferences of the human, and as such, it needs to make mistakes to gain knowledge. Therefore, it is important to help the human recover from these mistakes to avoid disrupting their user experience.

G7. Support efficient invocation. The application should allow the human to easily request the AI’s service. This means that if the output from the learner is not satisfactory, the human should be able to request new output quickly and easily.

In the current OCE workflow, a change in the environment triggers a new assembly to be pushed to the human, so the human doesn’t explicitly request the AI’s services. But the possibility for the human to trigger new assemblies at their discretion (S7-1) should be considered. Additionally, it would be possible for the human to express more precise needs in these composition requests, such as specifying components that should or should not be included in the final result (S7-2).

Alternatively, providing a blank editor would allow the human to freely edit component assemblies (S7-3). This would help them gain a better understanding of the way composite applications are built and provide an opportunity to gather data on the components they prefer or dislike, allowing to fine-tune the future propositions.

G8. Support efficient dismissal. The AI service should be easily dismissible or ignorable by the human, particularly when it produces incorrect results.

Currently, the OCE workflow allows the human to reject an assembly proposition from the composition engine (S8-1), which satisfies this guideline. However, further user testing is necessary to determine the effectiveness of this interaction and explore potential alternatives. In addition to rejection, we could also consider the option for the human to ignore OCE's assembly propositions (S8-2), which would be a lighter form of dismissal useful in situations where the human doesn't have time to review the proposed assembly. This action should have little to no impact on learning.

Another possible way for the human to dismiss OCE's proposition is to undo and rollback to a previous assembly (S8-3). Compared to basic rejection, this action would be a higher form of dismissal, and should send more negative feedback to OCE's agents. However, this may not always be possible in dynamic environments where components needed to deploy the previous assembly may no longer be available.

G9. Support efficient correction. Ensuring that the human can easily correct the AI when it is wrong is crucial for the success of the interaction.

In OCE, the correction process is already integrated, as the human can edit the links between components to modify an assembly (S9-1). However, this process requires a higher understanding of software composition and could be further improved to alleviate the workload on the human. For instance, we could consider highlighting alternative connections when the human wants to change the connection between two services (S9-2).

To explore alternative correction processes, we could look at existing solutions such as Crayons [7], which allows users to correct AI outputs by simply coloring over them (S9-3). Additional testing would be necessary to determine which solution is the most efficient and effective for the OCE workflow.

Guidelines G7, G8 and G9 propose to give the human the ability to choose between dismissing, ignoring, correcting, or reinvoking the AI service when it produces incorrect results. This flexibility allows the human to tailor their response to the severity of the mistake, which ultimately leads to a better user experience in the human-AI collaboration. However, it is important to consider the potential for placing an additional burden on the human when applying these guidelines.

G10. Scope services when in doubt. When uncertain about a human's goals or preferences, the AI system should engage in disambiguation to ask for clarification. This not only acknowledges the human's input, but also updates the system's assumptions on the human's preferences.

In the current OCE workflow, the system does not adapt to the level of uncertainty in the human's preferences in the current environment. As a result, when the engine does not know what would be the best service for the human, it chooses at random.

To improve this, a recommendation is to adjust the system's behavior based on the level of uncertainty. As an example, in situations where OCE is most uncertain and cannot differentiate between components, it could prompt the human to provide input by presenting them with a choice between

the options (*S10-1*) [8, 6]. Alternatively, the system could propose multiple assemblies to explore different possibilities more quickly (*S10-2*). Ultimately, a minimal way to implement this guideline would be to at least inform the human of the uncertainty (*S10-3*), so that they can raise their levels of awareness towards any potential mistake from the engine.

To implement this, the uncertainty of the engine needs to be formalized, which can be challenging given OCE decentralized decision-making process. Managing a large number of large assemblies could also pose scaling problems for the human. Finally, this places an additional burden on the human user.

G11. Make clear why the system did what it did. Providing an explanation of AI’s behavior to the human is crucial in building trust and understanding of the AI application. However, creating explainable AIs is a current challenge for the scientific community [3].

The current OCE interface does provide access to the agent’s knowledge (*S11-1*), which can be considered an element of explanation for the agent’s decision-making process, as it is the state of its learning process. However, the current interface is not suitable for the general public and lacks information on the actual decision process, which is what the agent did with its knowledge. To meet this guideline, we need to work towards providing an easily understandable explanation of an agent’s decision to the human (*S11-2*). Additionally, by aggregating these individual explanations, we could create an emergent explanation of the application (*S11-3*) [10].

V Over time

These guidelines provide recommendations for how to improve the interaction over time, based on feedback and usage data. This can lead to increased user satisfaction, better performance, and better trust in the system over time. Additionally, it can help to uncover potential issues or areas for improvement that may not have been apparent during the initial design and development phases.

G12. Remember recent interactions. The AI system should keep a memory of past interactions with the human to enhance efficiency.

OCE already takes into account the memory of past interactions and decisions made by the human, and gives a heavier weight to the most recent interaction (*S12-1*). Another way to see and implement this guideline would be to keep a memory of the last assemblies and allow the user to redeploy them if they are still available (*S12-2*). This could provide additional valuable feedback to the engine on the human’s preferences and help adapt to the changing environment.

G13. Learn from user behavior. The learner should personalize the human’s experience by learning from their actions, which is typically the main goal of an IML system.

In OCE, decisions are based on the past behavior of the human, such as accepting, rejecting, or modifying propositions of assemblies (*S13-1*). However, there are more sophisticated approaches proposed in the literature to better implement this guideline. For instance, in [1], authors suggest estimating the human’s error rate to improve recommendations (*S13-2*), while Zheng et al. [17] consider the time to return to the service as a potential reward, where a faster return is considered better (*S13-3*). Implicit interaction data such as eye tracking, mouse movement, and timing of certain events can also be used to personalize the experience (*S13-4*). It is also possible to gather this information without informing the human, as in [11], to avoid overloading them with information.

G14. Update and adapt cautiously. An AI system should limit disruptive changes when updating and adapting the system’s behavior, in order not to shock and overload the human.

In OCE, this guideline is implemented through positive feedback given to assembly propositions that are accepted by the user, which encourages the engine to stick to those assembly plans over time and limit extreme changes in the propositions (*S14-1*). However, the dynamics of the environment may introduce changes in the available components, which is beyond the engine’s control. Moreover, in a reinforcement learning setting, some exploration is necessary, which can introduce changes in the system’s behavior. Nonetheless, since OCE is a multi-agent system, the exploration/exploitation dilemma is managed at the agent’s level, and the probability of all agents exploring simultaneously is very low, so that the proposed assemblies would not be overly disruptive. To further limit the risk of disruptive changes, it may be possible to work on alternative exploration strategies (*S14-2*), such as the one proposed in [17].

G15. Encourage granular feedback. The human should be able to provide feedback at different levels of their interaction with the AI system.

The feedback in OCE is currently limited to the acceptance or rejection of proposed assemblies and editing of specific connections during modifications. To enrich the interaction, it would be interesting to allow the human to provide explicit feedback at all the other levels of interaction shown in Figure 2, such as the assembly level (*S15-1*), component level (*S15-2*), connection level (*S15-3*), and service level (*S15-4*).

- At the assembly level, the human could provide explicit feedback on whether they like or dislike an assembly, which would be an alternative feedback to the simple acceptance or rejection of an assembly.
- At the component level, the human could provide feedback on which components they prefer to see in future assemblies.
- At the connection level, they could communicate their desire for certain connections, *i.e.*, component associations, to be prioritized or de-prioritized in future assemblies.
- At the service level, they could provide input on the services they want to see more or less utilized in future assemblies.

All those changes may induce changes in OCE’s behavior, since it is necessary to take into account those new kinds of feedback. All this range of feedback may allow the human to fine-tune the feedback they give to the engine, leading to better adaptation to their preferences and ultimately better propositions.

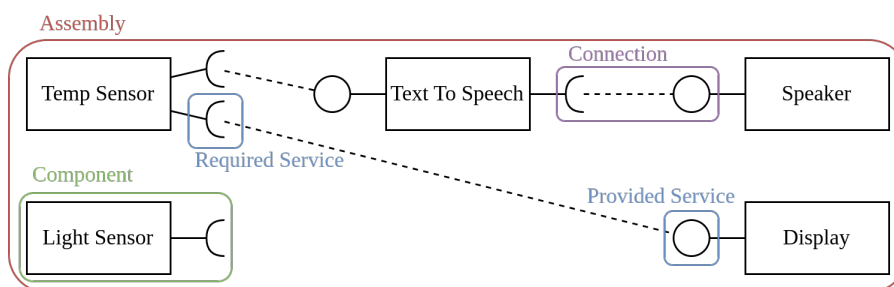


Figure 2: Component diagram with assembly, component, service, and connection highlighted

G16. Convey the consequences of user actions. It's important for the human to be aware of how their actions can impact the future behavior of the AI system.

However, in the case of OCE, the human is not explicitly informed of this impact. As the proposed assemblies improve over time, the human may eventually notice how the system has evolved, but this may take some time to realize. Textual information like "you will see more/less similar applications in the future" could be an easy way to implement this guideline (*S16-1*). However, this may place a heavier cognitive load on the human, as they need to make choices based on their preferences (e.g. "do I want this application now") and the impact of their choice on the system (e.g. "do I want to see more of this application in the future").

Implementing more advanced propositions, such as the pre-visualization of future recommendations after a human action (*S16-2*), as suggested in [14], may be too difficult to achieve in a dynamic MAS setting where knowledge is distributed. In this context, it is also possible that some human users may not have the capacity to understand such complex information.

G17. Provide global controls. The human should be able to customize the overall functioning of the AI system based on their individual preferences.

The OCE system currently allows for this through editable engine parameters in the interface (*S17-1*). However, these parameters may not be understandable for end-users without ML expertise, making the system difficult to use. To address this, adjustments can be made to the interface, such as hiding the options to edit the engine's global parameters initially (*S17-2*), to accommodate less experienced users [5]. Furthermore, it is worth considering the concept of automatic parameter tuning (*S17-3*), which could reduce the workload on the human user.

G18. Notify user about changes. The AI system should inform the user when updates or new features are added, and provide guidance to existing users during the transition.

While this guideline does not currently apply to OCE since it is still a research prototype, it is an important consideration for future implementation. By keeping users informed about updates and changes to the system's capabilities, users can feel more confident in their use of the system and understand how to take advantage of any new features or improvements.

VI Conclusion

We presented several design solutions taken from our analysis and from the IML literature that could help OCE adhere to the guidelines for human-AI interactions. These elements are summarized in Table 1.

As it stands, OCE does not comply with most of these guidelines, and several upgrades are needed to improve human-machine interaction. The next step in our approach is to prioritize those solutions and implement them into OCE.

Bibliography

- [1] Riad Akrou, Marc Schoenauer, Michèle Sebag, and Jean-Christophe Souplet. Programming by feedback. In *Int. Conf. on Machine Learning*, volume 32, pages 1503–1511. JMLR. org, 2014.
- [2] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N. Bennett, and Kori Inkpen. Guidelines for human-AI interaction. In *Proc. of the 2019 CHI conf. on human factors in computing systems*, pages 1–13, 2019.
- [3] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115, 2020.
- [4] Asier Aztiria, Alberto Izaguirre, and Juan Carlos Augusto. Learning patterns in ambient intelligence environments: a survey. *Artificial Intelligence Review*, 34:35–51, 2010.
- [5] Michelle Carney, Barron Webster, Irene Alvarado, Kyle Phillips, Noura Howell, Jordan Griffith, Jonas Jongejan, Amit Pitaru, and Alexander Chen. Teachable machine: Approachable Web-based tool for exploring machine learning classification. In *Extended abstracts of the 2020 CHI Conf. on human factors in computing systems*, pages 1–8, 2020.
- [6] Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences, 2017.
- [7] Jerry Alan Fails and Dan R. Olsen Jr. Interactive machine learning. In *Proceedings of the 8th Int. Conf. on Intelligent user interfaces*, pages 39–45, 2003.
- [8] Simon Flutura, Andreas Seiderer, Ilhan Aslan, Chi-Tai Dang, Raphael Schwarz, Dominik Schiller, and Elisabeth André. Drinkwatch: A mobile wellbeing application based on interactive and cooperative machine learning. In *Proceedings of the 2018 Int. Conf. on Digital Health*, pages 65–74, 2018.
- [9] Timnit Gebru. Oxford handbook on AI ethics, book chapter on race and gender. arXiv preprint arXiv:1908.06165, 2019.
- [10] Alexandre Heuillet, Fabien Couthouis, and Natalia Díaz-Rodríguez. Collective explainable AI: Explaining cooperative strategies and agent contribution in multiagent reinforcement learning with shapley values. *IEEE Computational Intelligence Magazine*, 17(1):59–71, 2022.
- [11] Andreas Holzinger, Markus Plass, Michael Kickmeier-Rust, Katharina Holzinger, Gloria Cerasela Crişan, Camelia-M. Pinte, and Vasile Palade. Interactive machine learning: experimental evidence for the human in the algorithmic loop. *Applied Intelligence*, 49(7):2401–2414, 2019.

- [12] Donald Honeycutt, Mahsan Nourani, and Eric Ragan. Soliciting human-in-the-loop user feedback for interactive machine learning reduces user trust and impressions of model accuracy. In *Proceedings of the AAAI Conf. on Human Computation and Crowdsourcing*, volume 8, pages 63–72, 2020.
- [13] Taylor A. Kessler Faulkner and Andrea Thomaz. Interactive Reinforcement Learning from Imperfect Teachers. In *Companion of the 2021 ACM/IEEE Int. Conf. on Human-Robot Interaction*, pages 577–579, 2021.
- [14] Tobias Schnabel, Saleema Amershi, Paul N. Bennett, Peter Bailey, and Thorsten Joachims. The Impact of More Transparent Interfaces on Behavior in Personalized Recommendation. In *Proceedings of the 43rd Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 991–1000, 2020.
- [15] Sylvie Trouilhet, Jean-Paul Arcangeli, Jean-Michel Bruel, and Maroun Koussaifi. Model-driven engineering for end-users in the loop in smart ambient systems. *Journal of Universal Computer Science*, 27(7):755–773, 2021.
- [16] Walid Younes, Sylvie Trouilhet, Françoise Adreit, and Jean-Paul Arcangeli. Agent-mediated application emergence through reinforcement learning from user feedback. In *29th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises*, 2020.
- [17] Guanjie Zheng, Fuzheng Zhang, Zihan Zheng, Yang Xiang, Nicholas Jing Yuan, Xing Xie, and Zhenhui Li. Drn: A deep reinforcement learning framework for news recommendation. In *Proceedings of the 2018 world wide web conference*, pages 167–176, 2018.