Parametric User Abstractions for Controllable and Adaptive Explanation in Human–Robot Interaction

Amar Halilovic[1,*], Vahidin Hasic[2] and Senka Krivic[2]

[1]Institute of Artificial Intelligence, Ulm University, James-Franck-Ring, 89081 Ulm, Germany
[2]Faculty of Electrical Engineering, University of Sarajevo, Zmaja od Bosne bb., 71000 Sarajevo, Bosnia and Herzegovina

Abstract

Robots interacting with humans should adapt explanations to diverse user preferences. Many explanation approaches rely on fixed strategies or treat adaptation as part of a broader decision policy, which can make user modeling difficult to inspect. We propose *parametric user abstraction*, a structured representation in which each user is modeled through latent parameters encoding explanation preferences, context sensitivity, and dependencies between explanation attributes. These parameters serve as an interpretable control interface for selecting and constraining explanation generation. We ground the approach in a robot librarian scenario, where a robot assists visitors while encountering interruptions, delays, and failures that call for adaptive explanations.

Keywords

Human–Robot Interaction, Explainable AI, Personalization, Adaptive Robot Explanations

1. Introduction

Robots operating in human-centered environments are increasingly expected not only to act competently but also to explain their behavior to support human understanding. Prior work in Explainable Artificial Intelligence (XAI) has emphasized that explanations are social and user-dependent, rather than merely technical outputs of a decision system [1]. In Human–Robot Interaction (HRI), explanations have similarly been studied as a mechanism for making robot behavior more understandable and transparent [2]. However, explanation preferences differ substantially across users. Some users may prefer concise progress updates, whereas others may prefer detailed justifications. This motivates a user-modeling problem: the robot should not only determine what happened in the environment, but also how this information should be communicated to a particular user. This perspective connects robot explanation to broader work on adaptive systems and player/user modeling, where compact representations of user characteristics are used to guide system behavior [3, 4].

We build on our prior work on robot navigation explanations, including multimodal, affordance-based, and preference-aware explanation methods [5, 6]. Here, we argue that the robot should maintain a compact abstraction of user explanation preferences and use it to condition explanation generation. We call this perspective *parametric user abstraction*. We treat the user model itself as the main technical object and generalize it into a structured abstraction with attribute-level, context-sensitive, and attribute-coupling parameters. Each user is represented by a low-dimensional latent parameter vector that encodes preferences over different explanation attributes. The robot estimates and updates these parameters online, while an explanation generator uses them as a control interface.

Consider a robot librarian assisting visitors in an indoor library. A visitor requests a book, and the robot navigates toward the corresponding shelf to retrieve it. During navigation, the robot may encounter unexpected obstacles, such as a chair left in the aisle or a nearby human occupying the space needed for passage. Such events can cause the robot to stop, replan, deviate from its expected path, or ask for assistance. These situations naturally create explanation opportunities.

Joint Proceedings of the ACM UMAP Workshops 2026, UMAP 2026, June 8–11, 2026, Gothenburg, Sweden
*Corresponding author.
amar.halilovic@uni-ulm.de (A. Halilovic); vahidin.hasic@etf.unsa.ba (V. Hasic); senka.krivic@etf.unsa.ba (S. Krivic)
0000-0002-2354-986X (A. Halilovic); 0009-0003-6306-8037 (V. Hasic); 0000-0001-8045-427X (S. Krivic)

2. Parametric User Abstraction

We represent each user h through a latent parameter vector

$$\theta_h = \{\theta_{h,f} \mid f \in \mathcal{F}\}, \tag{1}$$

where $\mathcal{F}$ denotes a set of explanation attribute families (e.g., modality (textual vs. multimodal) or initiative level (explanation only on request vs. proactive explanation)). Rather than treating θ_h only as a flat set of independent preferences, we view it as a structured abstraction composed of three parameter groups:

$$\theta_h = (\theta_h^{\text{attr}}, \theta_h^{\text{context}}, \theta_h^{\text{coupling}}). \tag{2}$$

The first group, θ_h^{attr}, captures preferences over explanation attributes. The second group, $\theta_h^{\text{context}}$, captures how these preferences vary across explanation-triggering contexts (e.g., delays, deviations, failures). The third group, $\theta_h^{\text{coupling}}$, captures dependencies between explanation attributes, for example, whether a user prefers detailed explanations only when they are accompanied by a visual overlay, or whether proactive explanations are acceptable only in high-urgency situations.

Given an interaction state s, the robot generates an explanation as

$$x = g(c(s), \phi(s), \theta_h), \tag{3}$$

where $c(s)$ denotes the explanation-triggering context extracted from the interaction state and $\phi(s)$ denotes the explanation content derived from the robot's task and environment model. The generator g therefore separates three roles: identifying what happened, selecting what content should be communicated, and adapting how that content is expressed for the user. This formulation separates *context understanding* from *preference adaptation*: the robot first reasons about what is happening, and then decides how to communicate it based on the user abstraction.

To enable learning under uncertainty, we model each parameter as a latent random variable. For binary explanation choices, a simple and interpretable formulation is

$$\theta_{h,f} \sim \text{Beta}(\alpha_{h,f}, \beta_{h,f}), \tag{4}$$

with posterior mean

$$\mathbb{E}[\theta_{h,f}] = \frac{\alpha_{h,f}}{\alpha_{h,f} + \beta_{h,f}}. \tag{5}$$

This allows the robot to maintain explicit uncertainty over user preferences and to adapt rapidly from sparse feedback.

After generating an explanation, the robot observes feedback $y \in \{0, 1\}$ and updates its belief:

$$\alpha_{h,f} \leftarrow \alpha_{h,f} + y, \qquad \beta_{h,f} \leftarrow \beta_{h,f} + (1 - y). \tag{6}$$

Controlling the Abstraction A benefit of parameterizing the user model explicitly is that adaptation can be constrained. Let $\mathcal{X}(s)$ denote the set of possible explanation configurations in state s, and let $A(x)$ denote the attribute vector of explanation x. The structured user abstraction induces a score

$$S(x; s, h) = \theta_h^{\text{attr}} \cdot A(x) + \theta_h^{\text{context}} \cdot C(s, x) + \theta_h^{\text{coupling}} \cdot B(A(x)), \tag{7}$$

where $C(s, x)$ represents context–explanation compatibility and $B(A(x))$ represents interactions among explanation attributes. For example, $C(s, x)$ can encode that navigation failures may require more detailed explanations than routine progress updates, while $B(A(x))$ can encode that visual and detailed explanations may be especially useful when combined.

The robot then selects

$$x^\star = \arg \max_{x \in \mathcal{X}_{\text{safe}}(s)} S(x; s, h), \tag{8}$$

where $\mathcal{X}_{\text{safe}}(s) \subseteq \mathcal{X}(s)$ restricts explanations to those that satisfy task, safety, and social constraints. Thus, the abstraction not only predicts user preferences but also provides a controllable interface for restricting, inspecting, and selecting explanations.

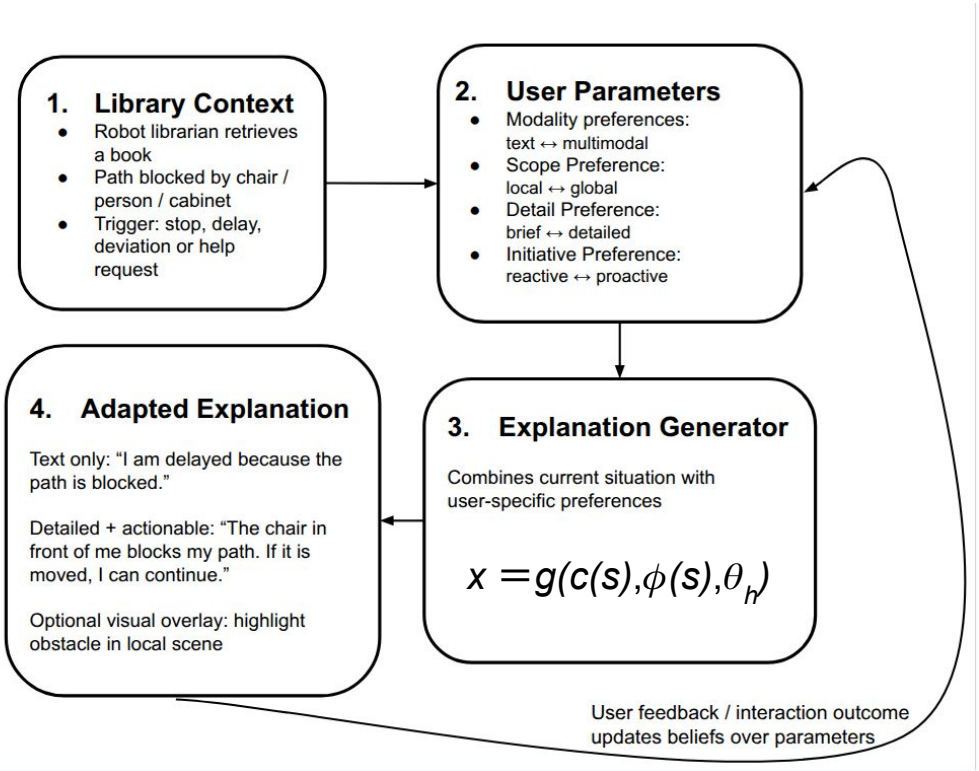

Figure 1: Parametric user abstraction in the robot librarian scenario. A library robot encounters an explanation-triggering event, such as a blocked path, a delay, or a deviation. The current interaction context is combined with a user-specific latent parameter vector encoding explanation preferences, context sensitivity, and attribute dependencies. The resulting explanation is adapted to the user and updated over time from feedback.

Scenario-Conditioned Explanation Generation In the robot librarian setting, the context s may include the robot's task state, navigation status, nearby humans, and salient environmental causes of interruption. Suppose the robot stops because a chair blocks the aisle to the requested bookshelf. The underlying explanation content may remain stable: the path is blocked by the chair. What adapts is the form of the explanation. Depending on θ_h, the system may produce a different explanation (see Figure 1). This conditioning view connects naturally to prior work on multimodal and affordance-based explanation generation [5, 6]. A blocked chair can be represented not only as an obstacle but also as an object with an actionable property, such as movability, relevant to resolving the task interruption. Parametric user abstraction provides a personalization layer over explanation representations and attributes. Algorithm 1 summarizes the adaptation loop.

Algorithm 1 Structured Parametric Explanation Adaptation

Initialize θ_h^{attr}, $\theta_h^{\text{context}}$, and $\theta_h^{\text{coupling}}$
for each interaction step **do**
  Observe interaction state s
  Extract explanation context $c(s)$ and candidate content $\phi(s)$
  Construct candidate explanations $\mathcal{X}(s)$
  Restrict candidates to admissible explanations $\mathcal{X}_{\text{safe}}(s)$
  Select $x^\star = \arg\max_{x \in \mathcal{X}_{\text{safe}}(s)} S(x; s, h)$
  Present explanation $x^\star$
  Observe user feedback y
  Update the relevant components of θ_h
end for

3. Discussion and Future Directions

Parametric user abstraction has three main advantages. First, it is interpretable: user adaptation is represented through explicit preference parameters rather than only through opaque internal policy states. Second, it is data-efficient: low-dimensional parameter updates can be learned from sparse interaction feedback. Third, it is modular: the abstraction can be placed on top of existing planning-based, affordance-based, verbal, or multimodal explanation generators. The structured formulation strengthens these advantages by making explicit which aspects of the user model are being adapted. Attribute-level parameters capture general preferences, context-sensitive parameters capture how those preferences change across explanation-triggering situations, and coupling parameters capture dependencies between explanation attributes. At the same time, this abstraction is intentionally limited and is conceptual. A compact parameter vector cannot capture all aspects of human communication preferences, which may depend on different factors. The framework should therefore be understood as an intermediate control layer. This limitation is also useful: by constraining personalization to a small set of explanation attributes, the robot's adaptation remains easier to inspect and constrain.

Future work should investigate which explanation attributes are most important to parameterize in repeated HRI, how explicit user parameters can be combined with contextual factors, and how parameter updates should be evaluated in real HRI studies.

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT and Grammarly for language proof-reading and writing assistance. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

Acknowledgment

This work was supported by the Federal Ministry of Education and Science of the Federation of Bosnia and Herzegovina (FMON) through the 2025 scientific project funding program (Official Gazette of FBiH, Nos. 7/25, 31/25, 42/25, 90/25).

References

[1] T. Miller, Explanation in artificial intelligence: Insights from the social sciences, Artificial Intelligence 267 (2019) 1–38. URL: https://www.sciencedirect.com/science/article/pii/S0004370218305988. doi:https://doi.org/10.1016/j.artint.2018.07.007.

[2] Z. Han, E. Phillips, H. A. Yanco, The need for verbal robot explanations and how people would like a robot to explain itself, J. Hum.-Robot Interact. 10 (2021). URL: https://doi.org/10.1145/3469652. doi:10.1145/3469652.

[3] P. Brusilovsky, E. Millán, User models for adaptive hypermedia and adaptive educational systems, in: The adaptive web: methods and strategies of web personalization, Springer, 2007, pp. 3–53.

[4] A. M. Smith, C. Lewis, K. Hullet, G. Smith, A. Sullivan, An inclusive view of player modeling, in: Proceedings of the 6th International Conference on Foundations of Digital Games, FDG '11, Association for Computing Machinery, New York, NY, USA, 2011, p. 301–303. URL: https://doi.org/10.1145/2159365.2159419. doi:10.1145/2159365.2159419.

[5] A. Halilovic, S. Krivic, Planning of explanations for robot navigation, in: 2024 IEEE International Conference on Robotics and Automation (ICRA), 2024, pp. 5478–5484. doi:10.1109/ICRA57147.2024.10611000.

[6] A. Halilovic, S. Krivic, Affordance-based explanations of robot navigation, in: 2025 IEEE International Conference on Robotics and Automation (ICRA), 2025, pp. 13523–13529. doi:10.1109/ICRA55743.2025.11128010.