Scaling Context-Aware Task Assistants that Learn from Demonstration and Adapt through Mixed-Initiative Dialogue

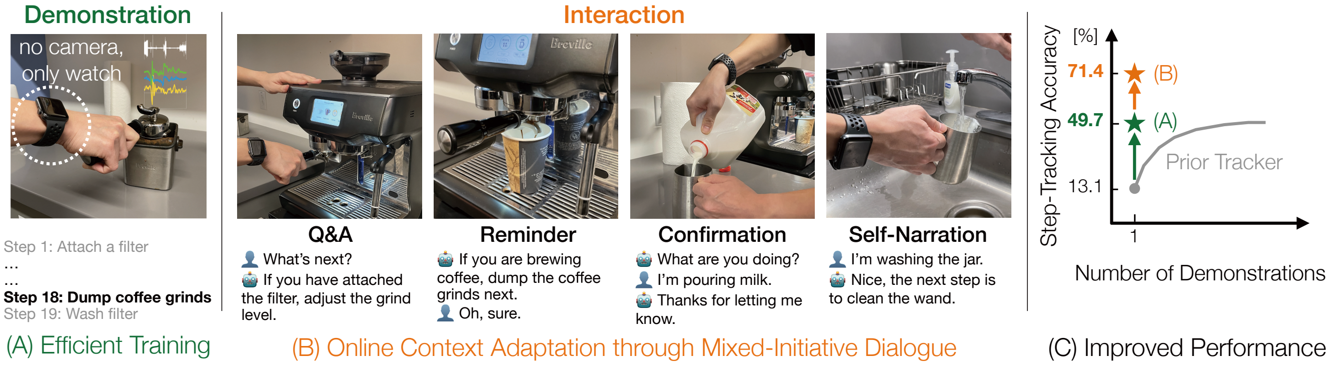
Riku Arakawa
rarakawa@cs.cmu.edu
Carnegie Mellon University
Pittsburgh, United States

Prasoon Patidar
ppatida2@andrew.cmu.edu
Carnegie Mellon University
Pittsburgh, United States

Will Page
wpage@andrew.cmu.edu
Carnegie Mellon University
Pittsburgh, United States

Jill Fain Lehman
jfl@andrew.cmu.edu
Carnegie Mellon University
Pittsburgh, United States

Mayank Goel
mayankgoel@cmu.edu
Carnegie Mellon University
Pittsburgh, United States



Step 1: Attach a filter
...
...
Step 18: Dump coffee grinds
Step 19: Wash filter

Q&A
👤 What's next?
🤖 If you have attached the filter, adjust the grind level.

Reminder
🤖 If you are brewing coffee, dump the coffee grinds next.
👤 Oh, sure.

Confirmation
🤖 What are you doing?
👤 I'm pouring milk.
🤖 Thanks for letting me know.

Self-Narration
👤 I'm washing the jar.
🤖 Nice, the next step is to clean the wand.

Figure 1: Sensor-based task assistants typically require substantial training for a single target task and still suffer from error-prone interactions. Our PrISM framework facilitates more scalable development of such context-aware assistants. (A) It supports the efficient training of sensing models from a demonstration. (B) It adapts its understanding of the user's actions and context through mixed-initiative dialogue. (C) Such scalable training for new tasks and real-time adaptation enhances the system's tracking accuracy and interaction quality, all without additional user effort (numbers are extracted from the results of Studies #1 and #2, latte-making task).

Abstract

Daily tasks such as cooking, machine operation, and medical self-care often require context-aware assistance, yet existing systems are hard to scale due to high training costs and unpredictable and imperfect performance. This work introduces the PrISM framework, which streamlines the process of creating an assistant for users' own tasks using demonstration and dialogue. First, our tracking algorithm effectively learns sensor representation for steps in procedures from a single demonstration. Second, and critically, to tackle the challenges of sensing imperfections and unpredictable user behaviors, we implement a dialogue-based context adaptation mechanism. The dialogue refines the system's understanding in real time, thereby reducing errors such as inappropriate responses to user queries. Evaluated through multiple studies involving several examples of daily tasks in a user's life, our approach demonstrates improved step-tracking accuracy, enhanced user interaction, and an improved sense of collaboration. These results promise a scalable, multimodal, context-aware assistant that effectively bridges the gap between human guidance and adaptive support in diverse real-world applications.

UIST '25, Busan, Republic of Korea

ACM ISBN 979-8-4007-2037-6/2025/09
https://doi.org/10.1145/3746059.3747700

CCS Concepts

• **Human-centered computing → Ubiquitous and mobile computing systems and tools**; *Interactive systems and tools.*

Keywords

task-support assistant, mixed-initiative dialogue, step tracking

ACM Reference Format:
Riku Arakawa, Prasoon Patidar, Will Page, Jill Fain Lehman, and Mayank Goel. 2025. Scaling Context-Aware Task Assistants that Learn from Demonstration and Adapt through Mixed-Initiative Dialogue. In *The 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25), September 28-October 1, 2025, Busan, Republic of Korea.* ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3746059.3747700

1 Introduction

While performing daily tasks like cooking or machine operation, users face challenges, such as missing critical steps and forgetting what to do. Digital assistants can help overcome these challenges by providing situated guidance and proactive intervention as though another human were present [42, 77]. Such assistants require context awareness [36]. One popular way to provide context is by using a camera [16, 35, 77], but a more practical, privacy-preserving, and comfortable alternative to a camera is a watch-based system [43, 52] (*i.e.*, microphones and IMUs).

Our **PrISM** framework, **Pr**ocedural **I**nteraction from **S**ensing **M**odule, supports people performing various real-world tasks by enabling context-aware assistants powered by multimodal sensors such as smartwatches. To provide the required context during user activities, *PrISM-Tracker* [8] applies a graph-based tracking algorithm to multimodal Human Activity Recognition (HAR) data, identifying which step a user is currently performing within a procedure. Building on this, *PrISM-Q&A* [5] augments Large Language Models (LLMs) with tracking outputs to enable context-aware question answering (Q&A), while *PrISM-Observer* [6] enables proactive interventions by reminding users of important steps as they approach the appropriate time to perform them.

These systems are in various stages of deployment in clinics and postoperative patient homes to help users recover and care for their surgical wounds [30, 73]. Although promising in a lab setting, when used in clinics and homes, such sensing-based systems struggle for two reasons. First, training these ML-based systems involves significant effort; for instance, a PrISM-Observer study [6] reported needing 17 user sessions for a single cooking task. Second, some sensors (especially non-visual ones) have a low signal-to-noise ratio, leading to noisy ML models. When tested in real time, these models result in erroneous interactions, such as ill-timed reminders and inappropriate responses to user queries. Even when used in a controlled lab environment, a PrISM-Q&A study [5] reported that 27% of the responses of a context-aware assistant were incorrect due to erroneous sensing of user actions. These deficiencies are consistent with other studies, too; even when cameras are used [83]. Consequently, sensor-based task assistants are limited in scalability because they require extensive data collection effort to train the sensing model for every new task and can still remain very inaccurate with no ability to recover from errors.

To address these challenges and improve the scalability of sensor-based task assistants in prior work [5, 6, 8], we introduce two features to the PrISM framework: (i) an efficient training method for sensing models (Figure 1A) and (ii) a mixed-initiative dialogue between the user and the assistant to update the assistant's contextual understanding in real-time (Figure 1B). We make training efficient by using a single demonstration of the procedure to achieve tracking accuracy comparable to prior methods that use data from multiple sessions from multiple users. This demonstration does not involve any cameras, and all sensor data (motion and sound) is from a consumer watch. To extract more signal from less data, our model integrates a filtering mechanism to distinguish crucial moments from idle moments and estimates how users transition between steps to refine the tracking output. To further improve performance and adapt to the user's variable context, the PrISM assistant utilizes mixed-initiative [2] dialogues to gather real-time feedback that refines its contextual understanding and addresses uncertainty during the interaction. Specifically, it infers the context from spontaneous dialogue exchanges, such as Q&A [5], user self-narration, and the assistant's proactive reminders [6] and confirmation. For instance, the PrISM assistant can both proactively inquire about the user's current step in cases of high uncertainty and remind the user about critical steps, updating its contextual understanding based on the user's response. We employ an LLM-based pipeline to extract contextual information from various dialogue exchanges and utilize it to improve the stochastic modeling of the user behavior.

We conducted three studies to evaluate the PrISM assistant. First, we demonstrated the efficacy of the training method in seven daily procedural tasks with smartwatch sensors (Study #1, total 190 sessions). For instance, with just one demonstration of a latte-making task, the proposed tracking resulted in 49.7% frame-level accuracy (Macro-F1), while the prior PrISM-Tracker resulted in 13.1% accuracy. At first glance, these performance numbers seem low, but the models provide a warm start to facilitate dialogue between the user and machine, and as the dialogue increases, users derive increased benefits from these sensing models.

Next, we developed a watch-based prototype to assist novice users in making a latte with an unfamiliar espresso machine with no human intervention. We conducted a user study (Study #2, *N*=20), comparing the PrISM assistant, which can adapt to the user context through *collaborative* mixed-initiative dialogue, against a *passive* Q&A system that only answered user questions (*i.e.*, [5]). Powered by user input and clarifications, the online context adaptation mechanism improved frame-level tracking accuracy from 45.4% to 71.3%. Further, due to the improved tracking, participants experienced fewer inappropriate responses from the collaborative assistant (9.4% *vs.* 27.1%) and reported an improved sense of collaboration and less cognitive load.

The flexibility offered by our mixed-initiative dialogue design appears critical in achieving a good balance of control between users and the PrISM assistant, as demonstrated in our multi-session study (Study #3, *N*=6). We observed that all participants engaged effectively with the assistant and that the number of queries and task completion times reduced over time.

Recent AI and sensing advances amplify the promise of intelligent assistants for physical tasks [69]. However, such sensor-based systems will need to be *scalable* and *practical*. This work provides an efficient and easy-to-train step tracker algorithm and a dialogue-based context adaptation mechanism to address uncertainty during interactions. As a result, the integrated PrISM framework allows end-users to create their own task assistants in various contexts, such as physics experiments, knitting, and exercising, as demonstrated by our student participants in Section 8. We open-source our code at https://github.com/cmusmashlab/prism, and will continue to maintain it with continued deployments across use cases and populations.

2 Related Work

Our work builds primarily on the fields of human activity recognition, context-aware task-support interactions, and human-in-the-loop systems.

2.1 Human Activity Recognition (HAR)

For task assistants, capturing the dynamic context—specifically, what the user is doing within a procedure—is essential [36]. One common approach is to develop specialized hardware [65, 72] or use camera-based systems [18, 35, 39]. However, creating new devices can be costly and may not generalize well to different tasks. Camera-based systems also pose challenges, such as placement, ensuring a good view of the action, privacy concerns, and user discomfort [47].

Researchers have also explored more ubiquitous approaches, such as using smartwatches [14, 37, 57] or ambient sensors [26, 60, 61] to recognize user actions, *i.e.*, Human Activity Recognition (HAR). Various sensors have been explored, including microphones [43, 79], IMU sensors [38], Doppler RADAR [1, 46], 2D LiDAR [49, 61], or a combination of these [52, 54]. However, their limited accuracy remains a major challenge for designing context-aware interactions, particularly when sensors are applied to real-world, complex task scenarios [21]. Procedural tasks often include steps that generate subtle signals, making them difficult for sensors to detect [13, 68].

Prior work has sought to improve tracking accuracy by incorporating task transition information as post-processing for frame-level HAR [8, 41, 81], but these methods still fall short, reporting almost 50% errors. This poses a critical issue, as misinterpreted context can lead to unhelpful, even annoying, interactions, especially when errors persist. Moreover, existing methods necessitate numerous demonstrations, which complicates the process of developing a new sensing model for a procedure. Ideally, a user should need to demonstrate only once, allowing the sensing model to be initialized with reasonable accuracy. This work introduces such an approach and demonstrates its feasibility.

2.2 Context-Aware Task-Support Interactions

HCI researchers have long studied interactions for physical task support in a variety of applications, such as physical computing system for electronics [78], smart kitchen for cooking [31, 65, 72] and mixed-reality for cooking [83], assembly [3, 66], and machine operation [17, 35]. Among them, voice-based interactions have become popular due to their availability on different platforms (*e.g.*, smartwatches, home speakers *etc.*) [63] and the recent surge in the development of LLMs [27, 53], leading to many supporting systems in various daily scenarios [28, 36, 75]. We focus on augmenting these task-support voice assistants with sensing technology.

Question-answering (Q&A) is a popular form of interaction for users engaged in complex tasks. Several studies employing the Wizard-of-Oz method [24] have analyzed the types of questions users frequently ask [36, 74, 75], reporting that step-related questions (*e.g.*, next action) are common in procedural tasks. The rapid advancement in LLMs has largely improved Q&A quality with various applications being proposed [20, 51]. Here, sharing context is essential in developing effective dialogue-based interactions [5, 12, 22, 74], as users often struggle to articulate queries clearly enough to receive accurate responses [9, 85] and frequently rely on references that are hard for language models to understand, such as pronouns [29]. Our prior work, PrISM-Q&A [5] extended the LLM-based Q&A with the HAR technology to generate context-aware responses.

Proactive intervention from assistant systems is another form of interaction to prevent inherent errors [34]. Chan *et al.* [18] developed a wearable-camera-based system for detecting clinical medication errors. Laput *et al.* [43] proposed a smartwatch-based interaction that leads users step-by-step by leveraging HAR techniques. Our prior work, PrISM-Observer [6] extended Laput *et al.*'s approach to enable system interventions as users naturally perform tasks instead of having to wait for step-by-step guidance.

While different interaction styles have been proposed, there has been limited exploration into how an assistant that integrates multiple forms of interaction can effectively support users, especially when there is uncertainty in context understanding. Multiple studies have documented instances of failure when the sensing model erred, prompting the need for solutions [5, 18]. This highlights a gap in prior research employing the Wizard-of-Oz [24] method to explore the *imaginary perfect* interaction for voice assistants [36, 74]. Here, we argue that the interaction should be *reciprocal*: the assistant should adapt and update its context understanding through dynamic feedback from users, even when such human feedback is spontaneous within the dialogue. This work introduces such an online context adaptation mechanism and demonstrates its effectiveness.

2.3 Real-Time Human-in-the-Loop Systems

Human-in-the-loop systems involve humans actively participating in running automated processes, allowing humans to provide feedback, make adjustments, or intervene when necessary to improve system performance or outcomes [80]. This approach is commonly used in interactive systems to ensure that the system adapts to real-world complexities while benefiting from human judgment and input [7, 19, 33, 48, 87]. For instance, Hatori *et al.* [33] developed a pick-and-place robot that can interactively execute tasks while talking to human operators to clarify ambiguity.

While the efficacy of human-in-the-loop systems has been demonstrated, most rely on an explicit, dedicated feedback process. Depending solely on this approach, however, can burden users, and they might not provide sufficient data in real-world scenarios, as noted by Unhelkar *et al.* [71]. To address this, researchers are exploring the use of natural but less structured signals, such as real-time reactions to a robot's performance [4, 10, 40]. These methods must be developed with care, as the feedback is implicit and difficult to capture accurately. In this work, we propose a computational approach that integrates unstructured dialogue exchanges with a structured step-tracking state of task assistants.

3 Proposed Approach

Figure 2 shows our proposed PrISM framework, which integrates and extends our prior work in PrISM-Tracker [8], PrISM-Q&A [5], and PrISM-Observer [6]. To address the scalability issue, this work makes the following novel technical advances:

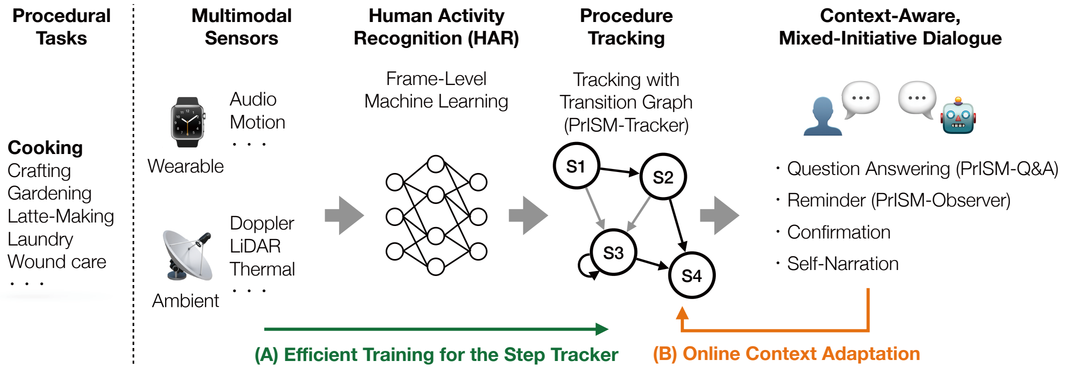

Procedural Tasks
Multimodal Sensors
Human Activity Recognition (HAR)
Procedure Tracking
Context-Aware, Mixed-Initiative Dialogue
Cooking
Crafting
Gardening
Latte-Making
Laundry
Wound care
. . .
Audio
Motion
. . .
Wearable
Doppler
LiDAR
Thermal
. . .
Ambient
Frame-Level Machine Learning
Tracking with Transition Graph (PrISM-Tracker)
S1
S2
S3
S4
· Question Answering (PrISM-Q&A)
· Reminder (PrISM-Observer)
· Confirmation
· Self-Narration
(A) Efficient Training for the Step Tracker
(B) Online Context Adaptation

Figure 2: In the PrISM framework, multimodal sensor signals are processed by a step tracker to infer task context, enabling the assistant to engage in a mixed-initiative dialogue. This paper presents (A) an efficient training method for the step tracker and (B) an online context-adaptation mechanism. We evaluate this framework using a watch-based prototype in this paper.

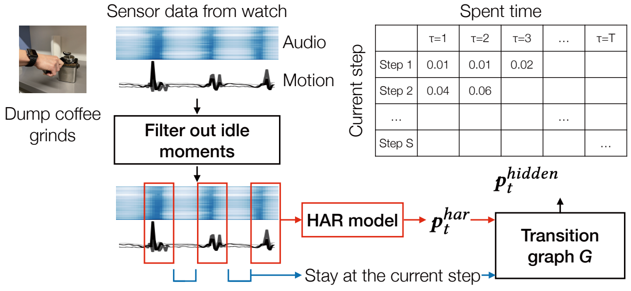
- An efficient training method for the step tracking that effectively learns from a demonstration with signal filtering and automatic generation of a transition graph (Figure 2A).
- A method to identify contextual information using a mixed initiative dialogue framework [2] and dynamically update contextual understanding to improve the stochastic modeling of the step tracker (Figure 2B).

Our real-time system maintains low latency while processing speech and sensor data and managing the dialogue state. The engineering behind this integration is described in Appendix A. This section focuses on the key research contributions enumerated above.

3.1 Efficient Training for the Step Tracker

As one of the state-of-the-art models for multimodal step tracking, PrISM-Tracker [8] processes sensor data using a HAR model and refines the estimated step probabilities based on transition graph probabilities. When there are T frames and S steps for the task, PrISM-Tracker prepares $S \times T$ hidden states, each labeled as $h(i, \tau)$ $(1 \leq i \leq S, 1 \leq \tau \leq T)$. Here, $h(i, \tau)$ denotes that the user has spent *at least* τ frames on the i-th step at time t. Then, $p_t(i, \tau)$ is used to indicate the probability that the user is at $h(i, \tau)$ at time t. PrISM-Tracker keeps updating the probability for every hidden state, $\boldsymbol{p}_t^{hidden} = \{p_t(i, \tau)\} \in \mathbb{R}^{S \times T}$, with frame-level HAR output $\boldsymbol{p}_t^{har} \in \mathbb{R}^S$ as well as a transition graph G. In this context, the vector $\boldsymbol{p}_t^{har}$ is the HAR output at time t, and the sum of its elements is 1. G holds information on the transition probability between steps and the time an average user spends at each step, which is learned from the demonstration sessions.

The effectiveness of this prior method is limited because it treats all frames equally, including moments of inactivity or irrelevance to the action of interest. In such instances, a probability distribution $\boldsymbol{p}_t^{har}$, nearly randomly estimated by the HAR model, is used to update the hidden state, which can confuse the tracking result. Moreover, multiple demonstrations are needed to obtain G, which adds to the cost of training. To address these issues, we propose a filtering technique and a method to automatically generate the transition graph.



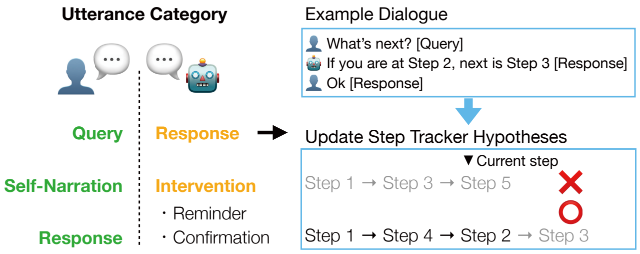
	τ=1	τ=2	τ=3	...	τ=T
Step 1	0.01	0.01	0.02		
Step 2	0.04	0.06			
...				...	
Step S					...

Figure 3: The step tracker filters sensor signals with unsupervised anomaly detection and selectively applies the HAR model to update $\boldsymbol{p}_t^{hidden}$, leading to efficient tracking when the training data is limited.

3.1.1 Filtering out Idle Moments. We adopt real-time, unsupervised anomaly detection for filtering based on the method described by Yamanishi *et al.* [82] (See Figure 3). This method evaluates incoming sensor data to determine deviations from historical data distributions, where multimodal sensor data is concatenated over modalities. The rationale for employing this technique is that procedural actions typically alter sensor signals, unlike periods of inactivity. Essentially, this method helps isolate periods pertinent to active task execution before conducting frame-level HAR.

Concretely, with a predefined threshold β, sensor data frames are identified as either critical (key moments) or non-critical. Frames deemed critical are further processed by a HAR model to derive $\boldsymbol{p}_t^{har}$ (Figure 3 red arrow), which updates $\boldsymbol{p}_t^{hidden}$. Conversely, if a frame is classified as non-critical, $p_{t+1}(i, \tau) = p_t(i, \tau - 1)$, indicating the continuation of the current procedural step (Figure 3 blue arrow).

This filtering process benefits not only the tracking but also the training phase of the HAR model because incorporating too many idle moments with similar signal patterns across different procedural steps can lead to misclassification. We found that excessive filtering may reduce the frequency at which $\boldsymbol{p}_t^{har}$ is generated, potentially impacting the tracker's performance. We empirically



What's next? [Query]

If you are at Step 2, next is Step 3 [Response]

Ok [Response]

Figure 4: We categorized utterances between the user and the assistant based on the mixed-initiative design of Allen *et al.* [2], then introduced an online context adaptation mechanism for the step tracker. In this example, the user first asks a query and then confirms the assistant's response by saying "Ok", so the assistant updates its hypotheses H to focus on those matching the context.

identified β for each task, and it classified approximately 10% of the training data as non-critical.

Note that the HAR model is trained in a supervised manner using step labels synchronized with sensor data. In practice, these labels can be obtained through various methods, such as users pressing a button on their watch at the beginning of each step or verbally narrating their actions while performing the task.

3.1.2 Generating the Transition Graph. The parameters for the transition graph G include (1) the transition probabilities between each step and (2) the mean and standard deviation of the time users spend at each step. The first parameter reflects a logical pattern in performing a procedural task, which we anticipate can be automatically guided by LLMs. Specifically, we instruct an LLM to generate a reasonable order of steps and repeat this process K times to estimate the transition probabilities as a proxy. For the second parameter, we use the time spent during demonstrations as the mean, and α% of this value as the standard deviation for each step. Although these are somewhat naïve heuristics, we empirically show they remain effective, especially when the number of training samples is limited. K and α were empirically set to be 20 and 10, respectively.

3.2 Online Context Adaptation through Mixed-Initiative Dialogue

Although our approach improves the step-tracking accuracy and provides a warm start for the sensing model (as presented in Study #1 result), sensing imperfection remains. Given that sensing imperfections can severely degrade the real-time user experience, we designed a dialogue between the user and the assistant that then informs an online context adaptation mechanism. The overview is presented in Figure 4.

3.2.1 Dialogue Design. We first organize the roles of utterances exchanged between the user and the assistant. Building on prior research discussing versatile user interactions with voice assistants [36, 75], we categorize *user* utterances into three types: *queries*, *self-narrations*, and *responses*. Queries enable users to ask task-related questions or request actions from the assistant, such as setting a timer, with the expectation of receiving a reply. In contrast, self-narrations involve users describing their actions voluntarily without expecting a specific response. Finally, users can also respond to the assistant's utterances, such as reacting to its reminder.

On the other hand, the *assistant*'s utterances are categorized into *responses* to user queries and *interventions* that occur when the assistant initiates dialogue based on its contextual understanding. Examples of interventions include reminding the user of important steps (*e.g.*, *"Do not forget to do ..."* and *"Have you done ...?"*) and asking questions when there is high uncertainty in the estimating context (*e.g.*, *"Can you tell me what you are doing?"*). A hyperparameter controls the maximum number of times the assistant can intervene to ensure we do not overly burden the user. The details are provided in Appendix A.

The assistant's role is rooted in a mixed-initiative design that supports the user achieving a shared goal with an agent [2]. For example, the assistant's confirmation about the user's current step is a type of *subdialogue initiation*, while its reminder can be associated with *unsolicited reporting*. When and what the assistant should say are carefully designed from this perspective, specifically,

- When the user asks a query, the assistant responds to the query.
- When the user self-narrates, the assistant acknowledges and provides additional information, such as suggesting the next step.
- When the user responds, the assistant evaluates whether further interaction is needed.
- If the assistant is unsure of the user's current step, it initiates a clarifying question.
- When the assistant identifies that the user is approaching predefined steps, it triggers a reminder.

The dialogue is generated with an LLM-based pipeline that is prompted to behave as outlined above. It is important to note that our assistant's context-awareness focuses on step information using multimodal sensors, and thus, queries that require visual information cannot be answered in a certain scenario. If such a query is posed, the assistant responds, *"I don't know."*

3.2.2 Extracting Step Context from Dialogue. Our assistant estimates step-related context from the dialogue exchange, determining either the current step or whether a specific step has been completed. For example, if a user asks, *"What should I do next after washing my hands?"* the assistant can infer that the user's current step is washing hands. Similarly, if the assistant asks, *"Have you washed your hands?"* and the user responds, *"No, not yet,"* the assistant can deduce that this step has not been completed. An LLM is prompted with task-specific information and the entire dialogue exchanged between the user and the assistant to determine whether the conversation included (i) current step information, (ii) past step information, (iii) future step information, or (iv) no-context information. If the LLM estimates (i), (ii), or (iii), it also estimates which step is involved.

3.2.3 Updating the Step Tracker. The estimated step context is then used to update the tracker accordingly. Our step tracker calculates and maintains the best hypotheses on the sequence $H(i, \tau) = \{h(1, 1), ...h(i, t)\}$ given the user is currently at $h(i, t)$. The probability for the hypotheses can be denoted as $\boldsymbol{P}_t^{hypo} = \{P_t(i, \tau)\}$, where $P_t(i, \tau)$ indicates the probability for the assistant to follow $H(i, \tau)$ at time t. This mechanism only marginally increases the algorithm's space complexity by storing at most $S \times T$ hypotheses.

The context information is used to update P_t^{hypo}. For instance, if the assistant estimates that the user is currently at the i-th step, hypotheses that contradict the context will be rejected (*i.e.*, their probability is set to be 0). Similarly, if the assistant knows whether the user has already completed the i-the step or not, the probability of any hypotheses that conflict with the information is set to be 0.

3.3 Prototype Implementation

We developed a prototype using an Apple Watch Series 7 and a MacBook Pro with a 16GB M1 processor. The audio and motion data, sampled at 16 kHz and 50 Hz, respectively, are streamed from the watch to the laptop via a custom app. The watch is worn on the wrist of the non-dominant hand to match user preference. The user also wears Bluetooth-connected earbuds to talk to and listen to the server. We used the HAR model developed by Mollyn *et al.* [52] within the step tracker (See Figure 2). More details are described in Appendix A. The system operates in real-time, with the dialogue latency of approximately 2 seconds from the end of user speech to the onset of the system's response.

4 Overview of Studies

We present a series of user studies to explore the effectiveness of our approach. The following is an overview of the key results: in Study #1, with a total of 190 session data, we demonstrate that the proposed training method achieves better step-tracking accuracy compared to the previous tracker when using a single demonstration; in Study #2, we show that the online feedback from mixed-initiative dialogue helps our assistant recover from errors and uncertainty, offering more accurate supports and enhancing user experience (N=20); in Study #3, we find that the assistant is perceived positively by supporting various user behaviors that evolve over time (N=6, four days). We obtained approval from our institution's ethics board before conducting the studies.

5 Study #1: Training Efficacy

We used datasets for seven tasks, which were recorded using smartwatch sensor data. These tasks include coffee-making, tea-making, cereal-making, sandwich-making, latte-making, stencil-making, and wound care. The coffee-making and cereal-making tasks each consist of 8 atomic steps, while the tea-making and sandwich-making tasks consist of 9 steps. We collected data across 32, 33, 26, and 31 sessions for coffee, tea, cereal, and sandwich tasks, respectively. Additionally, we incorporated the latte-making and wound care dataset from [8], which features 19 steps from 22 sessions and 12 steps from 24 sessions, respectively. To further diversify the task variety, we also gathered a new dataset for a stencil-making task, consisting of 17 steps from 22 sessions. The latte-making and stencil-making tasks represent more complex procedures involving machinery, and wound care is a clinical task performed by patients, while the other four are simpler, everyday activities performed in a kitchen. Note that the order of steps to be performed was not fixed. Details for the steps of each task are described in Appendix B.

Table 1: Frame-by-frame tracking accuracy (Macro-F1) comparison for the step trackers trained with a *single* demonstration in Study #1.

Task	Prior tracker [8]	This work
Coffee-making	21.8	44.9
Tea-making	21.5	47.1
Cereal-making	27.6	34.8
Sandwich-making	23.5	44.0
Latte-making	13.1	49.7
Stencil-making	24.6	34.4
Wound care	12.8	14.0

We measured the improvement of our approach from the prior tracker [8] with frame-level accuracy (Macro F-1 Score). We employed a leave-one-session-out evaluation while using a single demonstration as training data. The result is presented in Table 1. Overall, our approach outperforms the prior tracker when trained on a single session, achieving a 16.2% (SD = 6.8) improvement in tracking accuracy. Importantly, the relatively low absolute accuracy underscores the inherent challenge of procedural tracking, primarily due to the low signal-to-noise ratio, as discussed in Section 2.1.

Through ablation analysis, we found that filtering and task graph generation improved tracking accuracy by 6.3% (SD = 3.3) and 12.7% (SD = 6.1), respectively, across all seven tasks. These results demonstrate the effectiveness of the two proposed techniques, particularly the graph generation component. This improvement is likely because PrISM-Tracker [8] was limited by insufficient data for constructing the transition graph G.

While our improved tracker provides a warm start for the context-aware assistant, even with only a single demonstration, the performance varies across tasks. For instance, the accuracy for the wound care task was notably lower than for others. Graph generation was less effective for this task, likely because it lacks a branching structure. Similarly, filtering had a limited impact, suggesting that tasks with low sensor variability are inherently difficult to track, especially when training data is limited. These results highlight two key insights: (1) tasks must produce distinguishable sensor signals to enable accurate tracking, and (2) sensing imperfections persist, necessitating assistant designs that can robustly build on a warm-started tracker.

6 Study #2: Effectiveness of Online Context Adaptation

Study #1 reinforced our motivation to use imperfect sensing reliably, which is contrary to prior work's Wizard-of-Oz paradigm [36, 74]. Interestingly, Study #2 demonstrates that users benefit from such an imperfect assistant thanks to online context adaptation using user-assistant dialogue.

6.1 Task and System Configuration

We used the latte-making task (same as the one used in Study #1) because of its high complexity and greater accessibility for inexperienced users. We trained our step tracker with the same dataset. We provided the content of the machine manual as a system prompt to the LLM for the Q&A module. In addition, the authors decided on candidate steps that are meaningful for reminders (*i.e.*, that users often forget or that are important for safety), that is, Step 15: cleaning the steaming wand and Step 18: dumping coffee grinds (Also see Figure 10 in Appendix B for the task transition graph).

6.2 Design

We followed a between-subjects design with two conditions, *i.e.*, *passive* and *collaborative*. In the collaborative condition, we used our proposed PrISM assistant. In the passive condition, we disabled the state updater and proactive interventions while keeping the Q&A function, *i.e.*, the system answers user questions as in the collaborative condition, but it does not adapt its state, trigger reminders, or ask confirmation questions to check user context. In both conditions, the experimenter explained that the assistant uses information from the watch to try to keep track and answer questions. In the collaborative condition, the experimenter also explained that the assistant uses conversation to track steps, and thus, self-narration is effective in keeping the assistant informed, and the assistant may also remind or ask confirmation questions.

For comparison, we measured several metrics: task completion time, tracking accuracy, and subjective measures of user experience, including NASA-TLX [32] and System Usability Scale [15]. We also introduced three questions related to reliability and a sense of collaboration. Specifically, we asked how much the participants agreed with the statements "the responses to my questions are reliable," (*response reliability*) "The assistant knows what I am doing (*i.e.*, steps) precisely and behaves accordingly," (*tracking accuracy*), and "The assistant's help was worth the effort involved in working with it," (*sense of collaboration*) using a 7-point Likert scale.

6.3 Procedure

We recruited 20 participants (11 female, 9 male, aged 21-38 years) from our institution, none of whom had previously participated in this or related projects for a between-subjects study. Our inclusion criteria targeted individuals unfamiliar with latte preparation and with no experience in using the machine. After onboarding, participants rated their familiarity with voice assistants and chat-style AIs, which is summarized in Table 2 in Appendix B.

Before beginning the task, an experimenter briefly explained the procedure by showing a list of all the steps and a video demonstration. Participants were informed that they could perform the steps in any logical order as long as no steps, such as cleaning the machine, were omitted. Given that watching such instructional videos can prime the users to perform actions in a specific manner and order, we created five different videos with different sequences and randomly assigned one to each participant. Once familiar with the system, participants performed the task at their own pace. Upon task completion, they took the survey on subjective usability measures and participated in semi-structured interviews. The study lasted approximately 30 minutes for each participant, and participants received a compensation of $10 USD.

6.4 Results

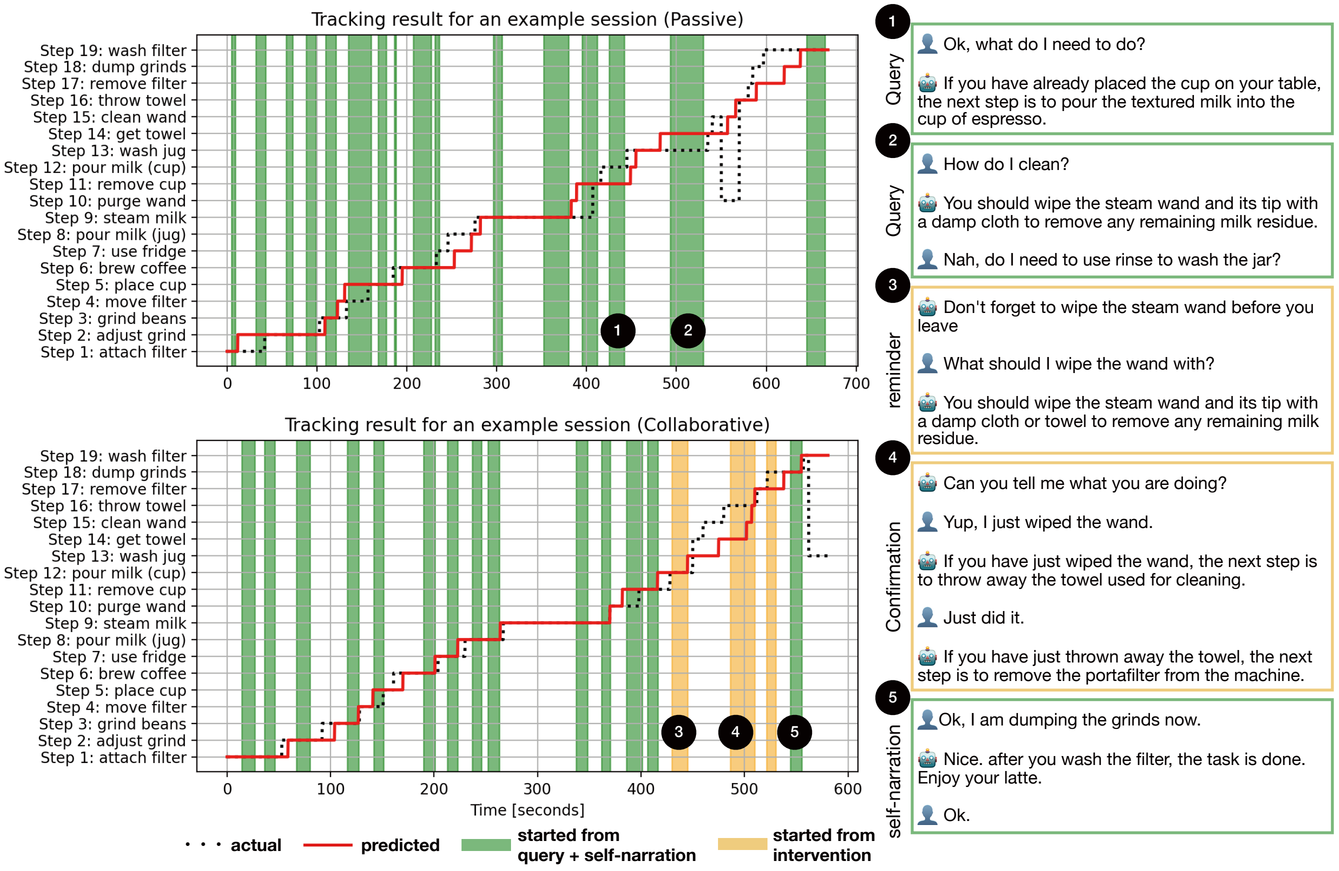
6.4.1 User Task Performance. There were two participants in the passive and one in the collaborative condition who did not complete the full task. In the passive condition, one participant (P5) forgot to do the cleaning process, while the other participant (P7) forgot to place a cup, resulting in unsuccessful brewing. In the collaborative condition, one participant (P18) used a milk jug to receive espresso and used a cup to steam the milk instead. The completion times among the other participants were 633.9 (SD = 283.5) and 482.0 (SD = 115.7) seconds for the passive and collaborative conditions, respectively. After confirming the normality with the Shapiro-Wilk test, we applied the unpaired two-tailed t-test. As a result, the difference was not statistically significant ($t(15) = 1.4,\ p = 0.19$). While there is a trend for shorter completion times in the collaborative condition, the large variance indicates that the time taken to complete the task highly depends on the user.

6.4.2 Assistant's Tracking Accuracy. We annotated the step timings for the session data of the participants who completed the task properly and compared them to the tracker's predictions. Removing moments when dialogue happened, the frame-by-frame accuracy (Macro F1-Score) was 45.4% (SD = 13.2) and 71.3% (SD = 12.8) for the passive and collaborative conditions, respectively. After confirming the normality with the Shapiro-Wilk test, we applied the unpaired two-tailed t-test. As a result, the difference is statistically significant ($t(15) = 9.22,\ p < 0.05$), suggesting that the collaborative assistant had better tracking accuracy thanks to the iterative feedback through dialogue.

Figure 5 shows the ground truth step transition and the assistant's prediction during example sessions. The tracking accuracy for these passive (above) and collaborative (below) conditions was 54.1% and 69.0% (Macro F1-Score), respectively. In the passive condition, it can be seen that the assistant got confused about the user's step in the latter part, as there were several branches in the transition graph (See Figure 10 in Appendix B). As a result, its responses to the user query became less grounded. For instance, in (1), the user's step is wrongly understood (*i.e.*, Step 11: remove cup instead of Step 12: pour milk to cup), and the assistant mentioned unhelpful information in the beginning. Similarly, in (2), the user's query was ambiguous, *"How do I clean?"*, and the assistant initially misunderstood that the user was cleaning the steam wand and offered an incorrect suggestion. On the other hand, in the collaborative condition, the assistant successfully used an active intervention by asking *"Can you tell me what you are doing?"* (4), which helped it recover from the tracking loss through the response from the user. Moreover, it successfully triggered a reminder intervention at a relevant moment (3), which was also used to adjust the prediction afterward. Additionally, the user proactively narrated their step to inform the assistant of their next action (5).

Note that, for the collaborative condition, system logs indicate that 92% of dialogue exchanges were correctly interpreted by our context estimation module (*i.e.*, three context types + right step defined in Section 3.2.2). In error cases, the tracker was typically corrected by subsequent dialogue exchanges.

6.4.3 Dialogue Behavior. We inspected the dialogue for each participant who completed the task. The average number of user utterances per session was 19.6 (SD = 18.8) and 12.7 (SD = 10.1) for the passive and collaborative conditions, respectively. After confirming the normality with the Shapiro-Wilk test, we applied the unpaired two-tailed t-test. As a result, the difference was not



1

👤 Ok, what do I need to do?

🤖 If you have already placed the cup on your table, the next step is to pour the textured milk into the cup of espresso.

2

👤 How do I clean?

🤖 You should wipe the steam wand and its tip with a damp cloth to remove any remaining milk residue.

👤 Nah, do I need to use rinse to wash the jar?

3

🤖 Don't forget to wipe the steam wand before you leave

👤 What should I wipe the wand with?

🤖 You should wipe the steam wand and its tip with a damp cloth or towel to remove any remaining milk residue.

4

🤖 Can you tell me what you are doing?

👤 Yup, I just wiped the wand.

🤖 If you have just wiped the wand, the next step is to throw away the towel used for cleaning.

👤 Just did it.

🤖 If you have just thrown away the towel, the next step is to remove the portafilter from the machine.

5

👤Ok, I am dumping the grinds now.

🤖 Nice. after you wash the filter, the task is done. Enjoy your latte.

👤 Ok.

Figure 5: Example session data for the passive (above) and collaborative conditions (below) in Study #2. Black dotted and red solid lines are the ground truth and the assistant's tracking prediction, respectively. Dialogues happened during the colored time: green for query (*e.g.*, "1" and "2") and self-narration (*e.g.*, "5"), orange for intervention (*e.g.*, "3" for reminder and "4" for confirmation). In the collaborative condition, the tracking was updated following each dialogue, enabling the assistants to maintain accurate context awareness with better tracking accuracy (69.0% *vs.* 54.1%, in the shown example).

statistically significant according to the unpaired two-tailed t-test ($t(15) = 0.89,\ p = 0.39$).

We checked the assistants' responses and judged whether they were inappropriate, then calculated the ratio of inappropriate responses each participant experienced. As a result, 27.1 (SD = 11.0)% of the responses per session were marked as inappropriate in the passive condition, whereas it was 9.4 (SD = 5.3)% in the collaborative condition. After confirming the normality with the Shapiro-Wilk test, we applied the unpaired two-tailed t-test. As a result, the difference was statistically significant according to the unpaired two-tailed t-test ($t(15) = 4.0,\ p < 0.05$). We observed that the majority of inappropriate responses stemmed from incorrect estimations of the current step, particularly when responding to ambiguous queries such as *"What is the next step?"* Thus, the differences can be attributed to enhanced tracking accuracy in the collaborative condition, which is facilitated by the newly introduced state updater.

Moreover, when the participants encountered inappropriate responses, they typically rephrased their inquiries to clarify any ambiguities and asked the assistant again to obtain the correct answer. This could explain the reason for a few participants making many more utterances in the passive condition, as they experienced more inappropriate responses.

Additionally, for the collaborative condition, there were 3.9 (SD = 0.73) active interventions from the assistant per session, *i.e.*, reminders for two steps and confirmation when faced with uncertainty. As discussed below, user comments indicate that they perceived the reminders from the assistant as timely and helpful, and the confirmation queries from the assistant led to an enhanced sense of collaboration.

6.4.4 User Experience Survey. We examined the participants' perceived experience via two instruments. According to an unpaired two-tailed t-test for each factor of the NASA-TLX measurements, the difference was statistically significant in the required effort ($t(18) = 2.4,\ p < 0.05$), performance quality ($t(18) = 2.5,\ p < 0.05$), and the total score ($t(18) = 2.5,\ p < 0.05$). This result indicates that the interaction in the collaborative condition, *i.e.*, the enhanced quality of Q&A and the proactive intervention, reduced the cognitive

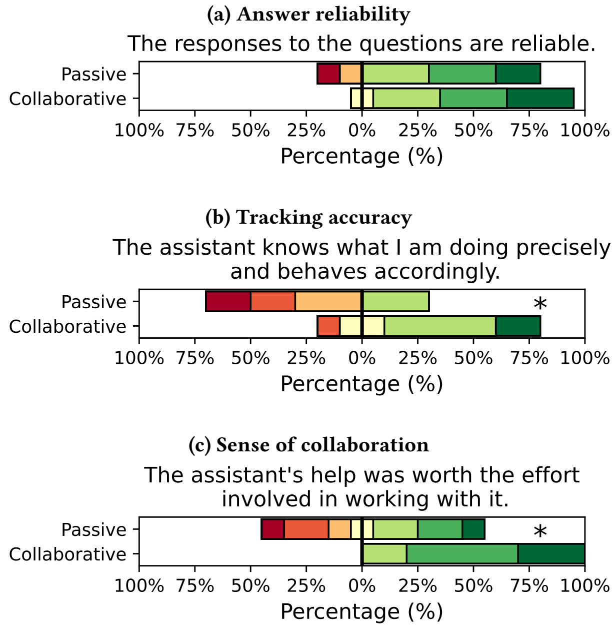


Figure 6: Participants' answers to the user experience questions in Study #2. The color indicates the 7-point Likert scale (red: agree, blue: disagree). There is a significant difference between the conditions for (b) tracking accuracy and (c) sense of collaboration according to a Mann-Whitney U test (*: $p < 0.05$).

load imposed on users performing the task. Similarly, the System Usability Scale score for the collaborative conditions was significantly higher than the passive condition, 68.5 (SD = 18.0) and 83.3 (SD = 10.4), respectively, according to an unpaired two-tailed t-test ($t(18) = 2.1,\ p < 0.05$). These results indicate a preferable experience with the collaborative assistant. The participants' comments corroborated this analysis, as described in the next subsection.

Participants' answers to three additional questions are presented in Figure 6. A Mann-Whitney U test was conducted to compare the answers for each factor between the conditions. The results indicated a significant difference between the two conditions for the tracking accuracy ($U = 20.5$, $p < 0.05$, Figure 6b) and sense of collaboration ($U = 19.5$, $p < 0.05$, Figure 6c). We expected that the fewer inappropriate responses and the timely reminders the participants experienced contributed to the perceived tracking accuracy Moreover, the proactive interventions enhanced the sense of collaboration, which we discuss more in the next subsection.

On the other hand, there was no statistically significant difference in response reliability (Figure 6a). We conjectured that the participants rephrased their questions to get the desired answers if the first responses were inappropriate, as described in Section 6.4.3, and eventually found the answer reliable regardless of the condition. This can be attributed to the designed transparency in the Q&A interaction, that is, mentioning its context understanding first in the response, like *"If you have brewed espresso, the next step is ..."* [5], which enabled easy correction by users.

6.5 User Comments

We analyzed the interview transcription using open coding [23] to gain qualitative insights into the user experience.

6.5.1 *Perceived Tracking Accuracy.*

The participants seemed to judge the assistant's tracking accuracy through the dialogue interaction. Specifically, those who received inappropriate responses when they asked questions due to the incorrect context understanding mentioned that it did not track actions very precisely. *"When I asked what's next, what's next, it can always give me the right answer, except the last few parts. I guess it's not that good to understand the physical world as I do."* [P6, passive]. These comments imply that the inappropriate Q&A responses negatively affected the user experience. On the other hand, participants in the collaborative condition felt that the assistant was accurate when the reminder happened reasonably. *"It was amazing that the system reminded me of cleaning the milk wand in advance, just after I finished using it. Yeah, it was accurate."* [P16, collaborative]. It was suggested that our collaborative assistant provided fewer inappropriate answers and intervened at reasonable moments. This made users feel the assistant understood the context, supporting the result shown in Figure 6b.

6.5.2 *Required Efforts to Use the Assistant.*

When participants experienced inappropriate responses, they tried articulating more to complement the imperfect sensing and be more helpful to the assistant. *"I felt it was a bad idea to ask ambiguous questions like, 'What's the next step?', and instead I tried to mention like, 'I have done brewing coffee, what's next?' and it worked well for most cases."* [P8, passive]. Further comments implied that these behaviors could have affected their perceived cognitive load. *"Since it could answer rough questions like 'What's next?', I did not make extra effort using the system."* [P19, collaborative]. *"It was a bit frustrating that it mistakenly understood my action, so I had to narrate everything every time."* [P10, passive]. These comments support the results on cognitive load discussed in Section 6.4.4.

6.5.3 *Factors Affecting the Sense of Collaboration.*

Comments from users in the passive condition underpinned the lower sense of collaboration in the user experience survey (Figure 6c). *"I definitely was like trying to do this collaborative thing. But when I was trying, it didn't know the step that I was on accurately. So that's when I got a little bit frustrated that I was like trying to collaborate."* [P3, passive]. To hint at the ideal behavior of the assistant, the participants agreed that the assistant's interventions are key to inducing a sense of collaboration. *"It felt like it was assisting me rather than collaboration because when I think of collaboration, usually the effort is 50-50. [...] For collaboration, it could offer, like, 'Hey, is there anything I can do to help you?' So because it responded when I asked for help only, it felt more like an assistant."* [P1, passive]. These comments align with comments from participants in the collaborative condition who experienced such proactive interventions. At the same time, being asked *"Can you tell me what you are doing?"* in the collaborative condition also contributed to the enhanced sense of collaboration. *"I felt that sensing is not perfect. I think that being able to talk back to the agent, the agent asking me where the procedure was helpful to give a sense that it is following."* [P13, collaborative]. These comments suggest that the designed mixed-initiative framework, specifically the assistant's confirmation and reminder, fostered a sense of collaboration, without overburdening the user.

6.6 Summary

In Study #2, we demonstrated the effectiveness of the online context adaptation through mixed-initiative dialogue, which includes the assistant's better tracking accuracy and reduced inappropriate Q&A responses. This also led to a better user experience in terms of System Usability Scale and NASA-TLX, with an enhanced sense of collaboration. The qualitative analysis further sheds light on factors that can affect the user experience, highlighting the effectiveness of the proactive interventions from the assistant as well as the users' adaptive strategy to collaborate with such an imperfect assistant.

7 Study #3: Multi-Session Experience

The previous study confirmed the clear benefit of the proposed system for a single-session experience. However, in daily tasks, users often interact with systems repeatedly over time, with their proficiency improved [44]. To dig deeper into how the user-assistant interaction evolves, we conducted a multi-session study.

7.1 Task and System Configuration

We used the same latte-making task with the collaborative PrISM assistant. Based on the result from Study #2, we slightly modified the reminder feature. Specifically, since two participants did not place a cup properly and failed in brewing, we newly considered Step 5 as a candidate for a reminder. Before each session, we asked each participant to customize the reminders, that is, turning on which step and how (`remind-in-advance` or `notify-if-forgotten`), according to the way described in [6]. We used the same parameters for other parts of the assistant as in Study #2.

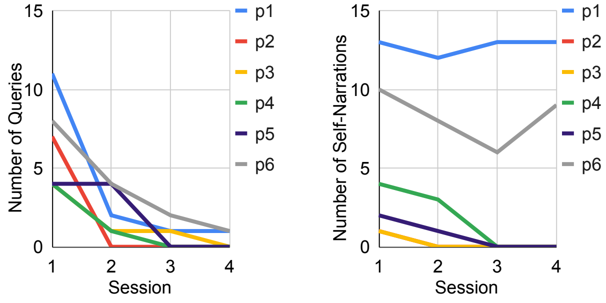
7.2 Procedure

We recruited six new participants from our institution (p1–p6, 5 male and 1 female), who were all beginners in the task. Their background information is presented in Table 3 in Appendix B. During the initial session, each participant arranged to complete four sessions within the week on different days. The instructions provided were consistent with those from Study #2. Prior to each session, participants discussed their preferences for reminders with the experimenter, who then configured the assistant accordingly. It was explained that they could reasonably choose the order of steps and adjust it across sessions, reflecting their natural approach to task execution. Each session lasted approximately 10 minutes. Upon completion of all sessions, we conducted 10-minute semi-structured interviews to explore changes in participants' usage of the assistant. Participants were compensated $10 USD for their involvement.

7.3 Results

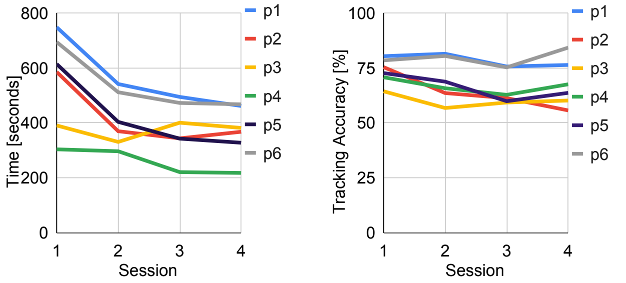
We report the results of the study by the participants' dialogue behavior, user task performance, and the PrISM assistant's step-tracking accuracy over the four sessions.

7.3.1 Dialogue Behavior. The number of utterances by type over sessions for each participant is shown in Figure 7. All participants reduced their queries as they became more familiar with the task over time (Figure 7a). On the other hand, two distinct trends emerge regarding self-narration (Figure 7b). While four participants (p2–p5) stopped narrating their ongoing steps, p1 and p6 maintained a consistent level of self-narration. As discussed later, these participants enjoyed the experience of talking to the assistant about their step-by-step progress. The results suggested individual differences in using our assistant over multiple sessions.



(a) Number of queries **(b) Number of self-narrations**

Figure 7: Number of utterances over sessions for each participant in Study #3.



(a) User task completion time **(b) Assistant's tracking accuracy**

Figure 8: Performance of the user and the assistant over sessions for each participant in Study #3.

7.3.2 User Task Performance. All participants successfully completed the task in every session. The completion times for each participant across the four sessions are shown in Figure 8a. The results indicate that participants required more time to complete the task during the initial session, with completion times decreasing in subsequent sessions. Both p1 and p6 took slightly longer, which aligned with their behavior of engaging in frequent interactions with the assistant (*i.e.*, self-narration), as noted earlier. In contrast, other participants significantly reduced their completion times, as they stopped asking questions once they became familiar with the task and performed it more fluently.

7.3.3 Step-Tracking Accuracy of the Assistant. Figure 8b presents the assistant's frame-by-frame tracking accuracy for each participant across sessions. The assistant maintained high accuracy (∼ 75%) for p1 and p6, as they frequently self-narrated their steps, which helped the assistant continuously update its context understanding. For other participants, the accuracy decreased as they had less linguistic interaction with the assistant over sessions, providing systematically less information to compensate for inherent sensing ambiguity over time. Nevertheless, the accuracy at the last session (~ 62%) was much better than the passive condition in Study #2 (~ 45%). We conjecture that this occurred for two reasons. First, the assistant benefited from responses to proactive interventions, such as reminders and confirmations. Second, as participants became more proficient, their behavior (*i.e.*, how long to spend at each step) increasingly aligned with the training data for the tracker, which was based on non-beginner users.

Note that each participant performed the tasks in a non-consistent order across sessions. We observed that participants flexibly adjusted the sequence, particularly for steps related to the cleaning process, reinforcing our design choice to prioritize user agency rather than enforcing a fixed order.

7.4 User Comments

The semi-structured interviews were analyzed using open coding [23]. Overall, participants provided positive feedback on the experience, noting that the assistant's support helped them learn the task. Simultaneously, we observed individual preferences for different dialogue types.

7.4.1 Evolving Patterns of Voluntary Utterances to the Assistant. Participants recognized that their needs changed over time. Early on, they wanted step-by-step instructions and detailed answers to their questions; later, they only wanted key reminders or error prevention cues. As we observed, four participants (p2–p5) eventually did not talk to the assistant voluntarily. *"I asked a lot in the first sessions and then I got confident and did not have to ask the assistant."* [p2]. In contrast, p1 and p6 liked the experience of doing the task while talking to the assistant. *"I naturally narrated what I was doing after each step, which helped me remember and make sure that I was doing the right thing. Also, hearing the answer from the assistant made me feel I was with someone else, which was fun."* [p6]. As discussed in Section 3.2.1, our assistant responded to the user's self-narration to acknowledge the action and guide the next steps, which seemed favorably accepted by these participants.

7.4.2 Different Strategies for Configuring Reminders. Participants shared different intentions behind their choice of reminders. Three participants mentioned that they wanted `remind-in-advance` reminders consistently even after they got familiar with the task, saying that they could forget things easily. *"I think giving me reminders anyway is helpful. I could be aware of important steps. The timing seemed okay, corresponding to my actions."* [p6]. The other three participants mentioned that they gradually switched `remind-in-advance` to `notify-if-forgotten` as they learned the task, saying that they thought too many reminders were not necessary. At the same time, they still wanted an option to return to reminders if they revisited the task after a long break. *"For example, if I do this process again in a month, then I would definitely want the reminders back."* [p3]. This suggests that the customizability of the reminder is critically helpful in adapting to individual preferences that change over time.

7.4.3 Acceptable Experience of the Assistant's Confirmation Query. The participants mentioned that one or two instances of confirmation from the assistant were acceptable, even after they got used to the task. They even mentioned that it was helpful to give their attention back to the task. *"Sometimes when the assistant is asking what exactly are you doing...that is actually very helpful...it is easy for me to say I'm doing that thing right now"* [p4]. *"It asked me what I was doing and when I said it, it responded to it. That conversation was fun and I felt as if I had been with someone there."* [p1]. While asking for confirmation too frequently could become annoying, these comments suggested that a limited number of proactive interactions from the assistant were generally well-received. This underpins our design of using a hyperparameter to control the maximum number of interventions per session.

7.5 Summary

In Study #3, we explored how user-assistant interaction changes as users learn the task. The results demonstrated that the mixed-initiative design effectively fostered user independence over sessions through its Q&A feature while providing occasional proactive interventions. Furthermore, the PrISM assistant adapted to diverse user behaviors and preferences, including variations in the frequency of self-narrations and the customization of reminders. While prior work [5, 6] has focused on atomic interactions, their impact on the entire user experience, especially across multiple sessions, remained unexplored. This study provides empirical evidence on how the integrated system supports users over time.

8 Application Scenario Exploration

Finally, we explored what kind of applications end-users might create with the PrISM framework. Six STEM graduate students from our institution (four male and two female, aged 26-35 years) participated, none of whom had previously interacted with the PrISM assistant. Initially, we introduced the concept of our task assistant and asked them to envision a scenario where such an assistant would be beneficial.

Each participant first defined a list of steps of their task scenario, and demonstrated an instance of the procedure. The experimenter noted down the timing of each step as an annotation. Then they wrote a step-by-step instruction to supply knowledge to the assistant. We consulted participants about the steps for which they preferred to receive reminders. The collected demonstration data, along with the documented instructions and reminder preferences, were input into our pipeline to train the step tracker and to configure the assistant. Subsequently, participants repeated the same task, this time interacting with our watch assistant. The session took almost an hour, and the participants received $10 USD.

The participants designed assistants for a wide variety of tasks, including preparing glue for a physics experiment, operating a knitting machine, crafting with cardboard, stretching before sports activities, maintaining a regular office cleaning routine, and preparing breakfast. Examples of these are illustrated in Figure 9. Additionally, exemplary interaction scenes are showcased in the Video Figure.

Users were free to design assistants either for personal use or to aid others. For personal applications, participants valued features like reminders (*e.g.*, stretching before sports, cleaning the office, or preparing breakfast) and the Q&A functionality for more complex tasks (*e.g.*, operating a knitting machine). For assisting others, they noted that mixed-initiative dialogue was particularly beneficial for novices (*e.g.*, when preparing glue). The PrISM framework supports both scenarios, and exploring across-session adaptation is a promising direction, particularly with tasks that encourage user stereotypy, as discussed later in Section 9.3.1.

9 Discussion

We demonstrated the efficiency of our training approach in Study #1 and the effectiveness of online context adaptation through mixed-initiative dialogue in Studies #2 and #3. These results have allowed end users to easily prototype their own task assistants, as demonstrated in Section 8. Here, we discuss key implications, potential applications, and important directions for future research.

9.1 Using Imperfect Sensing Reliably

Study #1 highlighted that while training costs for sensing models can be reduced with algorithmic improvements, the accuracy of these models is inherently limited. Therefore, we must focus on building effective user interactions that add reliability to the imperfection of sensing. Mackay [50] explored the idea of human-computer partnership, where humans and intelligent agents perform better together than individually, while Beaudouin-Lafon *et al.* [11] explored *reciprocal co-adaptation*, the point where both the user and the system adapt to and affect the other's behavior to achieve certain goals. Studies #2 and #3 demonstrated that combining sensing and mixed-initiative dialogue offers several benefits to facilitate such a partnership.

Becoming Robust against Uncertainty. Understanding user actions during procedural tasks perfectly is extremely challenging for many sensors [21]. In addition, users can exhibit unexpected behavior that may confuse the assistants. Using mixed-initiative dialogue, including the assistant's proactive query, as online feedback to the context understanding is critical to addressing such uncertainty.

Increasing Transparency and User Trust. Users might struggle to understand the rationale behind an intelligent system's suggestions, especially those based on complex algorithms. This lack of transparency can erode trust and lead to misjudgment [62]. Our assistant describes its context understanding in responses to allow users to correct it interactively, leading to an enhanced sense of collaboration, as discussed in Section 6.4.4.

Sharing Agency Effectively. Allowing for a flexible level of agency to be shared between humans and intelligent agents is key to collaborative interaction. A system that takes over too much control can leave users feeling disempowered [64]. For procedural tasks, it is crucial to design a system that allows users the freedom to follow their own path, disregard the system when needed, or express disagreements.

Adapting to Individual Preference. Dialogue behavior varies by individuals and over time [44]. Thus, maintaining a single balance of interaction would not be optimal. Our framework enables flexibility, such as customizing the frequency of proactive interventions and accommodating frequent self-narrations.

9.2 Application Space

As demonstrated in Section 8, the PrISM framework is capable of generating a variety of sensor-based task assistants. This framework allows for the creation of assistants by users for their own use or for others. For example, users might create their own assistants with proactive reminders to help them avoid mistakes in daily routines.

At the same time, tailoring our approach to specific populations holds significant promise. For instance, individuals with low vision can greatly benefit from such support during activities like cooking [45]. Also, postoperative skin cancer patients can be supported when performing the self-care procedure at home [30, 73]. In this instance, the enhanced tracking capability might also provide valuable information to enable healthcare professionals to monitor patients' self-care routines and track their progress. Such remote monitoring aspects are especially promising in cases like dementia, as discussed by Nunes *et al.* [56] and Wallace [76]. Ultimately, the effectiveness of the assistant is influenced by various factors, including the type of task, user demographics, and their previous experience with repetitive tasks. We plan to deploy our assistant in various settings to further explore these dynamics.

9.3 Limitations and Future Work

Achieving human-level collaboration in real-time human-AI interaction is an extremely challenging goal, and our work represents only a first step in integrating sensing and dialogue. In this section, we outline limitations and key directions for future research.

9.3.1 *Across-Session Adaptation.*

Dialogue-based feedback can be leveraged not just to update the step tracker but also to refine the underlying HAR model and the transition graph G, as post-deployment fine-tuning. Given that the assistant is intended for deployment in each user's environment, the accumulation of daily feedback presents an opportunity for across-session adaptation. For example, fine-tuning G for each user to reduce the variance in the prior distribution will help improve step tracking. Additionally, users may want to provide feedback to customize the assistant's behavior to match their individual preferences and evolving proficiency, as observed in Study #3.

9.3.2 *Implementation beyond Watch.*

While we focused on a watch due to its ubiquity, our step-tracking method is designed to be extended to different platforms. For instance, for delivering interventions, a display can support more visual information. Also, applying muscle stimulation to prevent errors can serve as a powerful and immediate reminder modality, as demonstrated by Nith *et al.* [55].

Moreover, other sensing techniques, as discussed in Section 2.1, can be integrated to complement the sensing capability. Here, the user experience is influenced by the performance of the underlying HAR, as suggested in Study #2. Investigating the relationship between sensing accuracy and end-user experience through experiments will provide valuable information to further support human-AI collaboration under uncertainty [59].

9.3.3 *Assumptions about User Behavior.*

While our assistant is designed to accommodate user initiative instead of providing step-by-step instructions, the current implementation imposes a few constraints on user behavior: users do not perform multiple steps simultaneously, and do not take unexpected actions that are not in

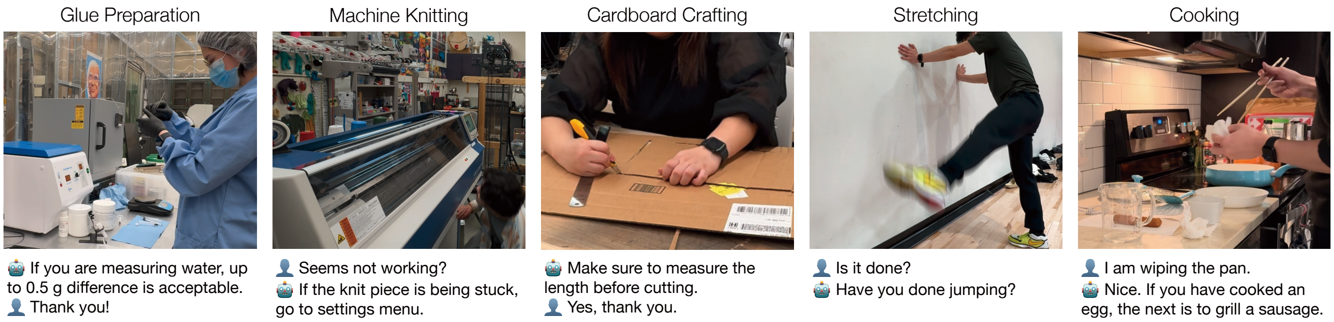

Glue Preparation
🤖 If you are measuring water, up to 0.5 g difference is acceptable.
👤 Thank you!
Machine Knitting
👤 Seems not working?
🤖 If the knit piece is being stuck, go to settings menu.
Cardboard Crafting
🤖 Make sure to measure the length before cutting.
👤 Yes, thank you.
Stretching
👤 Is it done?
🤖 Have you done jumping?
Cooking
👤 I am wiping the pan.
🤖 Nice. If you have cooked an egg, the next is to grill a sausage.

Figure 9: Sample interactions students demonstrated using the PrISM framework with a smartwatch.

the list of steps. Concurrent activity recognition remains an ongoing research area and is considered challenging, even for camera-based systems [70]. The second constraint is based on the transition graph of the step tracker, which would need to be modified to accommodate unexpected actions for enhanced flexibility.

9.3.4 Further Authoring Techniques. Additionally, we plan to enhance the demonstration and configuration processes. By integrating existing task knowledge and enabling users to provide narration during the single-demonstration phase for customization, we aim to explore efficient authoring methods, as recently explored by Dang *et al.* [25] and Yu and Mooney [84]. Also, since the assistant's responses are generated by LLMs, additional safeguards are needed to prevent erroneous information. Here, providing an intuitive way to configure the assistant's behavior will be critical, especially for system designers and end users [86].

10 Conclusion

Existing context-aware task assistants have significant limitations: they require extensive data collection to train sensing models, yet remain error-prone, frequently confusing users with inappropriate responses. Our approach enables training of the step tracker from a single demonstration as a warm start while addressing inherent sensing limitations through mixed-initiative dialogue. Through a series of studies involving various daily procedural tasks, we verified multiple benefits of our approach: improved training efficiency, dynamic adaptation during interaction, enhanced interaction quality and user experience, support for evolving user behaviors, and simplified creation of personalized task assistants. Though future work is needed, this represents a significant step toward intelligent systems by using ubiquitous yet imperfect sensing reliably.

Acknowledgments

This work was partially supported by NSF Award No. 2406099 and the Masason Foundation. We thank Hiromu Yakura for his early contributions to the framework development and Dr. Bryan Carroll for his valuable feedback and continued support during the in-clinic and at-home deployments with skin cancer patients.

References

[1] Karan Ahuja, Yue Jiang, Mayank Goel, and Chris Harrison. 2021. Vid2Doppler: Synthesizing Doppler Radar Data from Videos for Training Privacy-Preserving Activity Recognition. In *CHI '21: CHI Conference on Human Factors in Computing Systems, Virtual Event / Yokohama, Japan, May 8-13, 2021.* ACM, 292:1–292:10. https://doi.org/10.1145/3411764.3445138

[2] James E Allen, Curry I Guinn, and Eric Horvtz. 1999. Mixed-initiative interaction. *IEEE Intelligent Systems and their Applications* 14, 5 (1999), 14–23.

[3] João Bernardo Alves, Bernardo Marques, Carlos Ferreira, Paulo Dias, and Beatriz Sousa Santos. 2022. Comparing augmented reality visualization methods for assembly procedures. *Virtual Reality* 26, 1 (2022), 235–248. https://doi.org/10.1007/s10055-021-00557-8

[4] Riku Arakawa, Sosuke Kobayashi, Yuya Unno, Yuta Tsuboi, and Shin-ichi Maeda. 2018. DQN-TAMER: Human-in-the-Loop Reinforcement Learning with Intractable Feedback. *CoRR* abs/1810.11748 (2018). http://arxiv.org/abs/1810.11748

[5] Riku Arakawa, Jill Fain Lehman, and Mayank Goel. 2024. PrISM-Q&A: Step-Aware Voice Assistant on a Smartwatch enabled by Multimodal Procedure Tracking and Large Language Models. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 8, 4 (2024), 180:1–180:26. https://doi.org/10.1145/3699759

[6] Riku Arakawa, Hiromu Yakura, and Mayank Goel. 2024. PrISM-Observer: Intervention Agent to Help Users Perform Everyday Procedures Sensed using a Smartwatch. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology, UIST 2024, Pittsburgh, PA, USA, October 13-16, 2024.* ACM, 15:1–15:16. https://doi.org/10.1145/3654777.3676350

[7] Riku Arakawa, Hiromu Yakura, and Masataka Goto. 2022. BeParrot: Efficient Interface for Transcribing Unclear Speech via Respeaking. In *IUI 2022: 27th International Conference on Intelligent User Interfaces, Helsinki, Finland, March 22 - 25, 2022.* ACM, 832–840. https://doi.org/10.1145/3490099.3511164

[8] Riku Arakawa, Hiromu Yakura, Vimal Mollyn, Suzanne Nie, Emma Russell, Dustin P. DeMeo, Haarika A. Reddy, Alexander K. Maytin, Bryan T. Carroll, Jill Fain Lehman, and Mayank Goel. 2022. PrISM-Tracker: A Framework for Multimodal Procedure Tracking Using Wearable Sensors and State Transition Information with User-Driven Handling of Errors and Uncertainty. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 4 (2022), 156:1–156:27. https://doi.org/10.1145/3569504

[9] Ahmed Hassan Awadallah, Ranjitha Gurunath Kulkarni, Umut Ozertem, and Rosie Jones. 2015. Characterizing and Predicting Voice Query Reformulation. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015.* ACM, 543–552. https://doi.org/10.1145/2806416.2806491

[10] Agnes Axelsson and Gabriel Skantze. 2021. Multimodal User Feedback During Adaptive Robot-Human Presentations. *Frontiers Comput. Sci.* 3 (2021), 741148. https://doi.org/10.3389/FCOMP.2021.741148

[11] Michel Beaudouin-Lafon, Susanne Bødker, and Wendy E. Mackay. 2021. Generative Theories of Interaction. *ACM Trans. Comput. Hum. Interact.* 28, 6 (2021), 45:1–45:54. https://doi.org/10.1145/3468505

[12] Frank Bentley, Chris Luvogt, Max Silverman, Rushani Wirasinghe, Brooke White, and Danielle M. Lottridge. 2018. Understanding the Long-Term Use of Smart Speaker Assistants. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 3 (2018), 91:1–91:24. https://doi.org/10.1145/3264901

[13] Edgar A. Bernal, Xitong Yang, Qun Li, Jayant Kumar, Sriganesh Madhvanath, Palghat Ramesh, and Raja Bala. 2018. Deep temporal multimodal fusion for medical procedure monitoring using wearable sensors. *IEEE Transactions on Multimedia* 20, 1 (2018), 107–118. https://doi.org/10.1109/TMM.2017.2726187

[14] Sarnab Bhattacharya, Rebecca Adaimi, and Edison Thomaz. 2022. Leveraging sound and wrist motion to detect activities of daily living with commodity smartwatches. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 2 (2022), 42:1–42:28. https://doi.org/10.1145/3534582

[15] John Brooke. 1996. SUS: A 'Quick and Dirty' Usability Scale. In *Usability Evaluation In Industry*, Patrick W. Jordan, B. Thomas, Ian Lyall McClelland, and

Bernard Weerdmeester (Eds.). CRC Press, London, UK, 207–212.

[16] Arthur Caetano, Alejandro Aponte, and Misha Sra. 2024. An Interaction Design Toolkit for Physical Task Guidance with Artificial Intelligence and Mixed Reality. *CoRR* abs/2412.16892 (2024). https://doi.org/10.48550/ARXIV.2412.16892

[17] Yuanzhi Cao, Xun Qian, Tianyi Wang, Rachel Lee, Ke Huo, and Karthik Ramani. 2020. An Exploratory Study of Augmented Reality Presence for Tutoring Machine Tasks. In *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020.* ACM, 1–13. https://doi.org/10.1145/3313831.3376688

[18] Justin Chan, Solomon Nsumba, Mitchell Wortsman, Achal Dave, Ludwig Schmidt, Shyamnath Gollakota, and Kelly E. Michaelsen. 2024. Detecting clinical medication errors with AI enabled wearable cameras. *npj Digit. Medicine* 7, 1 (2024). https://doi.org/10.1038/S41746-024-01295-2

[19] Liwei Chan, Yi-Chi Liao, George B. Mo, John J. Dudley, Chun-Lien Cheng, Per Ola Kristensson, and Antti Oulasvirta. 2022. Investigating Positive and Negative Qualities of Human-in-the-Loop Optimization for Designing Interaction Techniques. In *CHI '22: CHI Conference on Human Factors in Computing Systems, New Orleans, LA, USA, 29 April 2022 - 5 May 2022.* ACM, 112:1–112:14. https://doi.org/10.1145/3491102.3501850

[20] Szeyi Chan, Jiachen Li, Bingsheng Yao, Amama Mahmood, Chien-Ming Huang, Holly Jimison, Elizabeth D. Mynatt, and Dakuo Wang. 2023. "Mango Mango, How to Let The Lettuce Dry Without A Spinner?": Exploring User Perceptions of Using An LLM-Based Conversational Assistant Toward Cooking Partner. *CoRR* abs/2310.05853 (2023). https://doi.org/10.48550/ARXIV.2310.05853

[21] Kaixuan Chen, Dalin Zhang, Lina Yao, Bin Guo, Zhiwen Yu, and Yunhao Liu. 2022. Deep Learning for Sensor-based Human Activity Recognition: Overview, Challenges, and Opportunities. *ACM Comput. Surv.* 54, 4 (2022), 77:1–77:40. https://doi.org/10.1145/3447744

[22] Yuanyuan Chen, Zhengjie Liu, and Juhani Vainio. 2013. Activity-Based Context-Aware Model. In *Design, User Experience, and Usability. Design Philosophy, Methods, and Tools - Second International Conference, DUXU 2013, Held as Part of HCI International 2013, Las Vegas, NV, USA, July 21-26, 2013, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 8012).* Springer, 479–487. https://doi.org/10.1007/978-3-642-39229-0_51

[23] Juliet Corbin and Anselm Strauss. 2008. *Basics of Qualitative Research (3rd ed.): Techniques and Procedures for Developing Grounded Theory.* SAGE Publications, Inc. https://doi.org/10.4135/9781452230153

[24] Nils Dahlbäck, Arne Jönsson, and Lars Ahrenberg. 1993. Wizard of Oz studies: why and how. In *Proceedings of the 1st International Workshop on Intelligent User Interfaces, IUI 1993, Orlando, Florida, USA, January 4-7, 1993.* ACM, 193–200. https://doi.org/10.1145/169891.169968

[25] Hai Dang, Ben Lafreniere, Tovi Grossman, Kashyap Todi, and Michelle Li. 2025. Authoring LLM-Based Assistance for Real-World Contexts and Tasks. In *Proceedings of the 30th International Conference on Intelligent User Interfaces (IUI '25).* ACM, 211–230. https://doi.org/10.1145/3708359.3712164

[26] Devleena Das, Yasutaka Nishimura, Rajan P. Vivek, Naoto Takeda, Sean T. Fish, Thomas Plötz, and Sonia Chernova. 2023. Explainable Activity Recognition for Smart Home Systems. *ACM Trans. Interact. Intell. Syst.* 13, 2 (2023), 7:1–7:39. https://doi.org/10.1145/3561533

[27] Xin Luna Dong, Seungwhan Moon, Yifan Ethan Xu, Kshitiz Malik, and Zhou Yu. 2023. Towards Next-Generation Intelligent Assistants Leveraging LLM Techniques. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023.* ACM, 5792–5793. https://doi.org/10.1145/3580305.3599572

[28] Alexander Frummet, Alessandro Speggiorin, David Elsweiler, Anton Leuski, and Jeff Dalton. 2024. Cooking with Conversation: Enhancing User Engagement and Learning with a Knowledge-Enhancing Assistant. *ACM Trans. Inf. Syst.* 42, 5 (2024), 122:1–122:29. https://doi.org/10.1145/3649500

[29] Raymonde Guindon, Kelly Shuldberg, and Joyce Conner. 1987. Grammatical and Ungrammatical Structures in User-Adviser Dialogues= Evidence for Sufficiency of Restricted Languages in Natural Language Interfaces to Advisory Systems. In *25th Annual Meeting of the Association for Computational Linguistics, Stanford University, Stanford, California, USA, July 6-9, 1987.* ACL, 41–44. https://doi.org/10.3115/981175.981181

[30] Megan V. Ha, Emma Russell, Haarika A. Reddy, Alexander K. Maytin, Dustin P. DeMeo, Riku Arakawa, Mayank Goel, Jill F. Lehman, and Bryan T. Carroll. 2024. Self-narration for patient monitoring with smartwatch technology in post-operative wound care after dermatologic surgery. *Archives of Dermatological Research* 316, 7 (June 2024). https://doi.org/10.1007/s00403-024-03149-z

[31] Reiko Hamada, Jun Okabe, Ichiro Ide, Shin'ichi Satoh, Shuichi Sakai, and Hidehiko Tanaka. 2005. Cooking navi: assistant for daily cooking in kitchen. In *Proceedings of the 13th ACM International Conference on Multimedia, Singapore, November 6-11, 2005.* ACM, 371–374. https://doi.org/10.1145/1101149.1101228

[32] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. Vol. 52. 139–183. https://doi.org/10.1016/s0166-4115(08)62386-9

[33] Jun Hatori, Yuta Kikuchi, Sosuke Kobayashi, Kuniyuki Takahashi, Yuta Tsuboi, Yuya Unno, Wilson Ko, and Jethro Tan. 2018. Interactively Picking Real-World Objects with Unconstrained Spoken Language Instructions. In *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018.* IEEE, 3774–3781. https://doi.org/10.1109/ICRA.2018.8460699

[34] Jinhui Hu, Cong Xin, Manman Zhang, and Youzhen Chen. 2023. The effect of cognitive load and time stress on prospective memory and its components. *Current Psychology* 43, 2 (Feb. 2023), 1670–1684. https://doi.org/10.1007/s12144-023-04354-1

[35] Gaoping Huang, Xun Qian, Tianyi Wang, Fagun Patel, Maitreya Sreeram, Yuanzhi Cao, Karthik Ramani, and Alexander J. Quinn. 2021. AdapTutAR: An adaptive tutoring system for machine tasks in augmented reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems.* ACM, New York, NY, 417:1–417:15. https://doi.org/10.1145/3411764.3445283

[36] Razan Jaber, Sabrina Zhong, Sanna Kuoppamäki, Aida Hosseini, Iona Gessinger, Duncan P Brumby, Benjamin R. Cowan, and Donald Mcmillan. 2024. Cooking With Agents: Designing Context-aware Voice Interaction. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24).* ACM. https://doi.org/10.1145/3613904.3642183

[37] Dhruv Jain, Hung Ngo, Pratyush Patel, Steven Goodman, Leah Findlater, and Jon Froehlich. 2020. SoundWatch: Exploring Smartwatch-based Deep Learning Approaches to Support Sound Awareness for Deaf and Hard of Hearing Users. In *ASSETS '20: The 22nd International ACM SIGACCESS Conference on Computers and Accessibility, Virtual Event, Greece, October 26-28, 2020.* ACM, 30:1–30:13. https://doi.org/10.1145/3373625.3416991

[38] Jeya Vikranth Jeyakumar, Liangzhen Lai, Naveen Suda, and Mani B. Srivastava. 2019. SenseHAR: a robust virtual activity sensor for smartphones and wearables. In *Proceedings of the 17th Conference on Embedded Networked Sensor Systems, SenSys 2019, New York, NY, USA, November 10-13, 2019.* ACM, 15–28. https://doi.org/10.1145/3356250.3360032

[39] Shian-Ru Ke, Le Uyen Thuc Hoang, Yong-Jin Lee, Jenq-Neng Hwang, Jang-Hee Yoo, and Kyoung-Ho Choi. 2013. A Review on Video-Based Human Activity Recognition. *Comput.* 2, 2 (2013), 88–131. https://doi.org/10.3390/COMPUTERS2020088

[40] W. Bradley Knox and Peter Stone. 2009. Interactively shaping agents via human reinforcement: the TAMER framework. In *Proceedings of the 5th International Conference on Knowledge Capture (K-CAP 2009), September 1-4, 2009, Redondo Beach, California, USA.* ACM, 9–16. https://doi.org/10.1145/1597735.1597738

[41] Yusaku Korematsu, Daisuke Saito, and Nobuaki Minematsu. 2019. Cooking state recognition based on acoustic event detection. In *Proceedings of the 11th Workshop on Multimedia for Cooking and Eating Activities.* ACM, New York, NY, 41–44. https://doi.org/10.1145/3326458.3326932

[42] Balasaravanan Thoravi Kumaravel, Fraser Anderson, George W. Fitzmaurice, Bjoern Hartmann, and Tovi Grossman. 2019. Loki: Facilitating Remote Instruction of Physical Tasks Using Bi-Directional Mixed-Reality Telepresence. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology, UIST 2019, New Orleans, LA, USA, October 20-23, 2019.* ACM, 161–174. https://doi.org/10.1145/3332165.3347872

[43] Gierad Laput, Karan Ahuja, Mayank Goel, and Chris Harrison. 2018. Ubicoustics: Plug-and-play acoustic activity recognition. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology.* ACM, New York, NY, 213–224. https://doi.org/10.1145/3242587.3242609

[44] Jill Fain Lehman. 1992. *Adaptive parsing - self-extending natural language interfaces.* The Kluwer international series in engineering and computer science, Vol. 161. Kluwer.

[45] Franklin Mingzhe Li, Michael Xieyang Liu, Shaun K. Kane, and Patrick Carrington. 2024. A Contextual Inquiry of People with Vision Impairments in Cooking. In *Proceedings of the CHI Conference on Human Factors in Computing Systems, CHI 2024, Honolulu, HI, USA, May 11-16, 2024.* ACM, 38:1–38:14. https://doi.org/10.1145/3613904.3642233

[46] Xinyu Li, Yuan He, Francesco Fioranelli, and Xiaojun Jing. 2022. Semisupervised Human Activity Recognition With Radar Micro-Doppler Signatures. *IEEE Trans. Geosci. Remote. Sens.* 60 (2022), 1–12. https://doi.org/10.1109/TGRS.2021.3090106

[47] Xiang Li, Heqian Qiu, Lanxiao Wang, Hanwen Zhang, Chenghao Qi, Linfeng Han, Huiyu Xiong, and Hongliang Li. 2025. Challenges and Trends in Egocentric Vision: A Survey. https://doi.org/10.48550/ARXIV.2503.15275

[48] Chuan-En Lin, Ta Ying Cheng, and Xiaojuan Ma. 2020. ARchitect: Building Interactive Virtual Experiences from Physical Affordances by Bringing Human-in-the-Loop. In *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020.* ACM, 1–13. https://doi.org/10.1145/3313831.3376614

[49] Fei Luo, Stefan Poslad, and Eliane L. Bodanese. 2020. Temporal Convolutional Networks for Multiperson Activity Recognition Using a 2-D LIDAR. *IEEE Internet Things J.* 7, 8 (2020), 7432–7442. https://doi.org/10.1109/JIOT.2020.2984544

[50] Wendy E. Mackay. 2023. Creating Human-Computer Partnerships. In *Computer-Human Interaction Research and Applications - 7th International Conference, CHIRA 2023, Rome, Italy, November 16-17, 2023, Proceedings, Part I (Communications in Computer and Information Science, Vol. 1996).* Springer, 3–17. https://doi.org/10.1007/978-3-031-49425-3_1

[51] Amama Mahmood, Junxiang Wang, Bingsheng Yao, Dakuo Wang, and Chien-Ming Huang. 2023. LLM-Powered Conversational Voice Assistants: Interaction Patterns, Opportunities, Challenges, and Design Guidelines. *CoRR* abs/2309.13879 (2023). https://doi.org/10.48550/ARXIV.2309.13879

[52] Vimal Mollyn, Karan Ahuja, Dhruv Verma, Chris Harrison, and Mayank Goel. 2022. SAMoSA: Sensing Activities with Motion and Subsampled Audio. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 3 (2022), 132:1–132:19. https://doi.org/10.1145/3550284

[53] Pha Nguyen, Sailik Sengupta, Girik Malik, Arshit Gupta, and Bonan Min. 2025. InsTALL: Context-aware Instructional Task Assistance with Multi-modal Large Language Models. *CoRR* abs/2501.12231 (2025). https://doi.org/10.48550/ARXIV.2501.12231 arXiv:2501.12231

[54] Jianyuan Ni, Hao Tang, Syed Tousiful Haque, Yan Yan, and Anne H. H. Ngu. 2024. A Survey on Multimodal Wearable Sensor-based Human Action Recognition. *CoRR* abs/2404.15349 (2024). https://doi.org/10.48550/ARXIV.2404.15349

[55] Romain Nith, Yun Ho, and Pedro Lopes. 2024. SplitBody: Reducing Mental Workload while Multitasking via Muscle Stimulation. In *Proceedings of the CHI Conference on Human Factors in Computing Systems, CHI 2024, Honolulu, HI, USA, May 11-16, 2024*. ACM, 81:1–81:11. https://doi.org/10.1145/3613904.3642629

[56] Francisco Nunes, Nervo Verdezoto, Geraldine Fitzpatrick, Morten Kyng, Erik Grönvall, and Cristiano Storni. 2015. Self-Care Technologies in HCI: Trends, Tensions, and Opportunities. *ACM Trans. Comput. Hum. Interact.* 22, 6 (2015), 33:1–33:45. https://doi.org/10.1145/2803173

[57] Henry Friday Nweke, Ying Wah Teh, Mohammed Ali Al-garadi, and Uzoma Rita Alo. 2018. Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges. *Expert Systems with Applications* 105 (2018), 233–261. https://doi.org/10.1016/j.eswa.2018.03.056

[58] OpenAI. 2023. GPT-4 Technical Report. *CoRR* abs/2303.08774 (2023). https://doi.org/10.48550/ARXIV.2303.08774

[59] Ioannis Papantonis and Vaishak Belle. 2023. Why not both? Complementing explanations with uncertainty, and the role of self-confidence in Human-AI collaboration. *CoRR* abs/2304.14130 (2023). https://doi.org/10.48550/ARXIV.2304.14130

[60] Ashish Patel and Jigarkumar Shah. 2019. Sensor-based activity recognition in the context of ambient assisted living systems: A review. *J. Ambient Intell. Smart Environ.* 11, 4 (2019), 301–322. https://doi.org/10.3233/AIS-190529

[61] Prasoon Patidar, Mayank Goel, and Yuvraj Agarwal. 2023. VAX: Using Existing Video and Audio-based Activity Recognition Models to Bootstrap Privacy-Sensitive Sensors. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 7, 3 (2023), 117:1–117:24. https://doi.org/10.1145/3610907

[62] Neil Perry, Megha Srivastava, Deepak Kumar, and Dan Boneh. 2023. Do users write more insecure code with AI assistants?. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*. 2785–2799.

[63] E. V. Polyakov, M. S. Mazhanov, A. Y. Rolich, L. S. Voskov, M. V. Kachalova, and S. V. Polyakov. 2018. Investigation and development of the intelligent voice assistant for the Internet of Things using machine learning. In *2018 Moscow Workshop on Electronic and Networking Technologies (MWENT)*. IEEE. https://doi.org/10.1109/mwent.2018.8337236

[64] Janet Rafner, Dominik Dellermann, Arthur Hjorth, Dóra Verasztó, Constance Kampf, Wendy Mackay, and Jacob Sherson. 2022. Deskilling, upskilling, and reskilling: a case for hybrid intelligence. *Morals & Machines* 1, 2 (2022), 24–39.

[65] Ayaka Sato, Keita Watanabe, and Jun Rekimoto. 2014. MimiCook: A cooking assistant system with situated guidance. In *Proceedings of the 8th International Conference on Tangible, Embedded, and Embodied Interaction*. ACM, New York, NY, 121–124. https://doi.org/10.1145/2540930.2540952

[66] Javier Serván, Fernando Mas, José Luis Menéndez, and José Ríos. 2012. Assembly work instruction deployment using augmented reality. *Key Engineering Materials* 502 (2012), 25–30. https://doi.org/10.4028/www.scientific.net/KEM.502.25

[67] Mohan Shi, Yuchun Shu, Lingyun Zuo, Qian Chen, Shiliang Zhang, Jie Zhang, and Li-Rong Dai. 2023. Semantic VAD: Low-Latency Voice Activity Detection for Speech Interaction. In *24th Annual Conference of the International Speech Communication Association, Interspeech 2023, Dublin, Ireland, August 20-24, 2023*. ISCA, 5047–5051. https://doi.org/10.21437/INTERSPEECH.2023-598

[68] Ekaterina H. Spriggs, Fernando De La Torre, and Martial Hebert. 2009. Temporal segmentation and activity classification from first-person sensing. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, Washington, DC, 17–24. https://doi.org/10.1109/CVPRW.2009.5204354

[69] Ryo Suzuki, Mar Gonzalez-Franco, Misha Sra, and David Lindlbauer. 2024. Everyday AR through AI-in-the-Loop. *arXiv preprint arXiv:2412.12681* (2024).

[70] Keshav Thapa, Zubaer Md. Abdullah Al, Barsha Lamichhane, and Sung-Hyun Yang. 2020. A Deep Machine Learning Method for Concurrent and Interleaved Human Activity Recognition. *Sensors* 20, 20 (2020), 5770. https://doi.org/10.3390/S20205770

[71] Vaibhav V. Unhelkar, Shen Li, and Julie A. Shah. 2020. Decision-making for bidirectional communication in sequential human-robot collaborative tasks. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, New York, NY, 329–341. https://doi.org/10.1145/3319502.3374779

[72] Daisuke Uriu, Mizuki Namai, Satoru Tokuhisa, Ryo Kashiwagi, Masahiko Inami, and Naohito Okude. 2012. Panavi: Recipe medium with a sensors-embedded pan for domestic users to master professional culinary arts. In *Proceedings of the 2012 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, 129–138. https://doi.org/10.1145/2207676.2207695

[73] Annalise Vaccarello, Alexander K. Maytin, Yash Kumar, Toluwalashe Onamusi, Haarika A. Reddy, Mayank Goel, Riku Arakawa, Jill Fain Lehman, and Bryan T. Carroll. 2024. Barriers to use of digital assistance for postoperative wound care: a single-center survey of dermatologic surgery patients. *Archives of Dermatological Research* 316, 7 (June 2024). https://doi.org/10.1007/s00403-024-03025-w

[74] Sarah Theres Völkel, Daniel Buschek, Malin Eiband, Benjamin R. Cowan, and Heinrich Hussmann. 2021. Eliciting and Analysing Users' Envisioned Dialogues with Perfect Voice Assistants. In *CHI '21: CHI Conference on Human Factors in Computing Systems, Virtual Event / Yokohama, Japan, May 8-13, 2021*. ACM, 254:1–254:15. https://doi.org/10.1145/3411764.3445536

[75] Alexandra Vtyurina and Adam Fourney. 2018. Exploring the Role of Conversational Cues in Guided Task Support with Virtual Assistants. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI 2018, Montreal, QC, Canada, April 21-26, 2018*. ACM, 208. https://doi.org/10.1145/3173574.3173782

[76] Eric L. Wallace, Mitchell H. Rosner, Mark Dominik Alscher, Claus Peter Schmitt, Arsh Jain, Francesca Tentori, Catherine Firanek, Karen S. Rheuban, Jose Florez-Arango, Vivekanand Jha, Marjorie Foo, Koen de Blok, Mark R. Marshall, Mauricio Sanabria, Timothy Kudelka, and James A. Sloand. 2017. Remote Patient Management for Home Dialysis Patients. *Kidney International Reports* 2, 6 (Nov. 2017), 1009–1017. https://doi.org/10.1016/j.ekir.2017.07.010

[77] Xin Wang, Taein Kwon, Mahdi Rad, Bowen Pan, Ishani Chakraborty, Sean Andrist, Dan Bohus, Ashley Feniello, Bugra Tekin, Felipe Vieira Frujeri, Neel Joshi, and Marc Pollefeys. 2023. HoloAssist: an Egocentric Human Interaction Dataset for Interactive AI Assistants in the Real World. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*. IEEE, 20213–20224. https://doi.org/10.1109/ICCV51070.2023.01854

[78] Jeremy Warner, Ben Lafreniere, George W. Fitzmaurice, and Tovi Grossman. 2018. ElectroTutor: Test-Driven Physical Computing Tutorials. In *The 31st Annual ACM Symposium on User Interface Software and Technology, UIST 2018, Berlin, Germany, October 14-17, 2018*. ACM, 435–446. https://doi.org/10.1145/3242587.3242591

[79] Jason Wu, Chris Harrison, Jeffrey P. Bigham, and Gierad Laput. 2020. Automated Class Discovery and One-Shot Interactions for Acoustic Activity Recognition. In *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020*. ACM, 1–14. https://doi.org/10.1145/3313831.3376875

[80] Xingjiao Wu, Luwei Xiao, Yixuan Sun, Junhang Zhang, Tianlong Ma, and Liang He. 2022. A survey of human-in-the-loop for machine learning. *Future Gener. Comput. Syst.* 135 (2022), 364–381. https://doi.org/10.1016/J.FUTURE.2022.05.014

[81] Qingxin Xia, Atsushi Wada, Joseph Korpela, Takuya Maekawa, and Yasuo Namioka. 2019. Unsupervised factory activity recognition with wearable sensors using process instruction information. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 2 (2019), 60:1–60:23. https://doi.org/10.1145/3328931

[82] Kenji Yamanishi, Jun'ichi Takeuchi, Graham J. Williams, and Peter Milne. 2004. On-Line Unsupervised Outlier Detection Using Finite Mixtures with Discounting Learning Algorithms. *Data Mining and Knowledge Discovery* 8, 3 (2004), 275–300. https://doi.org/10.1023/B:DAMI.0000023676.72185.7c

[83] Jackie (Junrui) Yang, Leping Qiu, Emmanuel Angel Corona-Moreno, Louisa Shi, Hung Bui, Monica S. Lam, and James A. Landay. 2024. AMMA: Adaptive Multimodal Assistants Through Automated State Tracking and User Model-Directed Guidance Planning. In *IEEE Conference Virtual Reality and 3D User Interfaces, VR 2024, Orlando, FL, USA, March 16-21, 2024*. IEEE, 892–902. https://doi.org/10.1109/VR58804.2024.00108

[84] Albert Yu and Raymond J. Mooney. 2023. Using Both Demonstrations and Language Instructions to Efficiently Learn Robotic Tasks. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

[85] Ye Yuan, Stryker Thompson, Kathleen Watson, Alice Chase, Ashwin Senthilkumar, A. J. Bernheim Brush, and Svetlana Yarosh. 2019. Speech interface reformulations and voice assistant personification preferences of children and parents. *Int. J. Child Comput. Interact.* 21 (2019), 77–88. https://doi.org/10.1016/J.IJCCI.2019.04.005

[86] J. D. Zamfirescu-Pereira, Richmond Y. Wong, Bjoern Hartmann, and Qian Yang. 2023. Why Johnny Can't Prompt: How Non-AI Experts Try (and Fail) to Design LLM Prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, CHI 2023, Hamburg, Germany, April 23-28, 2023*. ACM, 437:1–437:21. https://doi.org/10.1145/3544548.3581388

[87] Michael J. Q. Zhang and Eunsol Choi. 2025. Clarify When Necessary: Resolving Ambiguity Through Interaction with LMs. In *Findings of the Association for Computational Linguistics: NAACL 2025, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*. Association for Computational Linguistics, 5526–5543. https://doi.org/10.18653/V1/2025.FINDINGS-NAACL.306

A Details of Implementation

While we describe our key research contributions in Section 3, our prototype system involves several engineering efforts. In this appendix, we describe the details of the implementation.

Dialogue Policy. To enable mixed-initiative dialogue, the system operates three parallel processes. The first is the response process, which receives a user's utterance and generates a reply using an LLM. We follow the implementation of PrISM-Q&A [5], where task-specific instructions are embedded in the LLM's prompt.

The second process executes the PrISM-Observer algorithm [6]. System designers or end-users can customize which steps should trigger reminders, and whether these prompts should occur preemptively or only upon detecting a missed step. Crucially, because the observer monitors the step tracker, the dialogue exchange can dynamically influence its behavior via the state updater. For example, if the assistant initially believes the user is far from a target step, but the dialogue reveals that the user is actually near it, the system adapts and issues a reminder immediately.

The third process periodically monitors the step tracker to compute the entropy of the step hypotheses' probability distribution. If the entropy exceeds a predefined threshold E (a hyperparameter), it signals the dialogue module to initiate a clarification question. In other words, when the system is uncertain about the user's current activity, it may ask, *"Can you tell me what you are doing?"*, initiating an instance of *unsolicited reporting* [2]. To avoid overburdening the user, we limit the number of such confirmation prompts per session to a maximum value M (another hyperparameter).

Speech Processing. We define four states for the assistant to manage speech interaction: *user speaking*, *assistant speaking*, *response waiting*, and *sensor reading*. For instance, after a user asks a question, the assistant enters a *response waiting* state, during which sensor reading is paused; *i.e.*, the step tracker is not updated. Once the assistant provides a response, it transitions back to *response waiting*, awaiting the user's reply. In other words, the step tracker only processes sensor data when no speech interaction is occurring.

Note that we do not implement wake words. Instead, the PrISM assistant monitors the audio channel using voice activity detection (VAD) techniques [67] to manage state transitions. Specifically, it uses real-time VAD with a pre-trained model from `pyannote` to detect when the user is speaking, and applies a 0.7-second silence threshold to identify the end of an utterance. This threshold is based on prior research, which found it to be effective in minimizing recognition errors [67]. User speech is transcribed using `Whisper.cpp` [1] running locally, and Text-to-Speech is handled using the built-in `say` command on macOS.

Hyperparameters. For the HAR model and step tracker, the window length and hop length are 2.88 seconds and 0.21 seconds, respectively. Our PrISM assistant has the following hyperparameters: β as a threshold for the anomaly detection filter, E as a threshold for the entropy to trigger confirmation, and M as the maximum number of times for the confirmation during a session. β was set to filter out roughly 10% of the data points in the training data. Similarly, M was set to be 2, and E was set to ensure that the entropy exceeds E an average of $M = 2$ times per session in the training data by following the approach used in [8].

LLMs. The PrISM assistant incorporates an LLM in three key processes. We use OpenAI's GPT-4o-mini API [58] as the underlying model. First, the step tracker leverages the LLM to generate the transition graph. Second, the dialogue policy employs the LLM in the PrISM-Q&A component to produce appropriate responses. Third, the dialogue exchange is processed by the LLM to estimate the user's current step context. Each of these processes uses a tailored prompt, incorporating task-specific information (*e.g.*, step-by-step instructions and in-context examples). We include prompts and task-specific inputs in the Supplemental File and our code repository.

B Details of Studies

The detailed steps for the tasks we used in Study #1 are presented in Figure 10. The four breakfast tasks are relatively short. The latte-making and stencil-making tasks are longer and have multiple branches in the transition. The wound care task is a single-thread task, which is performed by patient participants at a clinic.

Figure 11 presents the confusion matrix of our step tracker for each task using a single demonstration in Study #1.

Table 2 and Table 3 summarize the participant information in Studies #2 and #3, respectively.

[1] https://github.com/ggerganov/whisper.cpp

Table 2: Participant information in Study #2. VA stands for voice assistant.

ID	Gender	Age	Handedness	Condition	Familiarity with VA	Familiarity with Chat AI
P1	Female	20s	ambidextrous	passive	regular user	regular user
P2	Female	20s	right-handed	passive	somewhat familiar	somewhat familiar
P3	Male	20s	right-handed	passive	regular user	regular user
P4	Female	20s	right-handed	passive	somewhat familiar	regular user
P5	Male	30s	right-handed	passive	somewhat familiar	regular user
P6	Male	20s	right-handed	passive	somewhat familiar	regular user
P7	Male	30s	right-handed	passive	somewhat familiar	regular user
P8	Female	30s	right-handed	passive	somewhat familiar	somewhat familiar
P9	Female	20s	right-handed	passive	somewhat familiar	somewhat familiar
P10	Female	30s	right-handed	collaborative	regular user	regular user
P11	Male	20s	right-handed	collaborative	somewhat familiar	regular user
P12	Female	30s	right-handed	collaborative	regular user	regular user
P13	Female	30s	right-handed	collaborative	regular user	regular user
P14	Female	20s	right-handed	collaborative	somewhat familiar	somewhat familiar
P15	Male	30s	right-handed	collaborative	somewhat familiar	regular user
P16	Male	20s	right-handed	collaborative	regular user	regular user
P17	Female	20s	right-handed	collaborative	somewhat familiar	somewhat familiar
P18	Female	30s	right-handed	collaborative	somewhat familiar	regular user
P19	Male	20s	right-handed	collaborative	somewhat familiar	regular user
P20	Male	20s	right-handed	collaborative	first-time user	regular user

Table 3: Participant information in Study #3. VA stands for voice assistant.

ID	Gender	Age	Handedness	Familiarity with VA	Familiarity with Chat AI
p1	Male	20s	right-handed	somewhat familiar	regular user
p2	Male	20s	right-handed	first-time user	regular user
p3	Male	20s	right-handed	first-time user	regular user
p4	Male	30s	right-handed	somewhat familiar	regular user
p5	Male	30s	right-handed	first-time user	regular user
p6	Female	30s	right-handed	first-time user	somewhat familiar

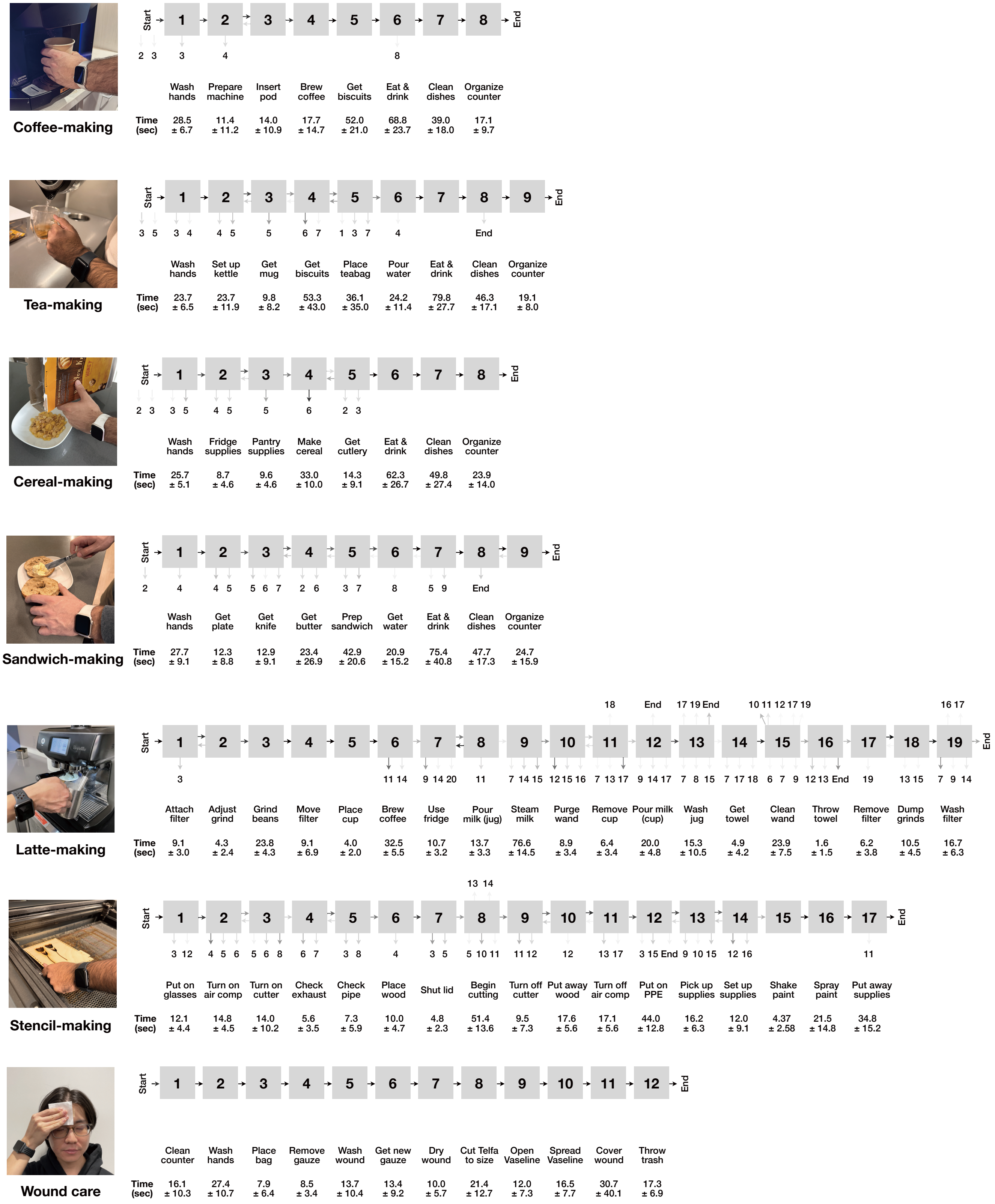

Coffee-making
Start 1 2 3 4 5 6 7 8 End
2 3 3 4 8
Wash hands / Prepare machine / Insert pod / Brew coffee / Get biscuits / Eat & drink / Clean dishes / Organize counter
Time (sec) 28.5 ± 6.7 / 11.4 ± 11.2 / 14.0 ± 10.9 / 17.7 ± 14.7 / 52.0 ± 21.0 / 68.8 ± 23.7 / 39.0 ± 18.0 / 17.1 ± 9.7
Tea-making
Start 1 2 3 4 5 6 7 8 9 End
3 5 3 4 4 5 5 6 7 1 3 7 4 End
Wash hands / Set up kettle / Get mug / Get biscuits / Place teabag / Pour water / Eat & drink / Clean dishes / Organize counter
Time (sec) 23.7 ± 6.5 / 23.7 ± 11.9 / 9.8 ± 8.2 / 53.3 ± 43.0 / 36.1 ± 35.0 / 24.2 ± 11.4 / 79.8 ± 27.7 / 46.3 ± 17.1 / 19.1 ± 8.0
Cereal-making
Start 1 2 3 4 5 6 7 8 End
2 3 3 5 4 5 5 6 2 3
Wash hands / Fridge supplies / Pantry supplies / Make cereal / Get cutlery / Eat & drink / Clean dishes / Organize counter
Time (sec) 25.7 ± 5.1 / 8.7 ± 4.6 / 9.6 ± 4.6 / 33.0 ± 10.0 / 14.3 ± 9.1 / 62.3 ± 26.7 / 49.8 ± 27.4 / 23.9 ± 14.0
Sandwich-making
Start 1 2 3 4 5 6 7 8 9 End
2 4 4 5 5 6 7 2 6 3 7 8 5 9 End
Wash hands / Get plate / Get knife / Get butter / Prep sandwich / Get water / Eat & drink / Clean dishes / Organize counter
Time (sec) 27.7 ± 9.1 / 12.3 ± 8.8 / 12.9 ± 9.1 / 23.4 ± 26.9 / 42.9 ± 20.6 / 20.9 ± 15.2 / 75.4 ± 40.8 / 47.7 ± 17.3 / 24.7 ± 15.9
Latte-making
18 End 17 19 End 10 11 12 17 19 16 17
Start 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 End
3 11 14 9 14 20 11 7 14 15 12 15 16 7 13 17 9 14 17 7 8 15 7 17 18 6 7 9 12 13 End 19 13 15 7 9 14
Attach filter / Adjust grind / Grind beans / Move filter / Place cup / Brew coffee / Use fridge / Pour milk (jug) / Steam milk / Purge wand / Remove cup / Pour milk (cup) / Wash jug / Get towel / Clean wand / Throw towel / Remove filter / Dump grinds / Wash filter
Time (sec) 9.1 ± 3.0 / 4.3 ± 2.4 / 23.8 ± 4.3 / 9.1 ± 6.9 / 4.0 ± 2.0 / 32.5 ± 5.5 / 10.7 ± 3.2 / 13.7 ± 3.3 / 76.6 ± 14.5 / 8.9 ± 3.4 / 6.4 ± 3.4 / 20.0 ± 4.8 / 15.3 ± 10.5 / 4.9 ± 4.2 / 23.9 ± 7.5 / 1.6 ± 1.5 / 6.2 ± 3.8 / 10.5 ± 4.5 / 16.7 ± 6.3
Stencil-making
13 14
Start 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 End
3 12 4 5 6 5 6 8 6 7 3 8 4 3 5 5 10 11 11 12 12 13 17 3 15 End 9 10 15 12 16 11
Put on glasses / Turn on air comp / Turn on cutter / Check exhaust / Check pipe / Place wood / Shut lid / Begin cutting / Turn off cutter / Put away wood / Turn off air comp / Put on PPE / Pick up supplies / Set up supplies / Shake paint / Spray paint / Put away supplies
Time (sec) 12.1 ± 4.4 / 14.8 ± 4.5 / 14.0 ± 10.2 / 5.6 ± 3.5 / 7.3 ± 5.9 / 10.0 ± 4.7 / 4.8 ± 2.3 / 51.4 ± 13.6 / 9.5 ± 7.3 / 17.6 ± 5.6 / 17.1 ± 5.6 / 44.0 ± 12.8 / 16.2 ± 6.3 / 12.0 ± 9.1 / 4.37 ± 2.58 / 21.5 ± 14.8 / 34.8 ± 15.2
Wound care
Start 1 2 3 4 5 6 7 8 9 10 11 12 End
Clean counter / Wash hands / Place bag / Remove gauze / Wash wound / Get new gauze / Dry wound / Cut Telfa to size / Open Vaseline / Spread Vaseline / Cover wound / Throw trash
Time (sec) 16.1 ± 10.3 / 27.4 ± 10.7 / 7.9 ± 6.4 / 8.5 ± 3.4 / 13.7 ± 10.4 / 13.4 ± 9.2 / 10.0 ± 5.7 / 21.4 ± 12.7 / 12.0 ± 7.3 / 16.5 ± 7.7 / 30.7 ± 40.1 / 17.3 ± 6.9

Figure 10: Transition graphs for the tasks we used in Study #1. The opacity of the arrows represents the probability of the transition. In other words, the sum of the transitions of arrows from a single step is 1.0.

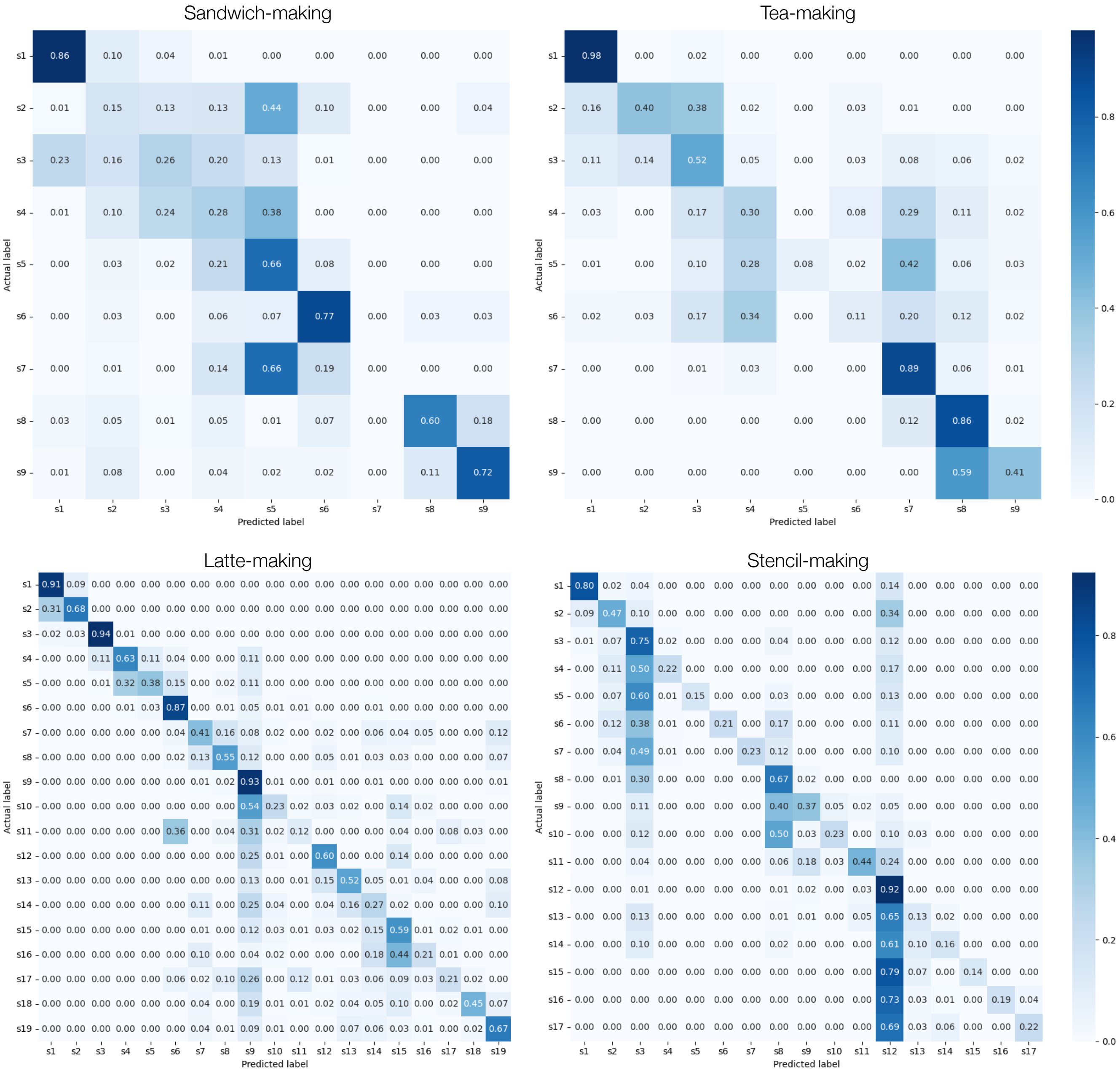


Figure 11: Confusion matrix of our step tracker trained on a single demonstration for the tasks we tested in Study #1. Step labels for each task can be found in Figure 10.