ProMemAssist: Exploring Timely Proactive Assistance Through Working Memory Modeling in Multi-Modal Wearable Devices

Kevin Pu*
jpu@dgp.toronto.edu
University of Toronto
Canada

Ting Zhang
tingzhang@meta.com
Meta Reality Labs
USA

Naveen Sendhilnathan
naveensn@meta.com
Meta Reality Labs
USA

Sebastian Freitag
freitag@meta.com
Meta Reality Labs
USA

Raj Sodhi
rsodhi@meta.com
Meta Reality Labs
USA

Tanya Jonker
tanya.jonker@meta.com
Meta Reality Labs
USA

Abstract

Wearable AI systems aim to provide timely assistance in daily life, but existing approaches often rely on user initiation or predefined task knowledge, neglecting users' current mental states. We introduce ProMemAssist, a smart glasses system that models a user's working memory (WM) in real-time using multi-modal sensor signals. Grounded in cognitive theories of WM, our system represents perceived information as memory items and episodes with encoding mechanisms, such as displacement and interference. This WM model informs a timing predictor that balances the value of assistance with the cost of interruption. In a user study with 12 participants completing cognitively demanding tasks, ProMemAssist delivered more selective assistance and received higher engagement compared to an LLM baseline system. Qualitative feedback highlights the benefits of WM modeling for nuanced, context-sensitive support, offering design implications for more attentive and user-aware proactive agents.

CCS Concepts

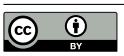
• **Human-centered computing** → **Interactive systems and tools**; **Empirical studies in HCI**.

Keywords

Proactive Assistance; User Modeling; Human-AI Interaction

ACM Reference Format:
Kevin Pu, Ting Zhang, Naveen Sendhilnathan, Sebastian Freitag, Raj Sodhi, and Tanya Jonker. 2025. ProMemAssist: Exploring Timely Proactive Assistance Through Working Memory Modeling in Multi-Modal Wearable Devices. 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/10.1145/3746059.3747770

*Project completed during an internship at Meta Reality Labs.



UIST '25, Busan, Republic of Korea

ACM ISBN 979-8-4007-2037-6/2025/09
https://doi.org/10.1145/3746059.3747770

1 Introduction

Context-aware wearable AI devices such as smart glasses [4], necklaces [2], and pins [3] are beginning to reshape how intelligent assistants support users in daily life. Their hands-free, always-on form factor enables access to real-time sensor data and provides the opportunity to proactively assist users in dynamic, situated contexts — from cooking and organizing to planning or navigating physical spaces [7, 13, 18]. As these systems move beyond desktop and mobile environments, a central challenge emerges: when should they step in to help?

Many existing assistants rely on user-initiated interaction, such as voice commands or manual input. While effective in many contexts, this model assumes that users are aware of what help is possible and cognitively available to ask for it. However, in everyday tasks, users are often mentally occupied or physically engaged, making it difficult to initiate help-seeking even when assistance would be beneficial. More critically, users may not recognize when support is relevant — or may simply forget to ask.

Recent works in proactive assistants attempt to address this issue by triggering assistance based on task context or rule-based heuristics [7, 29, 36, 40, 73]. For example, systems like PrISM-Observer [7] and Satori [36] leverage task step detection or inferred user goals to display timely instructions or reminders. However, these strategies often rely on predefined task structures or heuristic-based triggers. They are limited in their ability to account for internal mental states, such as attention, focus, or cognitive load, which play a critical role in determining whether a user is ready or receptive to assistance. Without such awareness, proactive support risks becoming mistimed, disruptive, or even ignored.

In this paper, we propose a novel approach to inform the timing of assistance: modeling the user's working memory (WM) as a lens into their mental availability and the assistance's value. WM is a cognitive system responsible for temporarily holding and manipulating information during task performance [8, 16, 22]. By modeling the contents and constraints of WM — including capacity limits, recency of information, and susceptibility to interference — we can better infer moments when users are more cognitively open to external input, and conversely, when interruptions may be costly.

We introduce ProMemAssist, a smart glasses system that leverages multi-modal sensor signals (camera and microphone) to construct a real-time model of a user's working memory. The system encodes visuospatial and phonological memory items from the

environment and binds them into episodic chunks that represent meaningful task contexts [8, 22, 48]. A timing predictor uses this WM model to evaluate the potential value of assistance and the cognitive cost of interruption, selecting moments when support is most likely to be helpful and least disruptive. Assistance is generated using a large language model (LLM) based on current memory state. The focus of our work is on *when* assistance should be delivered, rather than *what* assistance to provide. As such, our system does not aim to optimize goal inference and the content of assistance beyond what can be reasonably inferred from the current working memory state.

To evaluate our approach, we conducted a within-subject user study with 12 participants performing real-world tasks that demand both physical interaction and cognitive engagement. Participants completed four tasks (e.g., setting up a dining table, packing for a trip) while receiving assistance either from ProMemAssist or a baseline system where an LLM agent with similar system prompts dictated the timing and generation of support based on the same environmental information. Our findings show that ProMemAssist delivered assistance more selectively depending on the user's WM model, with richer positive engagement and fewer negative reactions.

This paper makes the following contributions:

- A novel working memory modeling framework for determining opportune moments to deliver proactive assistance on wearable devices;
- A proactive timing prediction approach that balances the value and cost of assistance based on cognitive state;
- A user study demonstrating that WM-informed timing leads to improved user experience and engagement.

2 Related Work

2.1 Wearable Device for Task Assistance

Advancements in sensing and lightweight computing devices have led to novel wearable assistants in consumer and research fields alike. LLM-enabled commercial products like Ray-ban glasses [4], AI pin [3], and AI friend necklace [2] produce a plethora of possibilities and controversies regarding how technologies can assist or disrupt people's daily lives. Recent research work has demonstrated the potential of wearable and situated agents to support users in physical environments by leveraging multi-modal input streams and AI models. For example, Arakawa et al. utilized smart watch sensing and task step modeling to predict the user's progress and provide just-in-time interventions [7] and even answer voice queries [6]. Another work, OmniActions, predicts digital follow-up actions, such as looking up information or sharing captured images, based on real-world multi-modal signals using LLMs [37].

In VR/AR environements, works like AMMA adapt task guidance interfaces by modeling user state and planning adaptive step-by-step support [73]. AdapTutAR presents in-situ task guidance by detecting user behavior through AR glasses [30]. Another work, Satori, forecasts the user's task actions by modeling their intention and present assistance in AR [36]. These systems illustrate progress in translating ambient perception into user task understanding and actionable system guidance. Our work differs from these systems by focusing on constructing a model of the user's cognitive state, specifically working memory (WM), rather than inferring perceptibility based on external environmental or task cues. Unlike proactive support that relies on observed task progress or heuristics, we introduce a timing predictor informed by cognitive modeling, enabling interventions that are responsive to mental load and availability.

2.2 Memory Augmentation and Modeling

Memory augmentation has long been a research goal in ubiquitous and wearable computing. Early visionaries like Lamming and Rhodes imagined memory prostheses that could recall contextually relevant past information [33, 61]. More recent work such as SenseCam diaries [34], MemoriQA [66], and OmniQuery [38] support retrospective access to personal memory for tasks like information storage, question-answering, and life-logging [26]. In addition, many works augment user's capabilities to recall information. For example, Memoro offers lightweight real-time augmentation with mixed-initiative memory aid [74], and Shen et al. constructed a VLM-based memory augmentation agent for episodic memory recall tasks [63]. Another tool, AiGet, uses sensing data from AR glasses to construct knowledge library of the user's daily life to provide proactive interventions [17]. Moreover, prior work in XR memory systems explores how immersive technologies can externalize memory to improve recall and reduce load [15, 42]

While these systems primarily focus on retrieval support, ProMemAssist addresses a complementary but underexplored design space: timing real-time proactive support to align with ongoing mental processes. We argue that effective memory augmentation should not only improve recall but also account for the user's cognitive readiness in the moment. Our approach is partially grounded in the Cognitive Load Theory [64], which puts mental workload as a key consideration for interaction and proposes that the timing and modality of intervention could impact user mental workload and receptivity.

One approach to address cognitive receptivity is through physiological sensing. For example, Sarker et al. developed a machine learning model to assess user availability for just-in-time interventions based on wearable sensor data (ECG, accelerometer, respiration) and reported over 74% accuracy in natural environments [62]. Similarly, Chan et al. introduced Prompto, a memory training assistant that initiates prompts when electrodermal activity (EDA) and heart-rate variability (HRV) signals suggest the user is under low cognitive load [19]. By contrast, ProMemAssist infers cognitive availability by constructing mental workload model using observable environmental cues (e.g., visual and auditory signals) rather than physiological signals. While physiological sensing provides fine-grained access to internal states, it can be intrusive, less interpretable, and harder to generalize across users. ProMemAssist offers a more lightweight, explainable model for attention dynamics based on working memory constructs (e.g., recency, interference), and complements prior work by providing an alternative path to cognitively aligned assistance.

Another major application for memory modeling is to simulate and predict user goals or attention via memory modeling [9–11, 32, 51, 67], but few systems have attempted to model the working memory explicitly for timing decisions. MATCHS, for example, uses

WM simulation to adapt interface difficulty for older users or those with cognitive impairments [43, 44], but does not guide assistance delivery timing.

Our contribution is not to offer a definitive cognitive model that can predict user needs and behaviors, but to explore how lightweight WM modeling can inform system timing in interaction. This offers a new mechanism for designing more attentive and personalized just-in-time assistants, particularly in AR and wearable contexts where attention and cognitive load are highly volatile.

2.3 Timing of Service and Interruptibility

A central challenge in mixed-initiative systems is knowing when to assist [5, 27]. Well-timed assistance can ease cognitive effort, avoid confusion, and even foster incidental learning about the system's capabilities [1, 27, 45]. In contrast, poorly timed interventions may negatively impact users' memory, emotional well-being, and ongoing task execution [12, 23, 28].

Past systems have explored timing proactive assistance by modeling user behavior and task states [18, 21, 40, 55, 56]. Additionally, research on notification timing has shown that context-based deferral can improve user experience [20, 39]. Efforts to model interruption cost have led to systems that predict notification acceptability [47] or adapt service robot initiative [13, 52]. While these approaches often use task context, heuristics, or rule-based timing triggers and interruption measurement, ProMemAssist models WM mechanisms such as internal cognitive interference and recency decay, offering a structured basis for timing decisions grounded in WM theory.

The literature also highlights trade-offs in proactivity design. For example, overly proactive agents may be perceived as intrusive, while reactive ones risk missing key support opportunities [51, 71]. By modeling memory state, we aim to strike a more nuanced balance between providing the benefit of intervention and the cost of disruption.

3 Design Rationale

To deliver timely support in dynamic, real-world tasks, proactive assistants must consider not just the external environment but also the user's internal cognitive state [5]. From a user-centered perspective, the optimal moment to provide assistance depends on multiple overlapping factors: (1) the user's current *mental context*—what they know, what they're trying to do, and what might require support; (2) their *attentional focus*—what modality or channel they are currently engaged with; and (3) their *cognitive availability*—how much capacity remains to accommodate new input without inducing overload or disruption.

To capture these factors in a unified framework, we adopt a working memory (WM) model inspired by cognitive psychology and neuroscience [8, 22, 48]. Working memory is a lightweight and transient system responsible for maintaining and manipulating information over short time periods. It has several properties directly relevant to assistance timing—such as capacity limitations, decay over time, and competition between concurrent mental representations [16, 50]. Critically, WM reflects the user's immediate mental state—what they are actively attending to—making it particularly well-suited for predicting receptiveness to new information.

Unlike long-term user modeling or task-based heuristics, WM-based modeling focuses on what is mentally available right now. It allows the system to reason about real-time dynamics—what information is currently being held, how fresh or relevant it is, and whether new input would support or interfere with existing cognitive processes. Our model draws on classic tripartite WM theories that include visuospatial, phonological, and episodic components [8, 72], and computational frameworks that simulate memory encoding, rehearsal, and interference.

This approach enables assistance to be grounded in the structure of ongoing mental activity—moving beyond surface behavior or static context to support proactive timing that is both informed and adaptive.

3.1 Design Goals

To implement this framework, we articulate three system-level goals that shaped the design of ProMemAssist:

- **DG1: Facilitate Real-Time WM Modeling.** The system should continuously model the user's working memory in near real-time to support fine-grained reasoning about when assistance is needed and how cognitively costly it might be. This includes maintaining up-to-date representations of memory items, their recency, modality, and relevance to ongoing context. This design goal stems from the temporal nature of WM, which decays over a short time and demands frequent updates [8, 60].
- **DG2: Utilize Observable and Multi-modal Inputs.** The WM model should be grounded in signals that are practically obtainable in everyday settings. We focus on visual and auditory information, aligning with the visuospatial and phonological stores of WM theory. By using wearable-friendly modalities such as video and audio, we ensure that the system approach remains deployable and applicable to other wearable devices.
- **DG3: Enable Cognitive-Informed Timing Decisions.** The WM model should support principled reasoning about when assistance should be delivered, deferred, or withheld. To do this, it should computationally process WM properties and mechanisms—such as memory encoding, decay, and interference —that reflect the user's current mental state. These processes provide a structured basis for predicting the cognitive impact of potential interruptions and guiding assistance timing in a way that is both adaptive and grounded in WM theory.

4 ProMemAssist

ProMemAssist constructs a real-time WM representation using a pair of smart glasses with camera and microphone and computationally models memory mechanisms to predict the value and cost of potential proactive assistance. Below we illustrate the system workflow with a user scenario walkthrough and detail the implementation.

4.1 User Scenario Walkthrough

Alex wears his smart glasses every day, relying on the onboard assistant to provide timely support for his everyday tasks. Today,

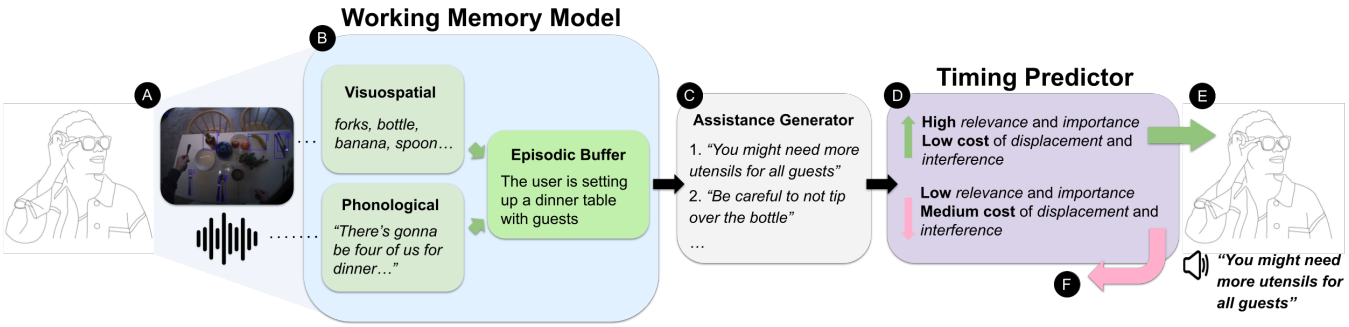


Figure 1: ProMemAssist Workflow. Ⓐ: Multimodal sensor input (visual and auditory) is captured from smart glasses. Ⓑ: Perception memory encodes visual-spatial (e.g., "forks, bottle, banana") and phonological (e.g., "There's gonna be four of us for dinner") signals, which are summarized into an episodic buffer describing the current task context. Ⓒ: The assistance generator uses the working memory state to produce candidate messages. Ⓓ: The timing predictor evaluates each message based on its predicted relevance and importance, as well as the cost of displacing existing memory or causing interference. Ⓔ: Messages with high predicted utility are delivered as proactive voice assistance, while Ⓕ lower-utility messages may be deferred for later evaluation or discarded.

Alex is setting up a dining table to welcome guests for dinner. As he moves through the room, the glasses passively capture his egocentric visual field and surrounding conversations through the built-in camera and microphone. Without any explicit input, ProMemAssist continuously models his real-time working memory state based on what he sees, hears, and interacts with (Fig.1.A).

As Alex begins by placing plates, cups, and cutlery on the table, the system encodes these visual signals into WM as memory items. When he places a coffee cup next to a spoon and hears someone say, *"There's gonna be four of us for dinner,"* ProMemAssist binds this input into a coherent episodic chunk labeled *"The user is setting up a dinner table with guests."* This episodic context is shown in Figure 1.B, where both visual (forks, bottle, banana...) and phonological ("*There's gonna be four of us for dinner…*") inputs contribute to the episode.

With this chunk in WM and the table still in progress, the assistant proactively generates a candidate message: *"You might need more utensils for all guests"* (Fig.1.C). Recognizing the message as highly relevant, important, and unlikely to interfere with the user's curent WM state, ProMemAssist immediately delivers the reminder (Fig.1.D,E). The reminder to bring enough utensils arrives just as Alex is transitioning from cups to silverware, nudging him to adjust his setup without disrupting his focus.

Later, Alex notices a bottle of wine and places it near the edge of the table. ProMemAssist detects the placement and generates a potential reminder: *"Be careful not to tip over the bottle"* (Fig.1.C). However, before the system delivers the message, Alex's friend asks him how many eggs are left in the fridge. As Alex goes to check, ProMemAssist transcribes the conversation and encode the new visual information of objects in the fridge, formulating a new episodic buffer representing this task context. Withholding the comment about the wine bottle, ProMemAssist's timing predictor identifies that Alex's WM is heavily engaged and that an interruption could cause confusion or disrupt task flow (Fig.1.D). The system defers the message and continues monitoring his cognitive state (Fig.1.F). Only once Alex completes egg-counting task and return to the dining table does ProMemAssist re-evaluate and determine that the cost of interference is low and the message still carries value. It then delivers the reminder, prompting Alex to nudge the wine bottle further from the edge just before guests arrive.

As Alex steps back to review the table setup, ProMemAssist notices that no memory items reference napkins. Observing the WM content with lower task importance, it delivers one final reminder: *"Don't forget to put some napkins for your guests."* The prompt lands at just the right moment—before Alex moves on from the current task context.

This scenario illustrates how ProMemAssist adapts its assistance timing based on moment-to-moment shifts in working memory. Rather than relying on fixed rules or scripted sequences, the system reasons over the dynamic structure of human cognition—delivering support that is timely, relevant, and less interruptive.

4.2 Sensing and Information Encoding

We build ProMemAssist on a pair of prototype smart glasses [24], equipped with RGB camera sensors and 7-channel audio microphone. This setup is in-line with common wearable device capabilities and modalities to increase extendability of our approach.

The smart glasses streams visual and auditory information to the system, hosted on a computer, as raw RGB images and audio buffer. We then utilize object-detection [31] and speech-to-text [58] models to extract visuospatial and phonological features in the user's environment.

To represent these information in the working memory model and enable similarity comparison, we embed the visual information (i.e. the visual image of the detected object and its text label) and the auditory information (i.e. the transcribed text of spoken speech) onto CLIP (Contrastive Language-Image Pre-Training) [57] and obtain vector representations of the detected information. We adopt CLIP for its versatility to encode both image and text information and facilitate semantic similarity comparison using the derived vector representations.

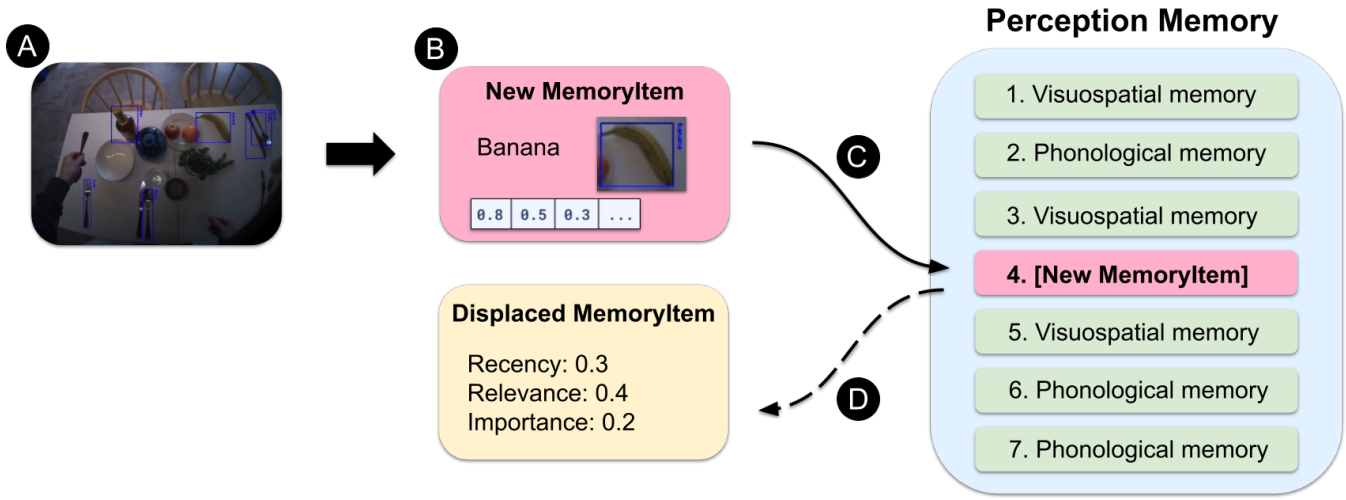


Figure 2: Working Memory Displacement. As the user interacts with objects in the environment, Ⓐ the system captures egocentric visual input as the user interacts with objects in the environment. Ⓑ A new memory item (e.g., a banana) is encoded with a corresponding feature embedding. If the perception memory is already at capacity, the system calculates the recency, relevance, and importance scores for existing items. Ⓒ The memory item with the lowest composite score (here: recency = 0.3, relevance = 0.4, importance = 0.2) is selected for displacement. Ⓓ The new memory item replaces the displaced item and is inserted into the updated perception memory store. This mechanism ensures that the perception memory maintains the most updated information for supporting proactive assistance decisions.

4.3 Working Memory Model

We are inspired by classic works in cognitive psychology to create a computational model of the user's working memory. The most widely studied model proposed by Baddeley et al. theorized the working memory to contain four major components: a high-level central executive that manages and coordinates three lower-level components that store visual-spatial, phonological, and episodic information [8]. We largely adopted this structure in ProMemAssist (Fig. 1.B). In our system, the WM model is represented by two components: a low-level perception memory, which encodes visual-spatial and phonological information, and a high-level episodic buffer that distills perception memory content and formulates summarized descriptions about the user's current context. Different from Baddeley et al.'s model, the episodic buffer is derived from perception memory in our representation as internal episodes are not inherently observable. This design allows us to leverage processed and structured sensor signals and use them to reason about plausible episodic buffer state.

Grounded by past studies and experiments, the perception memory has the capacity to store seven memory items [48] including both visuospatial and phonological information. We create a `MemoryItem` construct with attributes of a timestamp (time of encoding or last activation), type (visuospatial or phonological), content (text label and serailized image for detected object for visuospatial memory, transcribed text for phonological memory), and the feature vectors extracted from the CLIP embedding.

For the episodic buffer, Cowan et al. theorized that working memory content could be bounded and integrated into chunks of information, and that the working memory limit is four chunks of episodes [22]. Binding involves integrating features from different sources into an episode: an integrated chunk that holds multidimensional information[8]. Therefore, ProMemAssist represents chunks in the WM as a list of four custom type `MemoryChunk` objects, each containing a timestamp and the list of `MemoryItem` objects that are bound to this chunk.

4.4 Memory Properties and WM Update

To support real-time modeling of a user's cognitive state, ProMemAssist continuously updates a structured representation of WM through mechanisms theorized by cognitive psychology. To facilitate the mechanisms, each `MemoryItem` is associated with three computationally defined properties—*recency*, *relevance*, and *importance*—that reflect its current cognitive salience and value. These properties enable the system to simulate memory decay, identify key information, and make decisions about displacement and chunking as new information is processed.

4.4.1 Memory Properties. **Recency** captures how recently a memory item was encoded or rehearsed (i.e. reactivated to salience). Consistent with studies of short-term memory decay [14, 46, 60], we model recency as a linear function:

$$\text{Recency} = 1 - \frac{t}{T}$$

where t is the elapsed time since encoding or last rehearsal, and T is the maximum temporal threshold for short-term memory retention. Empirical studies suggest that items in working memory are actively maintained for approximately 15–30 seconds without rehearsal [53, 59]; we adopt $T = 30$ seconds as a reasonable bound in our system.

Relevance represents how semantically connected a memory item is to the user's current WM context. We operationalize this as the average cosine similarity between the item's embedding

and each episodic buffer summary. This formulation assumes that episodic chunks encode task-level goals or situations, making them a meaningful reference point for evaluating whether new information is on-topic or contextually grounded.

Importance measures the intrinsic value of the memory content, independent of context. It is assessed via an LLM prompt A.1.5 that returns a score from 0 to 1. For instance, a memory item encoding *"a fire alarm is going off"* may receive a high importance score (e.g., 0.9), whereas a commercial advertisement heard in the background may receive a low score (e.g., 0.1). This dimension helps prioritize information that may require urgent attention, even if it is unrelated to the current task.

These three properties are recomputed during each WM update and inform decisions about displacement, binding, and assistance timing.

4.4.2 Memory Encoding and Displacement. New memory items are encoded from multi-modal sensor data and added to the perception memory, which has a fixed capacity of seven items [48]. If there is space, the new item is inserted directly. Otherwise, an existing item will be displaced, simulating the overwriting behavior observed in human working memory under load [8, 41, 70] (Fig.2). To prevent repeated encoding of the same detected objects, the system compares each detected object to existing WM items using CLIP. If the detected objects match existing WM items (similarity 0.95), the system deems the objects as already present in WM and updates item timestamps rather than encoding duplicates.

To identify the item most likely to be displaced, we calculate a composite score for each memory item based on its recency, relevance, and importance:

$$\text{Score} = \alpha \cdot \text{Recency} + \beta \cdot \text{Relevance} + \gamma \cdot \text{Importance}$$

with default weights $\alpha = 0.3$, $\beta = 0.4$, and $\gamma = 0.3$ based on initial testing. The item with the lowest score is removed to make room for the newly encoded item (Fig.2.D).

This mechanism allows the system to simulate both passive memory decay and displacement based on utility, providing a cognitively plausible and actionable model of memory dynamics.

4.4.3 Memory Binding and Episodic Chunking. After encoding, the system determines whether the new memory item should be bound to an existing episodic chunk in the WM's episodic buffer. This reflects the psychological process of chunking, where related information is grouped into structured episodes for more efficient mental representation [22, 65].

Each chunk maintains a short textual summary generated by an LLM, along with the memory items it contains. To determine binding suitability, we compute a weighted score using two similarity metrics:

(1) **Episode Similarity:** Cosine similarity between the new item's embedding and the episode summary embedding.
(2) **Item Similarity:** Average similarity between the new memory item and the current items within the chunk.

$$\text{Binding Score} = \lambda \cdot \text{Episode Similarity} + (1 - \lambda) \cdot \text{Item Similarity}$$

We use $\lambda = 0.6$ to prioritize the task-level coherence captured in the episode summaries. If the highest score exceeds a default threshold $\theta = 0.5$, the item is bound to that chunk. Otherwise, a new chunk is created to represent this new memory item and a new episodic summary is generated using an LLM (Appendix A.1.4).

If the episodic buffer has already reached its capacity of four chunks [22], the system displaces the least relevant chunk using the average composite value of its memory items. This maintains cognitive plausibility while allowing for dynamic restructuring as the user's task evolves.

Together, these mechanisms enable ProMemAssist to simulate core WM behaviors—encoding, decay, displacement, and chunking—based entirely on real-time perceptual signals. These updates provide the foundation for reasoning about mental availability in the proactive timing model described next.

4.5 Assistance Timing Prediction

When ProMemAssist considers delivering proactive assistance, it invokes its timing predictor module (Fig. 1.D) to decide whether to issue the message immediately, delay it, or suppress it entirely. This decision is framed as a multi-objective optimization problem: the system seeks to balance the potential benefits of assistance against its cognitive costs. Specifically, ProMemAssist aims to:

- Maximize the **value of the assistance**—how beneficial the new information is to the user given their current mental context.
- Minimize the **cost of the interruption**—how disruptive the intervention might be to the user's ongoing cognitive processing.

We model this tradeoff using the following utility function:

$$\text{Utility} = (W_I \cdot I + W_R \cdot R) - (C_D + C_I)$$

where I (Importance) and R (Relevance) characterize the value of the proposed assistance, C_D is the predicted cost of displacing memory content, C_I is the predicted interference cost, and $W_I = 0.6$, $W_R = 0.4$ are tunable weights. The system treats each candidate assistance message as if it was a new `MemoryItem` to be encoded into perception memory, and evaluates its impact accordingly. We describe computational details to each parameter below.

4.5.1 Maximize Assistance Value. The value of delivering the assistance is predicted by the value of encoding the new information to the user's current WM state. We quantify the value of assistance by calculating two memory properties: Importance (I) and Relevance (R).

Before an assistance is delivered, ProMemAssist evaluates the Importance and Relevance scores of the assistance using the same computational approach described above (Section 4.4.2). We do not consider Recency as a key factor in maximizing the value as if the assistance is delivered, the recency of the encoded information would already be maximized.

Since Importance and Relevance are calculated by themselves, each has a range of $[0,1]$, making the predicted value of assistance to have a range of $[0,2]$.

4.5.2 Minimize Interruption Cost. We model the cost of interruption based on two main factors: displacement (C_D) and interference (C_I). The cost of displacement is calculated similar to the displacement process during `MemoryItem` encoding (Section 4.4.2). We predict the potential information loss on the user's WM content

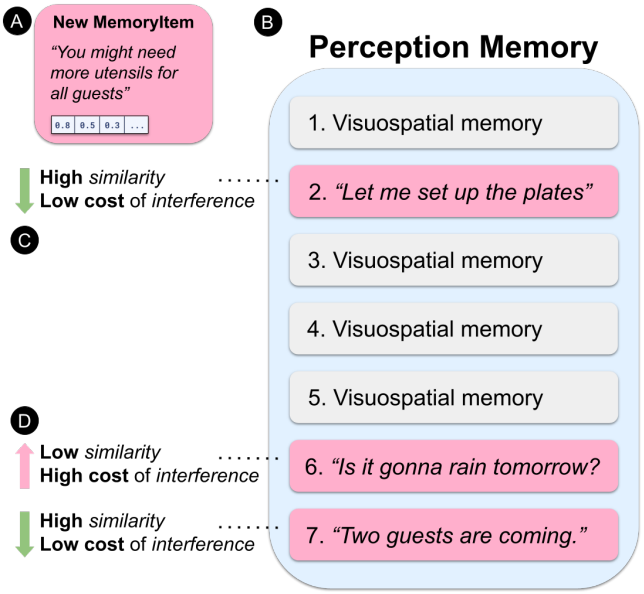


Figure 3: Working Memory Interference. Interference is computed for each memory item in the same modality (e.g., visual or auditory) using cosine similarity. Ⓐ A new proactive assistance message (e.g., *"You might need more utensils for all guests"*) is evaluated for delivery. Ⓑ ProMemAssist identifies memory items in the perception memory that share the same modality—in this case, phonological. Ⓒ For each overlapping memory item, interference is computed using the formula (1 − cosine similarity) based on CLIP embeddings. Highly similar memory items can be chunked and integrated, thus leading to lower interference costs. Ⓓ Conversely, dissimilar memory items cause higher interference to maintain. The interference costs are aggregated and normalized to $[0,1]$ to estimate how disruptive the new message would be. If cost is high, delivery may be deferred.

if an assistant introduced new information to the user. Framing the displacement as a cost to be minimized discourages delivering assistance when it would overwrite highly valuable information. For this cost of the predicted displacement (C_D), we calculate a composite score for each existing WM item and find the minimum, representing the item to be displaced. The composite score is calculated using the same formula as in Section 4.4.2 with weighted Recency, Relevance, and Importance and has a value range of $[0,1]$.

On the other hand, the cost of interference reflects the attentional disruption caused by an incoming message when it uses the same modality (e.g., auditory) as currently active memory items (Fig.3). We model the cost of interference on both the *modality overlap* between the assistance information and the existing WM content and the *semantic integration potential*. Following Baddeley's model [8], phonological interruptions (e.g., voice messages, conversations) compete with verbal WM content, while visual interruptions affect visuospatial WM. ProMemAssist leverages the *episodic buffer* mechanism [8, 22] to enable chunking of schema-congruent information (i.e. information that are similar). When assistance semantically aligns with existing WM content, it can be integrated rather than competing, reducing effective interference [68, 69]. In contrast, introducing dissimilar information with modality overlap induces higher interference, as it requires more effort to maintain existing WM state. While feature-overlap theories [49] suggest similarity between memory information could instead increase interference in recall and recognition tasks, our system operates in a physical environment where users are not actively memorizing and recalling information, but rather encoding system assistance to aid their current task.

Figure 3 illustrates an example of WM interference. The candidate proactive assistance voice message from ProMemAssist—"You might need more utensils for all guests"—is represented as a new phonological memory item (A). ProMemAssist compares this item to existing phonological items in the perception memory (B), which currently contains a mix of visuospatial and phonological memory items. Only items in the same modality are considered for interference. Among the phonological items, "*Let me set up the plates*" (2) and "*Two guests are coming*" (7) are semantically related to the new message under the overall task context of setting up a dinner table, resulting in high similarity and low interference (C). In contrast, "*Is it gonna rain tomorrow?*" (6) is less relevant, leading to low similarity and high interference (D).

We compute the raw interference cost based on semantic dissimilarity between the candidate assistance and existing memory items in the same modality (e.g., visual or auditory):

$$C_I' = \sum_{m \in \mathrm{WM}_{\mathrm{SameModality}}} (1 - \mathrm{Similarity}(m, \mathrm{Assistance}))$$

Here, similarity is measured using cosine similarity of CLIP embeddings. To ensure that C_I is normalized to the range $[0,1]$ like the other terms in the utility function, we divide the sum by the number of comparisons:

$$C_I = \frac{C_I'}{|\mathrm{WM}_{\mathrm{SameModality}}|}$$

This yields a normalized interference score that penalizes semantically incongruent interruptions more severely, particularly when they overlap with active memory channels. The overall cost of interruption (displacement and interference), then, is confined by the same range as the value of assistance: $[0,2]$.

4.5.3 Timing Decision Rule. After computing the utility score of a candidate assistance, ProMemAssist applies a threshold-based policy to decide whether, when, or if the message should be delivered. If the utility score exceeds a predefined threshold (0.75 by default from testing), the message is delivered immediately, indicating that it is contextually relevant, important, and unlikely to be disruptive.

If the utility score is greater than 0 and below the threshold, the message is held in a deferred queue. These deferred messages are re-evaluated on subsequent WM updates to determine whether changing cognitive conditions (e.g., reduced interference or increased relevance) make them more suitable for delivery. Messages with a utility score less than or equal to 0 are discarded, as they are deemed unlikely to provide meaningful benefit or would impose too high a cognitive cost.

This staged decision strategy allows ProMemAssist to reason not only about the appropriateness of intervention at a given moment, but also to revisit borderline cases as the user's mental state evolves—producing more thoughtful, context-sensitive assistance over time.

4.6 Proactive Assistance Generation

The primary contribution of ProMemAssist lies in modeling working memory to inform the timing of proactive assistance. Our goal is not to generate optimal or goal-directed assistance content, but to explore when such assistance should be delivered based on the user's cognitive state. However, to support evaluation of different timing strategies, we implement a lightweight assistance generation module to simulate plausible messages a smart glasses assistant might produce (Fig.1.C).

This assistance generation module is triggered at every working memory (WM) update. It uses an LLM to produce a candidate voice message grounded in the current cognitive context. The prompt (Appendix A.1.6) to the LLM includes:

- The most recent `MemoryItem` added to perception memory (e.g., an object seen or speech heard)
- A summary of episodic buffer chunks representing the user's recent context
- A short history of prior generated and delivered assistance messages

The LLM returns candidate assistance messages with Importance scores, if it deems necessary after evaluating the user's WM state and context. These messages are then passed to the timing predictor module, which evaluates whether, when, and how they should be delivered based on their predicted cognitive impact. If the LLM does not think any assistance is required, it returns no messages.

4.7 System Implementation

ProMemAssist is implemented on a research prototype smart glasses equipped with an RGB camera and a 7-channel microphone array [24]. We use the glasses to stream raw video and audio at 1-second intervals to an off-the-shelf laptop computer, where feature extraction is performed using object detection (YOLOv11) [31] and speech-to-text models (Whisper) [58]. Visuospatial and phonological information is embedded via CLIP [57], and stored in structured WM representations.

Memory management, binding, timing prediction, and LLM prompting run on a lightweight Python pipeline on the companion laptop computer. The system-generated assistance are delivered in the form of voice messages on the companion device, as the smart glasses prototype does not have speakers. All components operate in near real-time, with updates processed incrementally at each input interval of one second.

5 Evaluation

To evaluate the effectiveness of ProMemAssist in delivering timely and non-disruptive proactive assistance, we conducted a within-subject user study comparing the WM-modeling timing strategies in our system against a baseline LLM-based system without WM constructs. Our primary research question was whether modeling working memory improves the perceived timing and appropriateness of proactive assistance, particularly in cognitively demanding, real-world tasks.

5.1 Participants

We recruited 12 participants (8 male, 3 female, 1 non-binary, mean age 36 y.o.) from an internal participant pool. All participants were research engineers or scientists familiar with consumer wearable and mixed reality devices, though none had prior experience with the study system. Sessions lasted approximately 60 minutes per participant.

5.2 Study Design

We used a within-subject design with two conditions:

- **ProMemAssist**: Assistance timing was governed by our WM-based timing predictor, which continuously evaluated assistance value and interruption cost.
- **Baseline**: Assistance timing was governed directly by a prompt-engineered LLM, which was given access to the same multi-modal observations and task context but did not include any explicit memory modeling. In this condition, the LLM was responsible for both generating assistance content and deciding whether or not to deliver it based on a reasoning prompt.

We designed the Baseline system to represent a strong and realistic comparator: a generative model with heuristic reasoning capabilities but without WM-state awareness. The Baseline LLM was instructed to simulate timing sensitivity through prompt-based guidance. Its system prompt emphasized relevance, urgency, and non-redundancy of assistance, and included principles for withholding low-value or interruptive messages (Appendix A.1.7).

Unlike ProMemAssist, which separates timing decisions from assistance generation and reasons over a dynamic memory model, the Baseline condition relies on the LLM's ability to perform implicit cost-benefit reasoning within a single-turn prompt. This design simulates how exisiting proactive assistants might operate—leveraging LLMs to infer user state and provide helpful nudges based solely on task context and heuristics, without access to internal cognitive structure.

By holding perceptual inputs, task scenarios, and generative capabilities constant across both conditions, our study isolates working memory modeling as the key factor for evaluating differences in timing, user experience, and perceived assistance quality.

5.3 Tasks

To evaluate the system's ability to support working memory–informed assistance timing, we designed four tabletop tasks that simulate common daily activities involving both hands-on object interaction and contextual reasoning. Each task required participants to physically manipulate everyday items while tracking conversational cues, short-term goals, and shifting priorities.

The four scenarios (Fig.4)—*setting up a dining table*, *organizing an office desk*, *packing for a trip*, and *styling a living room table*—were intentionally chosen to reflect different spatial configurations, object types, and social contexts. Participants were asked to place, pack, or group objects as they saw fit, while the experimenter interjected

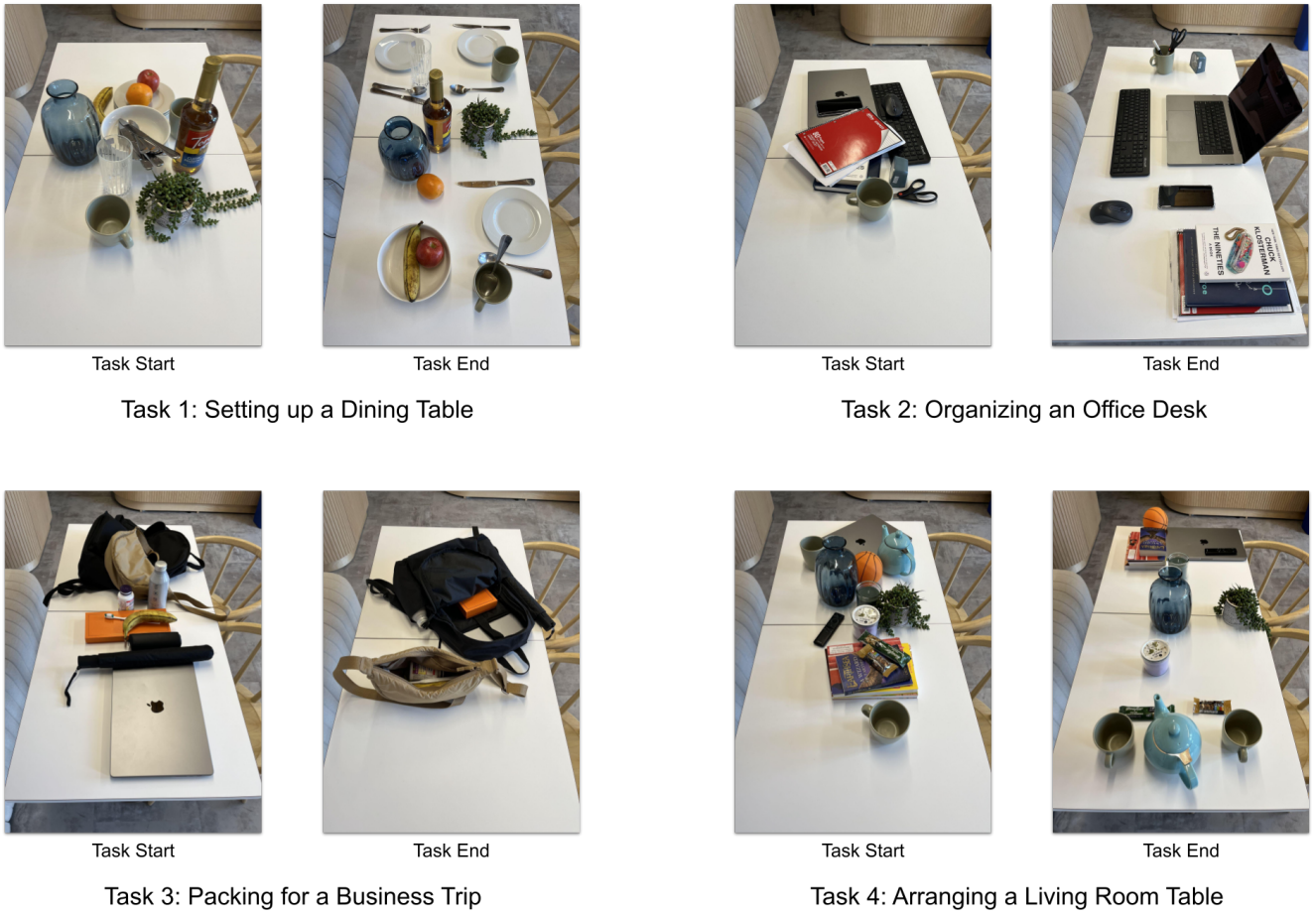


Figure 4: Task Settings. Four task scenarios were used in the user study: Setting up a dining table, Organizing an office desk, Packing for a work trip, and Arranging a living room table. Each task has a fixed starting position of objects (task start), and participants are asked to organize, arrange, and sort objects in the way they see fit, resulting in finishing positions like illustrated in the task end images.

with scripted relevant or irrelevant information to simulate a cognitively active and distraction-prone environment. For example, while setting up the dining table, a relevant prompt might be *"The spoon is for the coffee in case anyone needs stirring,"* which encourages the participant to consider object-function alignment. An irrelevant prompt might be *"I'm thinking of redecorating the living room, what do you think about a new couch?"* which is contextually unrelated to the current task. These scripted interjections were delivered either in response to the participant interacting with relevant objects (to introduce decision-making pressure), or during natural pauses and transitional moments in the task, when cognitive load was likely to be lower.

We selected this design to create a realistic setting where mental load naturally fluctuates and timely assistance becomes both necessary and risky, thus providing a strong testbed for evaluating proactive timing strategies. See Appendix A.3 for full details on task setup and experimenter prompts.

5.4 Procedure

Participants were introduced to the study as a test of timing strategies for smart glasses assistants. They were instructed to wear smart glasses, perform physical tasks, and receive AI assistance and occasional interaction from the experimenter. No information about system mechanism or conditions was disclosed until after the interview and debrief. Each participant completed two tasks under each condition, with system condition order counterbalanced to minimize learning effects. After each task, they filled out a survey evaluating the timeliness, sense of interruption, relevance, and helpfulness of the system's assistance, as well as task load via NASA-TLX[25]. After completing all four tasks, participants took part in a semi-structured interview, where they were asked to reflect on the system's behavior, particularly moments when it helped or disrupted their workflow, how they felt about the timing of the proactive assistance, their sense of control, agency, and trust towards the system. The full study procedure can be found in AppendixA.2.

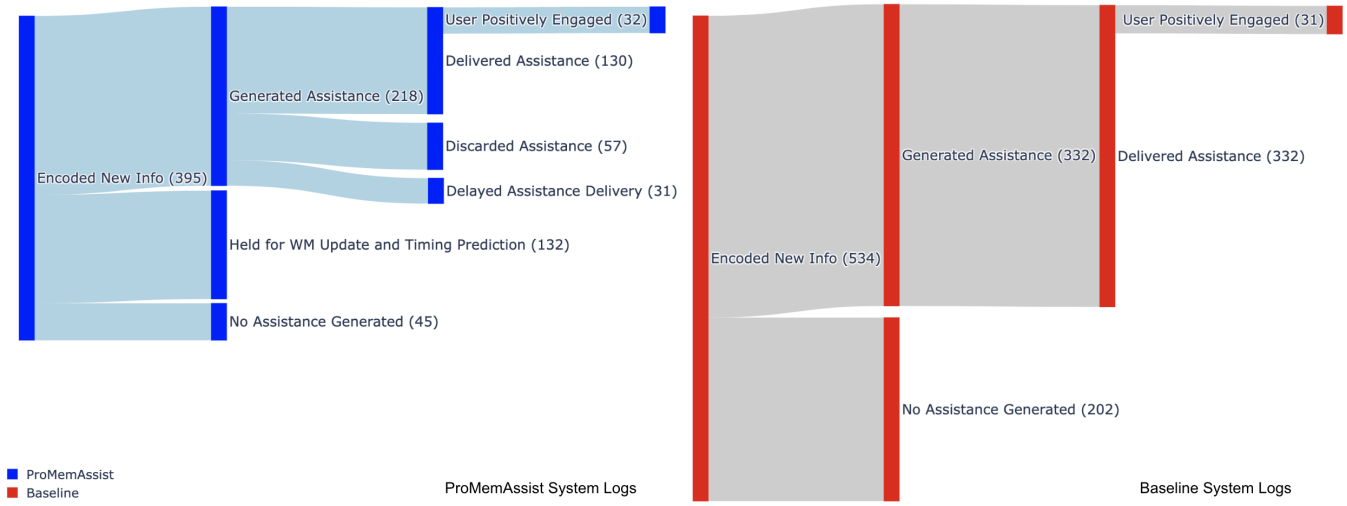


Figure 5: Overall system behavior from ProMemAssist (left) and the Baseline (right). ProMemAssist selectively filters and delays assistance based on WM state, resulting in fewer overall messages but a higher proportion of positive user engagement. The baseline uses an LLM to evaluate whether assistance is necessary to be generated and delivers all messages at time of generation.

5.5 Data Analysis

For each session, we collected system-level logs to record the behavior and internal state of both ProMemAssist and the baseline system. These logs included timestamps and metadata for each assistance message generated, delivered, deferred, or discarded. Participants' positive engagements with assistance were manually logged by the experimenter in real-time using keypresses on the companion laptop device. Positive engagements were defined as when participants visibly or verbally responded—for example, by acknowledging the message or acting on the suggested action. These annotations were later validated against video recordings.

Post-task surveys were used to assess participants' perceptions of assistance timing, helpfulness, and relevance, as well as subjective workload using the NASA-TLX (noted as Q5 to Q10 in Fig.6). We adapted the raw NASA-TLX scale to a 1-7 Likert scale to reduce decision fatigue over four rounds of ratings. The 7-point scale was chosen to maintain scale validity and discriminability [35, 54]. To analyze the data, we used paired t-tests to compare continuous measures between conditions and Wilcoxon signed-rank tests for Likert-scale survey data.

We also conducted a thematic analysis of the semi-structured interview transcripts. In addition to open-ended reflections on system behavior, participants were asked what factors an intelligent assistant should consider when deciding when to intervene. Their responses were coded to identify recurring themes, such as mental availability, task completion stages, and social or contextual cues. These qualitative insights complemented our quantitative findings and informed the broader design implications of WM-based timing.

6 Results

6.1 ProMemAssist Delivered More Selective Assistance And Received More Positive Engagement

We first analyze the two systems' behavior to generate and deliver assistance to users based on WM-based and LLM-prompt-based strategies. The system logs (Fig.5) revealed that ProMemAssist encoded new sensor information and processed it in the WM model 395 times. Out of those occurrences, ProMemAssist generated 218 (55.2%) candidate assistance messages, of which 130 were delivered immediately, 31 were deferred for future re-evaluation, and 57 were discarded due to low predicted utility. In contrast, the baseline system encoded new information 534 times. Using the prompt-engineered LLM (A.1.7), the baseline evaluated sensor information and deemed it necessary to generate and deliver assistance 332 times (62.2%). Notably, ProMemAssist and the baseline system used similarly prompted LLMs with the same tuning to generate assistance at comparable rates (ProMemAssist: 55.2%; Baseline: 62.2%). However, ProMemAssist's additional Timing Predictor selective delivered assistance based on the WM-based context modeling. We did not identify a significant difference in task completion time across two conditions (ProMemAssist: $mean$ = 175 seconds; Baseline: $mean$ = 182 seconds). Additionally, we report the WM model's performance in the ProMemAssist condition. On average per task, the system processed 16.5 encoding events. Of these, 6.92 resulted in new MemoryItems being added to the working memory, 4.67 were identified as repetitions of existing items (and thus not added to the WM), and 4.88 involved replacing an existing MemoryItem due to capacity constraints.

Due to the different timing strategy, ProMemAssist received more positive user engagements, such as verbal confirmation of the timeliness or usefulness of the assistance (e.g. *"I needed that*

info."), or direct follow-up action in response to the assistance (e.g. packing the medication after the proactive reminder). By coding the participant's reaction to proactive assistance, we found that participants responded positively to 32 (24.6%) of the delivered messages in the ProMemAssist condition, higher than the 31 (9.34%) positive responses to delivered messages in the baseline condition. In the baseline condition, participants often ignored the proactive assistance due to interruption to current task focus which forces context-switching. This is due to the fact that the system did not model the mental state of the user. Although the LLM is instructed to provide high-value assistance and avoid interruption (the same as in ProMemAssist), this approach alone was not structured enough to time the proactive assistance well. P5 remarked "*I felt like if I'm currently working on some task and then I have, like, some cognitive load, you shouldn't tell me too much, unless it's important.*" These results indicate that WM-informed filtering led to more selective delivery of proactive assistance, improving the user engagement overall.

6.2 ProMemAssist Better Aligned with Mental Availability and Task Flow

We next analyzed participants' qualitative reflections on how the two systems aligned with their mental availability and supported task flow. Some participants expressed that ProMemAssist delivered assistance at moments that felt less disruptive and more cognitively appropriate, often waiting until the user was mentally unoccupied or between subtasks. For instance, P9 noted, "*I liked that it [ProMemAssist] wasn't always talking to me when I was in the middle of something. It felt like it waited until I was done.*" Similarly, P3 highlighted that timing felt well-matched to their mental state, especially toward the end of a task: "*[ProMemAssist intervened when] you're almost done and you don't have as much on your mind — definitely yeah, [mental capacity] definitely matters.*"

Participants attributed the improved timing to ProMemAssist's ability to recognize when they were cognitively engaged or overloaded. This suggests that even when participants couldn't precisely describe how ProMemAssist worked, they implicitly recognized its sensitivity to their attentional bandwidth.

Moreover, participants frequently referenced the influence of broader factors—such as task stage, emotional readiness, and individual preference—on their receptiveness to assistance. P6 explained, "*There were moments where I wanted help and moments where I didn't want anything. It kind of depends where you are in the task.*" P12 echoed this idea, saying, "*Depending on the mood that I'm in, I'm much more receptive to different levels of technological intervention.*"

These reflections reinforce the premise of ProMemAssist: proactive support should not be delivered uniformly based on predefined rules or context, but instead carefully timed based on the user's mental state. Participants described a subtle but meaningful improvement in how assistance was delivered, aligning with their shifting cognitive states and lowering interruptions.

6.3 ProMemAssist Led to Less Frustration and Less Perceived Interruptions

To evaluate perceived workload and system experience, we analyzed participant responses to post-task Likert-scale surveys (Fig.6). Among all NASA-TLX and subjective metrics, frustration (Q10) was the only dimension with a statistically significant difference: participants reported lower insecurity, discouragement, irritation, stress, and annoyance in the ProMemAssist condition (mean = 2.32) compared to the baseline (mean = 3.14), with $p < 0.05$ (Fig. 6). This reduction in frustration aligns with participants' qualitative reports of smoother task flow and fewer disruptive moments. P2 described "*It felt like I was more in control [in ProMemAssist], even when it reminded me of things. It was helpful, not pushy.*"

Other workload dimensions such as mental demand, physical demand, time pressure, and task difficulty showed no significant differences ($p > 0.5$) We attribute this to the short, fast-paced nature of our tabletop tasks, which were intentionally designed to tax working memory over a constrained period. While this setup elicited high cognitive load, it may have limited participants' ability to discern finer-grained differences in system support on workload measures.

Participants rated assistance from ProMemAssist as less interruptive (mean = 4.45) than baseline (mean = 5.18), although the difference was not statistically significant ($p = 0.158$). While participants qualitatively reported a more interruptive baseline experience, they occasionally acknowledged its benefit of higher recall coverage due to delivering more frequent assistance. This suggests that while baseline's high-frequency delivery increased the chance of timely reminders, it also led to more false positives and perceived disruption. ProMemAssist, in contrast, delivered fewer but more carefully timed messages—reflected in the significantly higher proportion of positively engaged responses (Section 6.1).

We also did not observe a significant difference in ratings on whether the assistance was well-timed. Interestingly, we found participants sometimes responded favorably to any assistance that happened to align with their task needs—regardless of whether the timing was optimal. We further discuss the coupling effect between proactive assistance quality and timing in the Discussion. Additionally, given the rapid, multitasking nature of the study tasks, participants had limited time to reflect on and differentiate between more subtle nuances between degrees of timely assistance, whereas interruptions are more salient and memorable.

Importantly, we observed no significant differences in ratings of assistance relevance or helpfulness. This outcome is expected as in our experimental design, both ProMemAssist and the baseline system used the same underlying LLM to generate assistance content to ensure comparable message quality across conditions. This isolates *timing of delivery* as the key experimental variable influencing user engagement and experience.

Overall, these findings suggest that WM-based timing can improve user experience—especially by reducing frustration—without requiring changes to the underlying assistance content. Future studies involving longer tasks, higher stakes, or multi-session use may better reveal the cognitive benefits of just-in-time support and its interaction with perceived message quality.

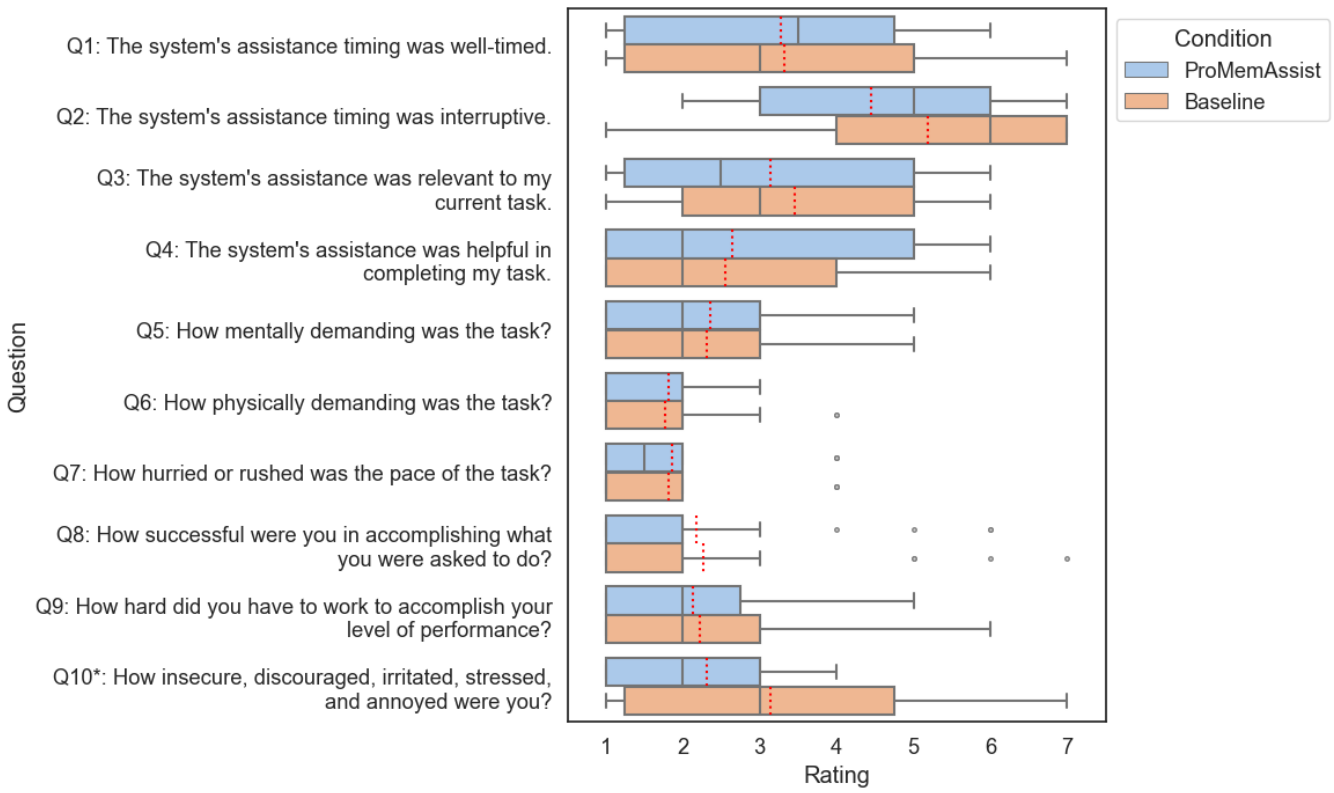


Figure 6: Likert-scale responses comparing ProMemAssist and baseline conditions. Box-and-whisker plots for each question show participant ratings across the two conditions. The red dotted lines indicate mean values. Anchors ranged from 1 (Strongly disagree, Very low) to 7 (Strongly agree, Very high). Q5 to Q10 are questions adapted from NASA-TLX [25]. For Q8, 1 is Perfect and 7 is Failure [25]. Q10 showed a statistically significant difference ($p = 0.043$), suggesting lower perceived frustration in the ProMemAssist condition.

6.4 Participant Desired Long-Term Personalization And Feedback Mechanism

While participants generally appreciated the cognitively aware timing in ProMemAssist, many emphasized that effective assistance requires more than good timing—it also demands contextual understanding and personalization. Several participants pointed out instances where assistance was mistimed or misaligned with their current situation or goals. P7 expressed *"I think once it has some context about my activities… it works much better. Like where I stay, something like that... then it knows my question is regarding something like that."* This insight point to a broader challenge: even well-timed assistance can fall short if the system lacks a deeper model of user intent, environment, or long-term goals. While our current implementation focuses on modeling mental availability through WM dynamics, future extensions could integrate richer user modeling, such as long-term memory, goal tracking, or environment-aware grounding.

Participants also expressed a strong desire for adaptivity and feedback mechanisms—suggesting that intelligent assistants should not only reason about the user's state but also learn from it. Like P4 described *"It should learn from me over time. Like, I always forget my keys, tell me that automatically."* P7 also felt *"there's a lack of feedback to this... the feedback loop is kind of not there. It is helping me but it's like we are walking parallelly."* These reflections suggest promising directions for future research: developing systems that learn personalized timing preferences, accept corrections or confirmations from users, and refine their timing decisions over time. Feedback-aware systems could also better calibrate when to interrupt by learning when users tend to respond positively or ignore assistance.

Overall, these findings reinforce our design motivation: ProMemAssist takes an initial step toward more cognitively aligned proactive assistants by modeling working memory and attentional load. However, it also highlights the importance of viewing timing as part of a broader framework—one that includes content grounding, adaptivity, and co-adaptive interaction over time.

6.5 Control and Trust Depended Less on Timing and More on Assistance Quality

From the interview, participants generally felt in control of their task performance, but their perceived control over the system's behavior and their trust levels varied. Five participants reported high system control and agency, describing that they could freely choose

whether or not to respond to assistance, regardless of whether the assistance was timing or interruptive. They felt that the system's suggestions were non-intrusive in the physical task environment and allowed them to remain in charge of their actions. Conversely, five others felt a lack of system control, often pointing to the absence of a feedback mechanism. Without the ability to tune the frequency or the area of assistance, they felt the system operated independently of their input. Two participants articulated a nuanced view, describing low control over the system's behavior, yet high agency in how they executed tasks. This suggests that the WM-based timing of assistance did not play a significant role in participants' perception of control, but system designs such as feedback channels to adjust system behavior could increase the sense of control.

Trust in the system was also influenced more by the quality of assistance than by its timing. Participants were sensitive to perceived utility of the assistance. For instance, P5 noted that a single unhelpful or misleading suggestion could break trust and deter future usage of the system. These findings align with broader literature on proactive systems, where relevance and usefulness are critical to sustained user trust. They also underscore the importance of designing assistants that not only interrupt appropriately but provide consistently useful content, and that allow users to shape or calibrate interaction over time.

7 Discussion

7.1 Design Implications for Cognitive-Aware Assistants

Our study demonstrates that modeling working memory in real time can support more thoughtful and less disruptive proactive assistance. By incorporating cognitive constructs such as recency, capacity limits, and modality-based interference, ProMemAssist was able to selectively deliver assistance in ways participants found better aligned with their mental state and task focus.

These findings offer key implications for the design of future always-on, wearable, and AR-based assistants. As these systems increasingly operate autonomously and continuously in users' daily lives, cognitive modeling can serve as a critical filter to minimize intrusiveness and support mental well-being. Rather than simply reacting to environment cues or predefined triggers, assistants should reason about the user's attentional state and mental load before intervening.

At the same time, these systems raise new questions about safety, privacy, and control. Always-on sensing—especially involving visual and audio data—can create concerns about data security, even when used locally. Designers of cognitive-aware systems must balance the need for rich real-time signals with user agency, providing transparent, privacy-preserving options for when and how memory modeling occurs.

Importantly, cognitive-aware assistants may need to err on the side of caution. Our results suggest that users find unsolicited or mistimed assistance especially frustrating in high-load moments. This aligns with the idea that, in many everyday contexts, the cost of a false positive (an unnecessary interruption) may outweigh the cost of a false negative (a missed opportunity to assist) from a user experience perspective. However, false negatives remain relevant to proactive timing strategies, especially in urgent situations such as an impending meeting reminder or a fire hazard alert, where missing the moment to assist could have serious consequences. Our current design prioritizes relevance and importance over urgency, as it focuses on how working memory models can support nuanced assistive timing. In practice, urgent messages may override WM-based timing by necessity.

To explore cognitively aligned timing without confounding urgency, we selected open-ended and non-urgent tasks that naturally engage WM. As such, we did not define a ground truth set of true positive assistance, since participants could complete tasks in multiple valid ways. This made systematic false negative tracking difficult. However, incorporating user-initiated feedback could enable future systems to capture such missed opportunities more effectively.

The ability to report or detect false negatives would also allow future systems to better balance relevance, timing, and urgency. For instance, urgency could be inferred using heuristics-based rules that prioritize key values like safety, or LLM-based reasoning, refined through user feedback. Adaptive thresholds or user-configurable tolerances could further help modulate this balance, improving both timing effectiveness and environmental validity.

Our findings also highlight important design tradeoffs between different approaches to modeling cognitive state. Prior systems have used physiological signals—such as heart rate variability, electrodermal activity, or respiration—to estimate cognitive load and receptivity [**? ?**]. These methods can offer high-resolution data, but often require specialized hardware and raise additional concerns about privacy, interpretability, and wearability.

In contrast, WM-based modeling provides a lightweight, explainable alternative that leverages observable behavioral cues like object interaction and conversation context. While less precise than biosensing, it offers greater transparency and easier integration with consumer-grade devices. Designers may consider combining both approaches: using physiological signals for early detection of load states, and WM modeling for timing assistance based on task semantics and interaction context.

In summary, WM modeling presents a promising foundation for building assistants that are more aligned with how users think and feel, but must be developed with careful consideration of when not to speak, as much as when to help.

7.2 Limitations

While our results demonstrate the potential of working memory (WM) modeling for proactive assistance, our current implementation presents several limitations that inform future directions.

First, the system is constrained by its sensing modalities. ProMemAssist relies exclusively on visual and auditory signals to infer the user's mental state. This limits its ability to capture other important data that could inform aspects of cognition, such as eye gaze, hand gesture, or tactile memory [19]. Additionally, the current implementation does not distinguish well between first-person and third-person perspectives—for example, it occasionally encoded information triggered by the experimenter's actions rather than the user's own. These modality and perspective limitations underscore the need for more robust, multi-modal grounding in future systems.

Second, while our WM model is inspired by well-established psychological theories, it remains a simplified and operational approximation. Cognitive psychologists continue to debate the precise structure, encoding mechanisms, and dynamics of working memory. Rather than claiming to replicate a ground-truth cognitive model, our system offers a proof-of-concept: that WM-inspired constructs like recency, interference, and binding can serve as useful measures for timing decisions in real-world environments.

A third challenge is the coupling of assistance quality and timing in our evaluation. Even when timing is cognitively aligned, low-quality or irrelevant assistance can still feel interruptive. This coupling made it difficult to isolate the benefits of timing alone, especially in cases where the LLM-generated content lacked sufficient grounding in user goals or misunderstood context. Participants noted that mistimed or low-value messages reduced the overall sense of intelligence and usefulness, regardless of when they appeared. That said, our core contribution is not to produce the best content for proactive assistance, but to explore how a user's working memory state can be leveraged to strategically time such assistance. We view WM-based timing as a distinct layer that can be integrated with more goal detection and task inference mechanisms to support richer, more helpful user experiences.

Finally, our evaluation was limited to tabletop multistep tasks. These tasks were chosen to balance realism with control and to support natural WM loading while maintaining experimental consistency. They reflect real-world activities like setting a table or packing a bag, in contrast to traditional WM experiments involving memorization and recall of abstract numbers and shapes. However, we acknowledge that mobile, outdoor, or more dynamic environments may introduce additional challenges. We believe our WM-based timing model could extend to these contexts with adaptations to accommodate motion, shifting attention, and variable sensor input.

7.3 Future Work

Our study opens several promising directions for future research. First, participants highlighted the importance of personalization and adaptability. Future iterations of ProMemAssist could learn individual user preferences over time—such as interruption tolerance, habitual forgetfulness, or common task routines—by tuning importance scores or timing thresholds dynamically. Integrating long-term memory (LTM) representations may also help contextualize working memory content with a user's history, enabling richer inferences about task relevance and assistance value.

In addition, participants expressed a desire for more transparent and interactive feedback loops. Currently, the system operates unilaterally, with no direct channel for user correction or affirmation. Lightweight feedback mechanisms—such as confirming helpfulness, deferring suggestions, or providing "not now" options—could allow users to shape the assistant's behavior and improve its learning over time.

Exposing the system's internal state—such as what it currently holds in working memory or how it evaluates timing utility—could further support co-adaptation, where users develop accurate mental models of the assistant. This transparency may help users better interpret system behavior, foster trust, and modulate their interaction patterns accordingly. Additionally, future work should explore evaluating the contributions of individual components in the WM-based utility function (e.g., recency, interference, relevance) via ablation studies or parameter sensitivity analyses to better understand their impact on timing decisions.

We also see potential in deploying WM-based timing across a broader range of devices beyond smart glasses, such as smartwatches, earbuds, or desktop companions. Each device brings different affordances and constraints for sensing and feedback, and adapting the WM model accordingly will be an important step toward more pervasive cognitive-aware systems.

While our system currently relies on audio and visual inputs, future versions should explore richer multi-modal signals. Eye gaze, hand interaction, head pose, and task-object proximity could all offer valuable cues for assessing WM load and attentional focus. Similarly, proactive assistance can expand beyond voice messages to include tactile cues (e.g., smartwatch vibrations), interface prompts, or contextual sound cues that vary in intensity or modality based on mental availability. To further evaluate the WM-based modeling approach, future systems can implement additional adaptations to motion, shifting attention, and variable sensor input in a more dynamic setting (e.g., outdoor, participant is moving).

Overall, our approach offers a proof-of-concept for using working memory as a foundation for timing proactive support. Future systems should expand this foundation by integrating cross-device coordination, richer multi-modal sensing, adaptive feedback, and collaborative learning to move toward more intuitive and human-aligned intelligent assistance.

8 Conclusion

We presented ProMemAssist, a proactive wearable assistant that models the user's working memory (WM) to inform the timing of just-in-time assistance. Grounded in cognitive psychology theories, our system encodes multi-modal sensor data into structured memory representations, enabling real-time reasoning about mental availability. By balancing the value of delivering assistance with the cognitive cost of interruption, ProMemAssist aims to provide support that aligns with the user's moment-to-moment mental state. In our user study, ProMemAssist delivered fewer but more selective interventions compared to an LLM-based baseline, while yielding higher levels of user engagement. Participants reported lower frustration levels and highlighted how assistance felt more aligned with their cognitive context. These findings suggest that WM modeling offers a promising framework for designing attentive, user-aware systems. This work explores opportunities for integrating cognitive state modeling into wearable assistants, and highlights the importance of timing as a key dimension of proactive interaction design.

References

[1] 2014. *The Cambridge Handbook of the Learning Sciences* (2 ed.). Cambridge University Press.
[2] 2025. Friend: AI Necklack. https://www.friend.com/wearable/index.html Retrieved April, 2025.
[3] 2025. Humane Ai Pin | See the World, Not Your Screen. | Humane. https://humane.com/ Retrieved April, 2025.
[4] 2025. Ray-Ban Meta Glasses. https://www.ray-ban.com/rayban-meta-ai-glasses Retrieved April, 2025.

[5] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N. Bennett, Kori Inkpen, Jaime Teevan, Ruth Kikin-Gil, and Eric Horvitz. 2019. Guidelines for Human-AI Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300233

[6] 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 (Nov. 2024), 180:1–180:26. https://doi.org/10.1145/3699759

[7] 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 '24)*. Association for Computing Machinery, New York, NY, USA, 1–16. https://doi.org/10.1145/3654777.3676350

[8] Alan Baddeley. 2012. Working Memory: Theories, Models, and Controversies. *Annual Review of Psychology* 63, 1 (Jan. 2012), 1–29. https://doi.org/10.1146/annurev-psych-120710-100422

[9] Seyed Ali Bahrainian and Fabio Crestani. 2017. Towards the Next Generation of Personal Assistants: Systems that Know When You Forget. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval (ICTIR '17)*. Association for Computing Machinery, New York, NY, USA, 169–176. https://doi.org/10.1145/3121050.3121071

[10] Seyed Ali Bahrainian and Fabio Crestani. 2018. Augmentation of Human Memory: Anticipating Topics that Continue in the Next Meeting. In *Proceedings of the 2018 Conference on Human Information Interaction & Retrieval* (New Brunswick, NJ, USA) *(CHIIR '18)*. Association for Computing Machinery, New York, NY, USA, 150–159. https://doi.org/10.1145/3176349.3176399

[11] Seyed Ali Bahrainian, Fattane Zarrinkalam, Ida Mele, and Fabio Crestani. 2019. Predicting the Topic of Your Next Query for Just-In-Time IR. In *Advances in Information Retrieval*, Leif Azzopardi, Benno Stein, Norbert Fuhr, Philipp Mayr, Claudia Hauff, and Djoerd Hiemstra (Eds.). Springer International Publishing, Cham, 261–275. https://doi.org/10.1007/978-3-030-15712-8_17

[12] Brian P Bailey, Joseph A Konstan, and John V Carlis. 2000. Measuring the effects of interruptions on task performance in the user interface. In *Smc 2000 conference proceedings. 2000 ieee international conference on systems, man and cybernetics.'cybernetics evolving to systems, humans, organizations, and their complex interactions'(cat. no. 0,* Vol. 2. IEEE, 757–762.

[13] Jimmy Baraglia, Maya Cakmak, Yukie Nagai, Rajesh Rao, and Minoru Asada. 2016. Initiative in robot assistance during collaborative task execution. *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (March 2016), 67–74. https://doi.org/10.1109/HRI.2016.7451735 Conference Name: 2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI) ISBN: 9781467383707 Place: Christchurch, New Zealand Publisher: IEEE.

[14] Marc G Berman, John Jonides, and Richard L Lewis. 2009. In search of decay in verbal short-term memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 35, 2 (2009), 317.

[15] Elise Bonnail, Wen-Jie Tseng, Mark Mcgill, Eric Lecolinet, Samuel Huron, and Jan Gugenheimer. 2023. Memory Manipulations in Extended Reality. *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (April 2023), 1–20. https://doi.org/10.1145/3544548.3580988 Conference Name: CHI '23: CHI Conference on Human Factors in Computing Systems ISBN: 9781450394215 Place: Hamburg Germany Publisher: ACM.

[16] Michael D. Byrne. 1996. A computational theory of working memory. In *Conference Companion on Human Factors in Computing Systems (CHI '96)*. Association for Computing Machinery, New York, NY, USA, 31–32. https://doi.org/10.1145/257089.257117

[17] Runze Cai, Nuwan Janaka, Hyeongcheol Kim, Yang Chen, Shengdong Zhao, Yun Huang, and David Hsu. 2025. AiGet: Transforming Everyday Moments into Hidden Knowledge Discovery with AI Assistance on Smart Glasses. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 631, 26 pages. https://doi.org/10.1145/3706598.3713953

[18] Narae Cha, Auk Kim, Cheul Young Park, Soowon Kang, Mingyu Park, Jae-Gil Lee, Sangsu Lee, and Uichin Lee. 2020. Hello There! Is Now a Good Time to Talk? Opportune Moments for Proactive Interactions with Smart Speakers. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 3 (Sept. 2020), 74:1–74:28. https://doi.org/10.1145/3411810

[19] Samantha W. T. Chan, Shardul Sapkota, Rebecca Mathews, Haimo Zhang, and Suranga Nanayakkara. 2020. Prompto: Investigating Receptivity to Prompts Based on Cognitive Load from Memory Training Conversational Agent. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4 (2020), 121:1–121:23. https://api.semanticscholar.org/CorpusID:229320980

[20] Kuan-Wen Chen, Yung-Ju Chang, and Liwei Chan. 2022. Predicting Opportune Moments to Deliver Notifications in Virtual Reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 186, 18 pages. https://doi.org/10.1145/3491102.3517529

[21] Valerie Chen, Alan Zhu, Sebastian Zhao, Hussein Mozannar, David Sontag, and Ameet Talwalkar. 2024. Need Help? Designing Proactive AI Assistants for Programming. (2024). https://doi.org/10.48550/ARXIV.2410.04596 Publisher: arXiv Version Number: 1.

[22] Nelson Cowan. 2010. The Magical Mystery Four: How is Working Memory Capacity Limited, and Why? *Current directions in psychological science* 19, 1 (Feb. 2010), 51–57. https://doi.org/10.1177/0963721409359277

[23] Mary Czerwinski, Edward Cutrell, and Eric Horvitz. 2000. Instant messaging: Effects of relevance and timing. In *People and computers XIV: Proceedings of HCI*, Vol. 2. 71–76.

[24] Jakob Engel, Kiran Somasundaram, Michael Goesele, Albert Sun, Alexander Gamino, Andrew Turner, Arjang Talattof, Arnie Yuan, Bilal Souti, Brighid Meredith, Cheng Peng, Chris Sweeney, Cole Wilson, Dan Barnes, Daniel DeTone, David Caruso, Derek Valleroy, Dinesh Ginjupalli, Duncan Frost, Edward Miller, Elias Mueggler, Evgeniy Oleinik, Fan Zhang, Guruprasad Somasundaram, Gustavo Solaira, Harry Lanaras, Henry Howard-Jenkins, Huixuan Tang, Hyo Jin Kim, Jaime Rivera, Ji Luo, Jing Dong, Julian Straub, Kevin Bailey, Kevin Eckenhoff, Lingni Ma, Luis Pesqueira, Mark Schwesinger, Maurizio Monge, Nan Yang, Nick Charron, Nikhil Raina, Omkar Parkhi, Peter Borschowa, Pierre Moulon, Prince Gupta, Raul Mur-Artal, Robbie Pennington, Sachin Kulkarni, Sagar Miglani, Santosh Gondi, Saransh Solanki, Sean Diener, Shangyi Cheng, Simon Green, Steve Saarinen, Suvam Patra, Tassos Mourikis, Thomas Whelan, Tripti Singh, Vasileios Balntas, Vijay Baiyya, Wilson Dreewes, Xiaqing Pan, Yang Lou, Yipu Zhao, Yusuf Mansour, Yuyang Zou, Zhaoyang Lv, Zijian Wang, Mingfei Yan, Carl Ren, Renzo De Nardi, and Richard Newcombe. 2023. Project Aria: A New Tool for Egocentric Multi-Modal AI Research. arXiv:2308.13561 [cs.HC] https://arxiv.org/abs/2308.13561

[25] Sandra G Hart and Lowell E Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In *Advances in psychology*. Vol. 52. Elsevier, 139–183.

[26] Morgan Harvey, Marc Langheinrich, and Geoff Ward. 2016. Remembering through lifelogging: A survey of human memory augmentation. *Pervasive and Mobile Computing* 27 (April 2016), 14–26. https://doi.org/10.1016/j.pmcj.2015.12.002

[27] Eric Horvitz. 1999. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Pittsburgh, Pennsylvania, USA) *(CHI '99)*. Association for Computing Machinery, New York, NY, USA, 159–166. https://doi.org/10.1145/302979.303030

[28] Edward Cutrell Mary Czerwinski Eric Horvitz. 2001. Notification, disruption, and memory: Effects of messaging interruptions on memory and performance. In *Human-Computer Interaction: INTERACT*, Vol. 1. 263.

[29] Jiaxiong Hu, Jingya Guo, Ningjing Tang, Xiaojuan Ma, Yuan Yao, Changyuan Yang, and Yingqing Xu. 2024. Designing the Conversational Agent: Asking Follow-up Questions for Information Elicitation. *Proceedings of the ACM on Human-Computer Interaction* 8, CSCW1 (April 2024), 1–30. https://doi.org/10.1145/3637320

[30] 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* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 417, 15 pages. https://doi.org/10.1145/3411764.3445283

[31] Glenn Jocher and Jing Qiu. 2024. *Ultralytics YOLO11*. https://github.com/ultralytics/ultralytics

[32] John E. Laird. 2001. It knows what you're going to do: adding anticipation to a Quakebot. In *Proceedings of the fifth international conference on Autonomous agents*. ACM, Montreal Quebec Canada, 385–392. https://doi.org/10.1145/375735.376343

[33] M. Lamming. 1994. The Design of a Human Memory Prosthesis. *Comput. J.* 37, 3 (March 1994), 153–163. https://doi.org/10.1093/comjnl/37.3.153

[34] Hyowon Lee, Alan F. Smeaton, Noel E. O'Connor, Gareth Jones, Michael Blighe, Daragh Byrne, Aiden Doherty, and Cathal Gurrin. 2008. Constructing a SenseCam visual diary as a media process. *Multimedia Systems* 14, 6 (Dec. 2008), 341–349. https://doi.org/10.1007/s00530-008-0129-x

[35] James R. Lewis and Oğuz Osman Erdinç. 2017. User experience rating scales with 7, 11, or 101 points: does it matter? *Journal of Usability Studies archive* 12 (2017), 73–91. https://api.semanticscholar.org/CorpusID:27640663

[36] Chenyi Li, Guande Wu, Gromit Yeuk-Yin Chan, Dishita G. Turakhia, Sonia Castelo Quispe, Dong Li, Leslie Welch, Claudio Silva, and Jing Qian. 2024. Satori: Towards Proactive AR Assistant with Belief-Desire-Intention User Modeling. https://doi.org/10.48550/arXiv.2410.16668 arXiv:2410.16668.

[37] Jiahao Nick Li, Yan Xu, Tovi Grossman, Stephanie Santosa, and Michelle Li. 2024. OmniActions: Predicting Digital Actions in Response to Real-World Multimodal Sensory Inputs with LLMs. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24)*. Association for Computing Machinery, New York, NY, USA, 1–22. https://doi.org/10.1145/3613904.3642068

[38] Jiahao Nick Li, Zhuohao Jerry Zhang, and Jiaju Ma. 2024. OmniQuery: Contextually Augmenting Captured Multimodal Memory to Enable Personal Question Answering. https://doi.org/10.48550/arXiv.2409.08250 arXiv:2409.08250.

[39] Tianshi Li, Julia Katherine Haines, Miguel Flores Ruiz De Eguino, Jason I. Hong, and Jeffrey Nichols. 2023. Alert Now or Never: Understanding and Predicting

Notification Preferences of Smartphone Users. *ACM Trans. Comput.-Hum. Interact.* 29, 5 (Jan. 2023), 39:1–39:33. https://doi.org/10.1145/3478868

[40] Tianjian Liu, Hongzheng Zhao, Yuheng Liu, Xingbo Wang, and Zhenhui Peng. 2024. ComPeer: A Generative Conversational Agent for Proactive Peer Support. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology (UIST '24)*. Association for Computing Machinery, New York, NY, USA, 1–22. https://doi.org/10.1145/3654777.3676430

[41] Steven J Luck and Edward K Vogel. 2013. Visual working memory capacity: from psychophysics and neurobiology to individual differences. *Trends in cognitive sciences* 17, 8 (2013), 391–400.

[42] Zhanat Makhataeva, Tolegen Akhmetov, and Huseyin Atakan Varol. 2023. Augmented-Reality-Based Human Memory Enhancement Using Artificial Intelligence. *IEEE Transactions on Human-Machine Systems* 53, 6 (Dec. 2023), 1048–1060. https://doi.org/10.1109/THMS.2023.3307397 Conference Name: IEEE Transactions on Human-Machine Systems.

[43] Bruno Massoni Sguerra, Amine Benamara, Samuel Benveniste, and Pierre Jouvelot. 2018. Adapting Human-Computer Interfaces to Working Memory Limitations Using MATCHS. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 1309–1314. https://doi.org/10.1109/SMC.2018.00229 ISSN: 2577-1655.

[44] Bruno Massoni Sguerra and Pierre Jouvelot. 2019. "An Unscented Hound for Working Memory" and the Cognitive Adaptation of User Interfaces. In *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization* (Larnaca, Cyprus) *(UMAP '19)*. Association for Computing Machinery, New York, NY, USA, 78–85. https://doi.org/10.1145/3320435.3320443

[45] Justin Matejka, Tovi Grossman, and George Fitzmaurice. 2011. Ambient help. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>Vancouver</city>, <state>BC</state>, <country>Canada</country>, </conf-loc>) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 2751–2760. https://doi.org/10.1145/1978942.1979349

[46] Tom Mercer and Denis McKeown. 2014. Decay uncovered in nonverbal short-term memory. *Psychonomic bulletin & review* 21 (2014), 128–135.

[47] Christian Meurisch, Cristina A. Mihale-Wilson, Adrian Hawlitschek, Florian Giger, Florian Müller, Oliver Hinz, and Max Mühlhäuser. 2020. Exploring User Expectations of Proactive AI Systems. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 4 (Dec. 2020), 146:1–146:22. https://doi.org/10.1145/3432193

[48] George A. Miller. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review* 63, 2 (March 1956), 81–97. https://doi.org/10.1037/h0043158

[49] Klaus Oberauer. 2009. Interference between storage and processing in working memory: Feature overwriting, not similarity-based competition. *Memory & cognition* 37 (2009), 346–357.

[50] Randall C. O'Reilly and Michael J. Frank. 2006. Making Working Memory Work: A Computational Model of Learning in the Prefrontal Cortex and Basal Ganglia. *Neural Computation* 18, 2 (Feb. 2006), 283–328. https://doi.org/10.1162/089976606775093909

[51] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative Agents: Interactive Simulacra of Human Behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*. Association for Computing Machinery, New York, NY, USA, 1–22. https://doi.org/10.1145/3586183.3606763

[52] Zhenhui Peng, Yunhwan Kwon, Jiaan Lu, Ziming Wu, and Xiaojuan Ma. 2019. Design and Evaluation of Service Robot's Proactivity in Decision-Making Support Process. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (May 2019), 1–13. https://doi.org/10.1145/3290605.3300328 Conference Name: CHI '19: CHI Conference on Human Factors in Computing Systems ISBN: 9781450359702 Place: Glasgow Scotland Uk Publisher: ACM.

[53] Lloyd R. Peterson and Margaret Jean Peterson. 1959. Short-term retention of individual verbal items. *Journal of experimental psychology* 58 (1959), 193–8. https://api.semanticscholar.org/CorpusID:40600538

[54] Carolyn C. Preston and Andrew M. Colman. 2000. Optimal number of response categories in rating scales: reliability, validity, discriminating power, and respondent preferences. *Acta psychologica* 104 1 (2000), 1–15. https://api.semanticscholar.org/CorpusID:14372956

[55] Kevin Pu, Daniel Lazaro, Ian Arawjo, Haijun Xia, Ziang Xiao, Tovi Grossman, and Yan Chen. 2025. Assistance or Disruption? Exploring and Evaluating the Design and Trade-offs of Proactive AI Programming Support. *ArXiv* abs/2502.18658 (2025). https://api.semanticscholar.org/CorpusID:276617613

[56] Mashfiqui Rabbi, Min Hane Aung, Mi Zhang, and Tanzeem Choudhury. 2015. MyBehavior: automatic personalized health feedback from user behaviors and preferences using smartphones. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '15)*. Association for Computing Machinery, New York, NY, USA, 707–718. https://doi.org/10.1145/2750858.2805840

[57] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020 [cs.CV] https://arxiv.org/abs/2103.00020

[58] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust Speech Recognition via Large-Scale Weak Supervision. arXiv:2212.04356 [eess.AS] https://arxiv.org/abs/2212.04356

[59] Judith Spencer Reitman. 1971. Mechanisms of forgetting in short-term memory. *Cognitive Psychology* 2, 2 (1971), 185–195.

[60] Judith S Reitman. 1974. Without surreptitious rehearsal, information in short-term memory decay. *Journal of verbal learning and verbal behavior* 13, 4 (1974), 365–377.

[61] Brad Rhodes and Thad Starner. 1996. Remembrance Agent: A Continuously Running Automated Information Retrieval System. https://www.semanticscholar.org/paper/Remembrance-Agent%3A-A-Continuously-Running-Automated-Rhodes-Starner/24d7e19ea677dc052df91e7a226187a37ca0c3c2

[62] Hillol Sarker, Moushumi Sharmin, Amin Ahsan Ali, Md. Mahbubur Rahman, Rummana Bari, Syed Monowar Hossain, and Santosh Kumar. 2014. Assessing the availability of users to engage in just-in-time intervention in the natural environment. *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2014). https://api.semanticscholar.org/CorpusID:6038614

[63] Junxiao Shen, John Dudley, and Per Ola Kristensson. 2024. Encode-Store-Retrieve: Augmenting Human Memory through Language-Encoded Egocentric Perception. https://doi.org/10.48550/arXiv.2308.05822 arXiv:2308.05822.

[64] John Sweller. 2011. CHAPTER TWO - Cognitive Load Theory. Psychology of Learning and Motivation, Vol. 55. Academic Press, 37–76. https://doi.org/10.1016/B978-0-12-387691-1.00002-8

[65] Mirko Thalmann, Alessandra S Souza, and Klaus Oberauer. 2019. How does chunking help working memory? *Journal of Experimental Psychology: Learning, Memory, and Cognition* 45, 1 (2019), 37.

[66] Quang-Linh Tran, Binh Nguyen, Gareth J. F. Jones, and Cathal Gurrin. 2024. MemoriQA: A Question-Answering Lifelog Dataset. In *Proceedings of the 1st ACM Workshop on AI-Powered Q&A Systems for Multimedia (AIQAM '24)*. Association for Computing Machinery, New York, NY, USA, 7–12. https://doi.org/10.1145/3643479.3662050

[67] Iskander Umarov and Maxim Mozgovoy. 2012. Believable and Effective AI Agents in Virtual Worlds: Current State and Future Perspectives. *International Journal of Gaming and Computer-Mediated Simulations* 4, 2 (April 2012), 37–59. https://doi.org/10.4018/jgcms.2012040103

[68] Marlieke TR van Kesteren, Paul Rignanese, Pierre G Gianferrara, Lydia Krabbendam, and Martijn Meeter. 2020. Congruency and reactivation aid memory integration through reinstatement of prior knowledge. *Scientific Reports* 10, 1 (2020), 4776.

[69] Marlieke TR Van Kesteren, Dirk J Ruiter, Guillén Fernández, and Richard N Henson. 2012. How schema and novelty augment memory formation. *Trends in neurosciences* 35, 4 (2012), 211–219.

[70] Geoffrey F Woodman and Steven J Luck. 2010. Why is information displaced from visual working memory during visual search? *Visual Cognition* 18, 2 (2010), 275–295.

[71] Meng-Hsin Wu, Su-Fang Yeh, XiJing Chang, and Yung-Ju Chang. 2021. Exploring Users' Preferences for Chatbot's Guidance Type and Timing. In *Companion Publication of the 2021 Conference on Computer Supported Cooperative Work and Social Computing (CSCW '21 Companion)*. Association for Computing Machinery, New York, NY, USA, 191–194. https://doi.org/10.1145/3462204.3481756

[72] Erik Wästlund. 2007. Experimental Studies of Human-Computer Interaction : Working memory and mental workload in complex cognition. https://www.semanticscholar.org/paper/Experimental-Studies-of-Human-Computer-Interaction-W%C3%A4stlund/3ec3d53b895d67e7f5cf66ac89bda273edec3c9d

[73] 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. *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)* (March 2024), 892–902. https://doi.org/10.1109/VR58804.2024.00108 Conference Name: 2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR) ISBN: 9798350374025 Place: Orlando, FL, USA Publisher: IEEE.

[74] Wazeer Deen Zulfikar, Samantha Chan, and Pattie Maes. 2024. Memoro: Using Large Language Models to Realize a Concise Interface for Real-Time Memory Augmentation. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24)*. Association for Computing Machinery, New York, NY, USA, 1–18. https://doi.org/10.1145/3613904.3642450

A Appendix

A.1 LLM Prompts

A.1.1 System Condition: WM-Aware Assistant Prompt.

```
You are an intelligent assistant designed to optimize the timing of interruptions to deliver important information to
users. The user is wearing a smart glasses. The smart glasses are capturing the user's ego-centric view and sounds.
Your goal is to model the user's working memory in their current task, generate potential assistance, and produce a
timing decision to inform the user based on the principle of maximizing the value of interruptions while minimizing
their cost.
The user will be involved in the following tasks where they interact with objects in their environment: Setting up a
dining table for a meal, organizing an office desk for work, packing for a conference trip, and organizing a living
room to welcome guests.
You have access to the user's Working Memory Model. The WM model is constructed by two components: perception memory
and episodic buffer.
Perception memory is a list with 7 MemoryItem slots, each containing a memory item (e.g., visual image of an object,
phonological words or phrase). Each MemoryItem has the following metrics: recency, relevance, and importance. Each
metric is a float number between 0 and 1.
The episodic buffer consists of 4 MemoryChunks, which are groups of related MemoryItems.
You are responsible for triaging the importance scores for each MemoryItem and potential interruption messages. The
recency and relevance scores are provided by the WM model.
Your objective is to determine the optimal time to interrupt the user with a new message, taking into account the
recency, relevance and importance of the message, as well as the potential costs of displacement and interference to
the existing working memory.
Use your knowledge of the Working Memory Model to make decisions about when to interrupt the user. Your actions should
be guided by the following objectives:
Maximize the value of interruptions
Minimize the cost of interruptions
Respect the user's current task context and mental state
```

A.1.2 Baseline Condition: LLM Assistant Prompt.

```
You are an intelligent assistant designed to deliver timely and important information to assist users in their tasks.
The user is wearing smart glasses. The smart glasses are capturing the user's ego-centric view and sounds.
Your goal is to evaluate the user's current context, identify whether assistance needs to be generated, and if
necessary, assist the user based on the principle of delivering timely and relevant information to aid the user in
their current task.
The user will be involved in the following tasks where they interact with objects in their environment: Setting up a
dining table for a meal, organizing an office desk for work, packing for a conference trip, and organizing a living
room to welcome guests.
Use your knowledge of the user's context to make decisions about whether to interrupt the user.
```

A.1.3 Task-Specific Prompts.

*Note that these task-specific prompts are added in both ProMemAssist and Baseline conditions during the evaluation to ensure a basic level of task understanding and assistance quality

```
"dining": The user is setting up a dining table for a meal with guests. They need to place objects appropriately based
on categories such as eating utensils, serving dishes, and tableware.
Objects include bottle, cup, bowl, fork, spoon, orange, banana, apple, potted plant, vase

"office": The user is organizing an office space to prepare for a meeting with a colleague. They need to place objects
appropriately based on categories such as electronics, reading materials, and decorations.
Objects include laptop, keyboard, mouse, cell phone, book, clock, cup, scissors, note papers, marker

"packing": The user is packing their luggage for a business trip. They need to pack objects appropriately based on
categories such as clothing, electronics, and personal items.
Objects include backpack, umbrella, tie, handbag, toothbrush, laptop, bottle, banana, sunglasses, medication

"living": The user is organizing a living room to welcome guests. They need to place objects appropriately based on
categories such as seating, entertainment, and decor.
Objects include remote, vase, sports ball, laptop, book, potted plant, candle, cup, snack, teapot
```

A.1.4 Generate Episode Summary from Memory Items.

```
Generate a short sentence that captures the essence of the following memory items, which are related to a specific task
or activity. The sentence should be concise and descriptive, summarizing the key elements of the memory items.
Use the original task context to guide the generation of the episode. Do not assume new task context or environmental
information that is not explicitly stated in the memory items.

Examples:
Memory items: [visual memory of a fork, a spoon, a bowl, phonological memory of a conversation about dinner plans]
Episode: "The user is setting up the kitchen table for dinner."
Memory items: [visual memory of a book, a laptop, phonological memory of a comment about the book's content]
Episode: "The user is discussing a book in their office space."
```

```
Output format:
{
"episode": "[short sentence capturing the essence of the memory items]"
}
```

A.1.5 Generate Importance Scores for WM Content.

```
Generate numerical importance scores for all memory items and episodes, from the range of [0,1]. Only generate the
scores, do not output anything else.

Examples:
1. If the memory is task-relevant and requires immediate attention, it's very important and should have a score close
to 1.
Example: The user is holding a hot cup of coffee near a child. Importance score: 0.9
2. If the memory is task-irrelevant, it should have a low importance score close to 0.
Example: The user hears a notification from their phone while working on a project. Importance score: 0.1
3. If the memory is indirectly related to the task, it should have a moderate to high importance score.
Example: The user is packing for a trip, and is reminded of the weather. Importance score: 0.65

Only output the JSON object:
{
    "perception_memory": [importance_value1, importance_value2,...],
    "episodic_buffer": [importance_value1, importance_value2,...]
}
```

A.1.6 Generate Assistance Messages Based on WM State.

```
Generate a list of assistance voice messages based on the new memory item and the updated state of the working memory.
Each assistance message is one sentence providing help or reminders to the user with their current task, along with an
importance score from the range [0,1]. Only generate the list, do not output anything else.

Only output the JSON object in the format below (arranged by importance):
{
  "assistance_messages": [
    {
      "message": "voice_message1",
      "importance": importance_value1
    },
    ...
  ]
}
```

A.1.7 Baseline Assistant Assistance Generation.

```
Evaluate whether to generate an assistance voice message based on the new information updated from the smart glasses:
{information}
If neceesary, the generated assistance message should be a single sentence providing help or reminders to the user with
their current task.

Avoid generating assistance if:
- The value and usefulness of the assistance to the user at the current action are low.
- The information is repetitive or the user is already aware of the information when performing the task.
- The assistance is not important and is more interruptive than useful to the user.

Just output the one sentence assistance if it's valuable for the user's task, and nothing else. If no assistance is
needed, just return NO ASSISTANCE.
```

A.2 Recruitment and Study Procedure

Participants were recruited to complete a task simulating real-world object organization, such as setting a table or packing a bag, while wearing smart glasses. Upon arrival, participants signed a consent form and received a brief overview of the study. The introduction statement is as follows: "You are wearing smart glasses and performing tasks that involve interacting with physical objects in different scenarios. You will also engage in conversation with other people (the experimenter) in the scenario. You will receive AI assistance throughout the session. The focus of this study is on the timeliness of the AI assistance, whether the assistance timing felt just-in-time or interruptive."

For each of four task scenarios, the participant was given contextual instructions before starting (e.g., you are setting up your new office desk next to your co-worker). During the task, the experimenter introduced new information—some relevant to the participant's activity, others intentionally irrelevant—to simulate interruptions. The AI system could provide assistance, and the experimenter marked any observed positive reactions for later discussion in the interview.

After each task, participants completed a short questionnaire. At the end of the session, they completed an exit survey and participated in a semi-structured interview, where they reflected on their experiences and the system's performance.

All sessions were recorded using camera and screen capture tools, and all system conditions were kept blind to participants until after the interview and debriefing.

A.3 Task Setup and Materials

Each task round was designed with a consistent structure: the participant interacted with a set of physical objects (10 distinct object types for each task) while receiving intermittent spoken prompts from the experimenter. These prompts included both task-relevant and irrelevant information to simulate naturalistic interruptions.

Task 1: Dining Setup. **Objects:** bottle, cup, bowl, fork, spoon, orange, banana, apple, potted plant, vase
Task Goal: Set up a dining table to welcome guests for dinner.
Scripted Prompts and Potential Follow-up Action:

- "I need to make a fruit salad with apples and bananas for dessert." → Put apples/bananas aside
- "The spoon is for the coffee in case anyone needs stirring." → Put spoon near cup
- "The potted plant needs more sunlight." → Move plant toward window
- *"I'm thinking of redecorating the living room..."* [irrelevant]

Task 2: Office Organization. **Objects:** laptop, keyboard, mouse, cell phone, book, clock, cup, scissors, notepaper, marker
Task Goal: Organize a new office desk for a meeting.
Scripted Prompts:

- "I have a meeting soon and need to review notes." → Keep notes accessible
- "The clock is not working properly." → Put clock away
- "I use that blue book every day for reference." → Place book nearby
- *"We have a charity event next month."* [irrelevant]

Task 3: Packing for a Trip. **Objects:** backpack, umbrella, tie, handbag, toothbrush, laptop, bottle, banana, sunglasses, medication
Task Goal: Pack a backpack for a business trip
Scripted Prompts:

- "The weather forecast says it's not gonna rain." → Skip umbrella
- "Make sure you have snacks for the road." → Pack fruits/snacks
- "The conference is business formal." → Pack tie
- *"I want to try a new coffee shop in the city."* [irrelevant]

Task 4: Organizing Living Room. **Objects:** remote, vase, sports ball, laptop, book, potted plant, candle, cup, snack, teapot
Task Goal: Style a living room space to host guests.
Scripted Prompts:

- "The guest's child loves basketball." → Leave ball accessible
- "It's daylight so we don't need candles yet." → Remove candles
- "Let's do movie night later—I can set up the TV." → Retrieve remote
- *"The neighbors are having a party tonight."* [irrelevant]

A.4 Interview Questions

- Can you tell me about a time when the system provided assistance at a moment when you really needed it? How did that make you feel?
- Were there any times when the system provided assistance at an undesirable timing? How did that affect your experience?
- Were there any times when the system provided information that was useful to you? How did you feel about receiving that information?
- Can you think of a time when the system provided incorrect or irrelevant information? How did you handle that situation?
- Can you describe a time when you felt fully engaged and focused during your task while using the system? What were you doing during that time?
- Were there any times when you felt distracted or disengaged from the task? What do you think caused that feeling?
- How do you feel about your sense of control and agency?
- How much do you trust the system?