

Decision-aware User Simulation Agent for Evaluating Conversational Recommender Systems

Yuan-Chi Li¹ Li-Chi Chen¹ Sung-Yi Wu¹ Yu-Che Tsai¹ Shou-De Lin^{1,2}

¹National Taiwan University ²NTU AI-CoRE

{r13944007, r14922012, b11701201, f09922081, sdlin}@csie.ntu.edu.tw

Abstract

Conversational recommender systems (CRS) increasingly rely on user simulators for automated evaluation of sales agents. A key requirement for such simulators is the ability to model human decision-making. However, most LLM-based simulators often exhibit unrealistically strong information-processing capabilities, rarely exhibiting the hesitation and decision deferral commonly observed in real consumer behavior, resulting in overly high acceptance rates. To address this limitation, we propose Hesitator, a theory-grounded user simulation framework that explicitly models a specific dimension of human decision-making: decision deferral under choice overload. The framework introduces a modular Decision Module that separates utility-based item selection from overload-aware commitment decisions. Experiments across multiple user simulation frameworks, domains, sales modes, and LLM backbones show that integrating our module consistently mitigates unrealistic behaviors under increasing overload conditions. Furthermore, Hesitator reproduces established findings from behavioral economics, demonstrating its ability to model human decision behavior.

1 Introduction

Large language model (LLM) agents have recently been deployed across a wide range of applications, including conversational assistants (Achiam et al., 2023), autonomous task planning and tool use (Yao et al., 2022; Schick et al., 2023), and complex multi-step reasoning tasks (Wei et al., 2022). Among these, conversational recommender systems (CRS) represent a particularly compelling use case, where agent-driven dialogues guide users toward purchase decisions. Building effective sales agents for CRS, however, demands reliable automated evaluation pipelines, which in turn depend on realistic user simulation (Zhang and Balog, 2020; Yoon et al., 2024). In such simulated environments, faithfully

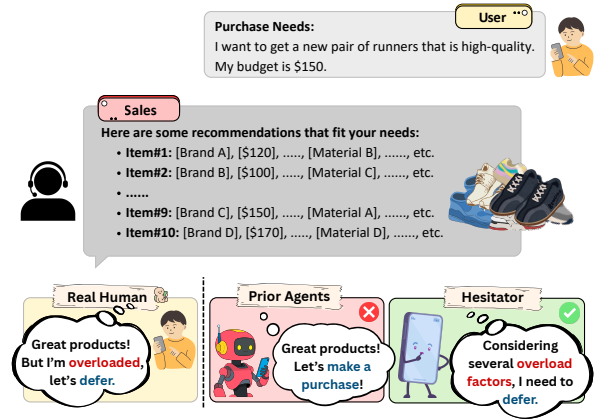


Figure 1: Motivation of Hesitator. Prior user agents often make immediate purchase decisions after identifying good items, ignoring cognitive constraints. In contrast, humans may defer decisions under choice overload. This gap motivates Hesitator, which models hesitation by incorporating overload factors.

modeling human decision-making is critical for assessing CRS effectiveness (Yoon et al., 2024). When user simulators fail to capture the gap between stated intentions and actual behavior, estimated purchase probabilities tend to be overly optimistic, obscuring the intention–behavior frictions that reduce real-world conversion rates (Sheeran, 2002; Carrington et al., 2014).

A prominent research direction for evaluating sales agents involves simulating human behavior through LLM-based agents. For instance, RecUserSim (Chen et al., 2025) implements a multi-module LLM agent architecture with persona, memory, and action modules, whereas UserSimCRS (Bernard and Balog, 2026) follows an agenda-based framework relying on predefined agenda transitions and predefined goals. While these works have demonstrated strong capabilities in modeling realistic individual personas and diverse user populations (Zhao et al., 2025; Bernard and Balog, 2026; Chen et al., 2025), a critical dimension remains largely overlooked: the cognitive decision processes underlying

ing user choices (Bettman et al., 1998). Specifically, in real-world settings, users do not always respond to recommendations immediately or decisively. They may defer decisions when presented with an excessive number of options. This is a well-documented phenomenon known as choice overload (Chernev et al., 2015). However, because LLM-based user agents inherit the strong information-processing capabilities of their underlying models (Brown et al., 2020; Minaee et al., 2024), they can handle large volumes of unstructured information without exhibiting such cognitive strain. As a result, existing user simulation frameworks tend to underrepresent the hesitation and decision deferral that commonly arise in real consumer behavior.

To address this limitation, we propose *Hesitator*¹, a decision-aware user simulation framework built around a Decision Module with two cognitively grounded components. The **Selection Module** employs a two-stage decision process from Bettman et al. (1998), applying non-compensatory filtering to prune the candidate set before compensatory evaluation selects the final item. Without this explicit filtering stage, LLM-based agents tend to give equal consideration to all presented items, which does not reflect how real users selectively attend to options under cognitive constraints. The **Hesitation Module** quantifies choice overload through a four-dimensional overload vector: assortment size, choice set complexity, task difficulty, and preference uncertainty. These follow the moderator structure of Chernev et al. (2015), with assortment size re-parameterized as an explicit moderator and decision goal omitted as inapplicable to our setting (Appendices B.1 and B.4). Crucially, rather than relying on LLM judgment alone, we leverage the regression coefficients derived from that meta-analysis to construct a calibrated mapping function that translates the overload vector into an acceptance probability, grounding the agent’s deferral behavior in large-scale empirical evidence rather than model priors.

Experimental results yield two key findings. First, *Hesitator* reproduces well-established results from behavioral economics, including the inverted-U effect of total information load (Jacoby et al., 1974), the increasing-then-plateau effect of attribute information (Fasolo et al., 2007), and de-

cision conflict theory (Tversky and Shafir, 1992; Anderson, 2003). Second, integrating our decision module into three existing user simulation frameworks demonstrates that this lightweight addition significantly mitigates unrealistic user behaviors across diverse datasets, sales agent configurations, and LLM backbones. Notably, while existing user agents exhibit flat or even increasing acceptance rates under severe choice overload, augmented agents with our decision module display a clear decline consistent with human cognitive patterns. These results together demonstrate that *Hesitator* reliably models the deferral aspect of human decision behavior under diverse overload conditions. We summarize our contributions as follows:

- We identify a key limitation of existing LLM-based user simulators for conversational recommender systems (CRS): their lack of cognitive constraints when processing large amounts of information often leads to unrealistically high acceptance rates, inflating estimates of CRS performance.
- We propose a plug-and-play *Decision Module* that models intention–behavior friction under choice overload, enabling simulated users to adapt their commitment decisions according to the cognitive load of the decision environment.
- We introduce *Hesitator*, a modular user simulation framework that integrates the Decision Module and reproduces established behavioral patterns from decision-making research, enabling more realistic and reliable CRS evaluation.

2 Preliminary

2.1 Behavioral Economics Theory

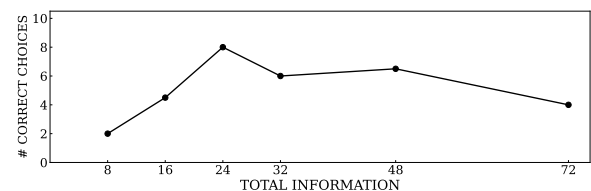


Figure 2: Illustration of the inverted-U relationship between information load and decision quality in consumer decision research (Jacoby et al., 1974). As information increases, decision performance initially improves but eventually declines due to cognitive overload.

Inverted-U Information Overload Curve Early work in consumer psychology established the concept of information overload. As shown in Figure 2,

¹Our code, including the plug-and-play Decision Module interface, the full prompt set, and the evaluation pipeline, is available at <https://github.com/energybubu/Hesitator>.

prior studies proposed an inverted-U relationship between total information load and consumer decision quality (Jacoby et al., 1974). This model jointly considers assortment size and attribute quantity, suggesting that decision performance improves up to an optimal point but declines once cognitive load exceeds an individual’s processing capacity (Eppler and Mengis, 2004; Lurie, 2004). Furthermore, meta-analytic evidence suggests that individuals with lower preference uncertainty are less susceptible to information overload, as experts process and filter large amounts of information better than novices (Chernev et al., 2015).

Independent Effects of Assortment and Attributes Subsequent research suggests that assortment size and the number of attributes should be treated as independent factors rather than a single aggregate information measure (Malhotra, 1982; Keller and Staelin, 1987). Increasing the number of attributes can improve decision performance by providing more diagnostic information (Fasolo et al., 2007). In contrast, larger assortments may impair decision performance when additional options introduce substantial tradeoffs. According to decision conflict theory, when alternatives possess complementary attributes such that none clearly dominates, consumers experience psychological conflict (Tversky and Shafir, 1992; Anderson, 2003), often leading to decision deferral or reduced transaction likelihood (Iyengar and Lepper, 2000; Chernev, 2003).

2.2 Notations

The notation used throughout this work is defined as follows. Calligraphic letters (e.g., $\mathcal{G}$) denote high-level modules or sets, bold symbols (e.g., $\mathbf{v}$) represent multidimensional vectors, and ϕ and σ denote internal persona traits and external scenario constraints, while the remaining Greek letters denote scalar parameters and thresholds. Table 2 in Appendix A summarizes the main symbols.

Agent State Representation The core of our framework is the global user agent state $\mathcal{G} = (\mathcal{P}, \mathcal{S})$, which encapsulates the interaction between a user’s stable characteristics and their immediate environment. The **persona** ($\mathcal{P}$) represents the intrinsic psychological profile of the user, parameterized by a vector of traits $\phi = [\phi_O, \phi_K, \phi_U]^\top$ corresponding to *Openness*, *Pickiness*, and *Preference Uncertainty*. The **scenario** ($\mathcal{S}$) captures the contextual conditions of the recommendation task through

environmental constraints $\sigma = [\sigma_N, \sigma_B, \sigma_T]^\top$, representing *Current Needs*, *Budget*, and *Time Pressure*. In addition, the **dialogue history** ($\mathcal{H}$) records the sequential interaction between the agents, formulated as $\mathcal{H} = \{(\tilde{u}_0), (r_1, \tilde{u}_1), \dots, (r_t, \tilde{u}_t)\}$, where r denotes the response of the Sales Agent and $\tilde{u}$ denotes the response of the User Agent.

Cognitive Overload Hierarchy Following the theoretical framework of Chernev et al. (2015), the consumer cognitive state is modeled using a multidimensional overload vector $\mathbf{v} = [v_a, v_s, v_d, v_u]^\top$. This vector decomposes the choice environment into four primary antecedents: **Assortment Size** (v_a), **Choice Set Complexity** (v_s), **Task Difficulty** (v_d), and **Preference Uncertainty** (v_u). These factors are further organized into a hierarchical structure of leaf variables. The aggregate cognitive impact is quantified as the total overload effect size d_{total} , which serves as the primary latent variable governing the agent’s decision to accept or reject.

2.3 Simulation Process

The simulation framework operates in a CRS as a turn-based dialogue between a Sales Agent and a User Agent. The interaction is governed by a global user state $\mathcal{G} = (\mathcal{P}, \mathcal{S})$, where $\mathcal{P}$ denotes the user persona and $\mathcal{S}$ the shopping scenario. A dialogue history $\mathcal{H} = \{(\tilde{u}_0), (r_1, \tilde{u}_1), \dots, (r_t, \tilde{u}_t)\}$ is maintained, where r_t and $\tilde{u}_t$ denote the responses of the Sales Agent and User Agent at turn t , respectively.

The interaction is initialized by the User Agent with an initial message $\tilde{u}_0$ describing the user’s needs given $\mathcal{G}$. At each turn t , the Sales Agent generates r_t by retrieving candidate items from the database based on inferred preferences and dialogue history $\mathcal{H}$. The User Agent then evaluates r_t together with $\mathcal{G}$ and produces $\tilde{u}_t$, which includes the decision outcome (e.g., accept, reject, or defer) and an explanation when rejecting. The dialogue history is updated as $\mathcal{H} \leftarrow \mathcal{H} \cup \{(r_t, \tilde{u}_t)\}$, and the interaction continues until a purchase decision is reported or the predefined turn limit T is reached.

3 Method

In this section, we present *Hesitator*, an LLM-based framework that simulates user decisions via three different modules. As illustrated in Fig. 3, the Decision Module consists of a *Selection Module* that identifies items satisfying the user’s needs and a *Hesitation Module* that models commitment decisions under cognitive overload. To support

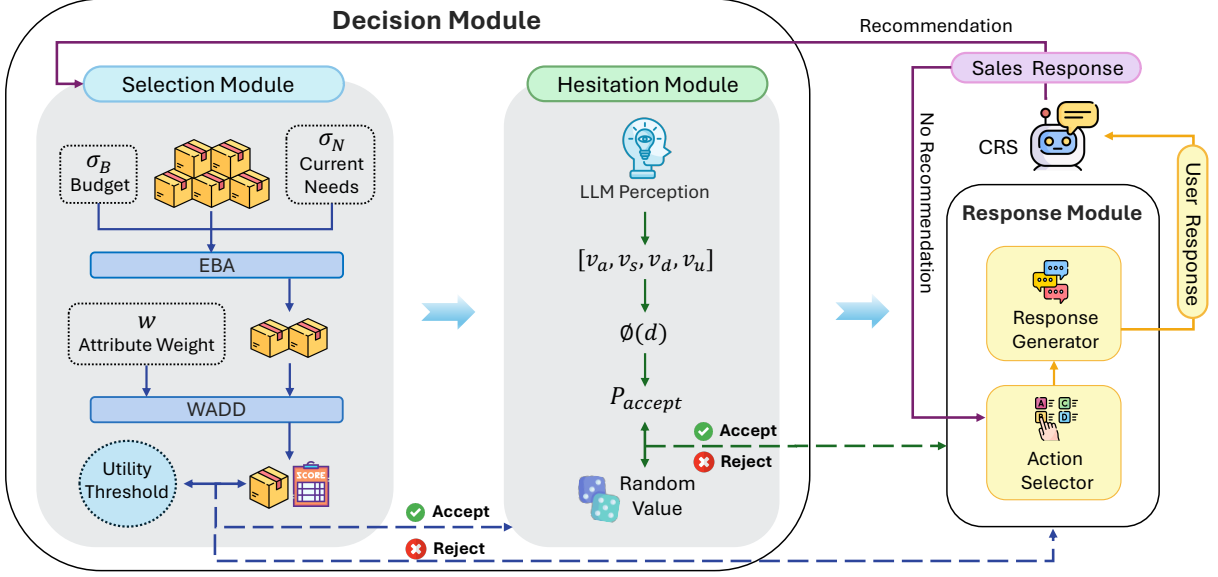


Figure 3: Architecture of the Hesitator framework. The Selection Module filters and ranks items (EBA $\rightarrow$ WADD). If the recommendation passes the selection stage (accept), it proceeds to the Hesitation Module, which estimates the final acceptance probability based on perceived overload factors; otherwise, the system directly generates a rejection response. Solid lines indicate data flow, while dashed lines denote internal stage transitions within the user agent.

conversational recommendation, a *Response Module* further converts internal decisions into natural language responses.

3.1 Decision Module

Real-world decisions involve both evaluating option quality and determining whether to commit under cognitive constraints. Accordingly, the *Decision Module* is formulated as a conditional sequential pipeline: first, the *Selection Module* evaluates the intrinsic utility of the recommended items $\mathcal{I}$ based on the last sales agent recommendation response r_t . If the options fail to meet basic expectations (encoded in $\mathcal{G}$), the agent directly rejects the recommendation at this turn. Conversely, if an acceptable option is identified, the process advances to the *Hesitation Module*, which models choice overload to determine whether the user commits to the choice or defers. The final outcome $\mathcal{D}$ is subsequently routed to the *Response Module*. Appendix B.5 summarizes which parts of this pipeline are inherited from prior theory and which are our own engineering choices.

3.1.1 Selection Module

Evaluating multiple recommended items can be cognitively demanding, making it impractical to exhaustively compare all options. To approximate human decision strategies, the *Selection Mod-*

ule adopts a two-stage heuristic-to-compensatory process commonly observed in human decision-making (Bettman et al., 1998): it first eliminates options that violate the user’s hard constraints, and then estimates compensatory utility over the remaining candidates. This sequence is one cognitively plausible strategy rather than the only one. It targets attribute-driven shopping, the regime most CRS work addresses, where users state constraints and preferences over discrete attributes such as price or brand. Purchases driven by aesthetics or emotion, as well as purely exploratory browsing, may follow different cognitive strategies, which we leave to future work.

Stage 1: Non-compensatory Filtering. If the initial set $|\mathcal{I}| > \theta$, the agent employs Elimination by Aspects (EBA). An item $i \in \mathcal{I}$ is retained in the candidate set $\mathcal{C}$ if it satisfies all critical constraints $\mathcal{K}$ (e.g., price $\leq \sigma_B$):

$$\mathcal{C} = \{i \in \mathcal{I} \mid \forall k \in \mathcal{K}, \mathbb{I}_k(i) = 1\} \quad (1)$$

where $\mathbb{I}_k(\cdot)$ is the indicator function that equals 1 when item i satisfies constraint k and 0 otherwise.

Stage 2: Compensatory Evaluation. When $|\mathcal{C}| \leq \theta$ (we set $\theta = 3$ following established heuristics (Bettman et al., 1998)), the agent evaluates utility u_i via a Weighted Additive (WADD) func-

tion:

$$u_i = \mathbf{w}^\top \mathbf{a}_i = \sum_{j=1}^n w_j \cdot a_{ij} \quad (2)$$

where n is the number of attributes, a_{ij} is the j -th attribute value of item i perceived by the LLM, and w_j is the corresponding user weight extracted from historical reviews (see Appendix C.1). The prompt eliciting a_{ij} is given in Appendix G.2.

Acceptance Mechanism Following compensatory evaluation, the highest utility is $s_u = \max_{i \in \mathcal{C}} u_i$ and the corresponding item is $s_i = \arg \max_{i \in \mathcal{C}} u_i$. The agent compares s_u with a pickiness-dependent threshold $\tau(\phi_K) = \gamma + \alpha \phi_K$ (with $\gamma = 0.6$, $\alpha = 0.1$), where γ and α are tunable hyperparameters used to simulate different user acceptance levels. If $s_u < \tau(\phi_K)$, the recommendation is rejected and passed to the *Response Module*; otherwise, it proceeds to the *Hesitation Module* to evaluate potential deferral. The Selection Module is architecturally modular: alternative selection strategies can be substituted without modifying the Hesitation Module or Response Module.

3.1.2 Hesitation Module

Even when a satisfactory option exists, users may defer decisions due to cognitive overload (Chernev et al., 2015). To capture this phenomenon, the *Hesitation Module* models the probability that a user commits to the identified option given the perceived cognitive load of the decision environment. This module operates in three phases: (1) inferring the user’s cognitive state ($\mathbf{v}$) from dialogue, (2) calibrating these observations to behavioral effect sizes (d_{total}), and (3) mapping the aggregated overload to the final acceptance probability (P_{accept}).

Phase 1: State Perception. The framework elicits its granular intensity scores $v \in \{1, 2, 3\}$ for five leaf dimensions (i.e., $v_a, v_{s,\text{dom}}, v_{s,\text{align}}, v_{d,\text{attr}}$, and $v_{d,\text{fmt}}$) via LLM-based prediction. The remaining leaf variables, v_u and $v_{d,\text{time}}$, are predefined constants determined by the user persona (ϕ_U) and task constraints (σ_T). Composite factors $v_k \in \{v_s, v_d\}$ are formally defined as the arithmetic mean of their respective constituent leaf sets. The elicitation prompt, including the scoring criteria for each leaf dimension, is given in Appendix G.3.

Phase 2: Meta-Analytic Calibration. This phase must turn a perceived overload state into a real-valued acceptance probability. We ground this mapping in the meta-analytic regression of

et al. (2015), estimated over 99 published experimental observations, which places heterogeneous experimental paradigms on a common scale of standardized effect sizes (Cohen’s d) and relates overload moderators to that scale. Its regression coefficients calibrate the module at population scale, so the magnitude of deferral is fixed by human data rather than by the LLM’s own priors.

Four aggregate overload factors enter the calibration: **assortment size** (v_a), **choice set complexity** (v_s), **task difficulty** (v_d), and **preference uncertainty** (v_u), while factors with negligible effects in CRS are omitted (Appendix B.1).

Following the meta-analytic regression formulation, the aggregate overload effect is computed as:

$$d_{\text{total}} = \beta^\top f_{\text{interp}}(\mathbf{v}, \delta) \quad (3)$$

where $\mathbf{v} = [v_a, v_s, v_d, v_u]$ denotes the factor levels from Phase 1 and β are moderator coefficients from the meta-analysis. The bounds δ collect, for each overload factor, the minimum and maximum effect sizes observed across the meta-analyzed studies, and the interpolation $f_{\text{interp}}(\cdot)$ maps a discrete perceived level $v \in \{1, 2, 3\}$ onto a continuous effect-size value inside that empirical range. Detailed mapping procedures and parameter settings are provided in Appendix B.4.

Phase 3: Probabilistic Mapping. To convert the aggregated overload effect size into a decision probability, we adopt the arcsine-based proportion transformation commonly used in statistical power analysis (Cohen, 2013; Fleiss et al., 2013). Rearranging the formulation yields the following closed-form expression:

$$P_{\text{accept}} = \sin^2 \left(\arcsin(\sqrt{P_{\text{base}}}) - \frac{d_{\text{total}}}{2} \right) \quad (4)$$

The derivation is provided in Appendix B.2. Here, $P_{\text{base}} = 0.5$ represents a neutral prior that balances the bidirectional effects of the “more-is-better” tendency and choice overload (see Appendix B.3). The value of d_{total} is bounded to ensure that the resulting probability remains within $[0, 1]$. With $P_{\text{base}} = 0.5$, d_{total} is truncated to the range $[-\pi/2, \pi/2]$. A purchase is triggered only if a random draw $\eta \sim \text{Uniform}(0, 1)$ satisfies $\eta \leq P_{\text{accept}}$; otherwise, the agent defers the decision and provides a rationale.

3.2 Response Module

The *Response Module* converts internal decisions into communicative actions through two prompt-based substeps. It first performs **Action Selection** $\mathcal{A} = \text{LLM}(\mathcal{G}, \mathcal{H}, \mathcal{D})$, prompting an LLM to select one or more high-level communicative actions from a fixed inventory (request recommendation, clarify preference, provide feedback, and inquire attribute) given the global state, dialogue history, and decision outcome. Finally, **Response Synthesis** $\mathcal{R} = \text{LLM}(\mathcal{G}, \mathcal{H}, \mathcal{D}, \mathcal{A})$ prompts the LLM to verbalize the final response $\mathcal{R}$ conditioned on the same context together with the selected actions. The full prompts used for both substeps are provided in Appendix G.4.

4 Experiments

4.1 Experimental Setup

Simulation Environment. We simulate conversational recommendation interactions using two domains from the **Amazon Reviews 2023** dataset (Hou et al., 2026): *Electronics* and *Video Games*. Each experimental configuration is evaluated over $N = 40$ independent sessions. A session follows a turn-based dialogue history $\mathcal{H}$ and imposes a maximum of $T = 20$ interaction turns. Each session is initialized with a unique global state $\mathcal{G}$ that defines the user persona and shopping scenario.

User Profiles. The user state $\mathcal{G}$ consists of a persona $\mathcal{P}$ with traits $\phi = [\phi_O, \phi_K, \phi_U]^\top$ and a shopping scenario $\mathcal{S}$ with constraints $\sigma = [\sigma_N, \sigma_B, \sigma_T]^\top$. Persona traits include openness, pickiness, and preference uncertainty, while the scenario specifies the user’s needs, budget, and time pressure, without revealing the item title to avoid data leakage (Zhu et al., 2024).

Sales Agent. We adopt CSI (Kim et al., 2025) as the sales agent framework and use gpt-oss-20b as the backbone model. The agent retrieves candidate items through a vector retrieval system built on *Qwen3-Embedding-0.6B*. It operates in two modes: **Basic** (preference probing and suggestion) and **Persuasive** (strategic persuasion actions).

Baselines. We compare our method against three representative user simulators: **PersonaLens** (Zhao et al., 2025), **UserSimCRS** (Bernard and Balog, 2026), and **RecUserSim** (Chen et al., 2025), the current SOTA user simulator for CRS. All baselines share the same backbone model (gpt-oss-20b) and identical initialization states to ensure a controlled comparison. Section 4.2 additionally re-

ports a variant in which Qwen3-14B-AWQ replaces gpt-oss-20b as the simulation backbone.

Evaluation Metrics. We evaluate the user agent systems using three complementary metrics: **Success Rate (SR)**, **Subjective Dialogue Quality** (Chen et al., 2025), and **Believability** (Xiao et al., 2023). SR measures the proportion of sessions ending with a purchase decision. Dialogue quality is assessed through pairwise comparisons along five dimensions (realism, naturalness, relevance, clarity, adaptability). Believability is measured via the hallucination ratio $\bar{H}_R$. Detailed system implementation and metrics are provided in Appendix C and D, respectively.

4.2 Overload Effect

We analyze how user-agent decision behavior changes under different cognitive overload levels. Specifically, we evaluate three conditions: *Low*, *Medium*, and *Severe*. The exact parameter settings for these conditions are described in Appendix D.2.

Anomalous Baseline Behaviors Figure 4 shows the success rates of different user-agent frameworks under increasing cognitive overload. Without the Decision Module, baseline simulators exhibit unrealistic patterns: their success rates remain stable or even increase as overload intensifies from Low to Severe. For example, in the Electronics domain, *PersonaLens* rises from 60.0 to 70.0 and *UserSimCRS* from 35.0 to 55.0. Similar trends appear in the Video Games domain, where *PersonaLens* reaches 75.0 and *UserSimCRS* maintains 70.0 under Severe overload. These behaviors contradict well-established findings in consumer decision-making, where increasing overload typically raises cognitive conflict and leads to decision deferral.

Behavioral Correction via Decision Module

After integrating the Decision Module (+DM), all baseline user simulators exhibit behavior consistent with human decision patterns: success rates decline as overload increases from Low to Severe. For example, in the Electronics domain, *RecUserSim+DM* drops from 67.5 to 22.5 and *PersonaLens+DM* from 90.0 to 35.0. A similar trend is observed for *UserSimCRS+DM*, which decreases from 65.0 to 50.0. Consistent patterns also appear in the Video Games domain, where *RecUserSim+DM* falls from 70.0 to 12.5 and *PersonaLens+DM* from 77.5 to 22.5 under Severe overload. Overall, these results indicate that the

	Electronics			Video Games		
UserSimCRS	35.0*	42.5	55.0	60.0	72.5	70.0
PersonaLens	60.0	62.5	70.0	60.0	50.0	75.0
RecUserSim	50.0	45.0	45.0	60.0	40.0	60.0
Hesitator	85.0***	95.0	35.0	82.5***	80.0	32.5
UserSimCRS + DM	65.0	67.5	50.0	62.5*	70.0	37.5
PersonaLens + DM	90.0***	77.5	35.0	77.5***	62.5	22.5
RecUserSim + DM	67.5***	55.0	22.5	70.0***	47.5	12.5
	Low Overload	Medium	Severe Overload	Low Overload	Medium	Severe Overload

Figure 4: Simulation success rates (%) of different user agents under varying levels of cognitive overload in two domains (Electronics and Video Games). The Decision Module (+DM) is designed to mitigate the original tendency of user agents to remain insensitive to cognitive overload, enabling their decision behavior to vary more appropriately with overload level. As a result, success rates decline under severe overload, better reflecting human decision patterns. Statistical significance is determined using the Wilcoxon signed-rank test (Low vs. Severe overload); * ($p < 0.05$), ** ($p < 0.01$), and *** ($p < 0.001$).

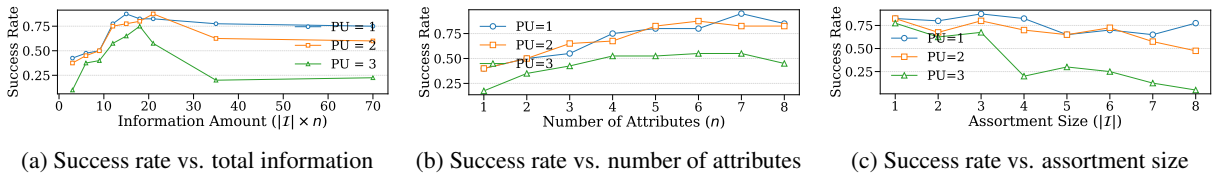


Figure 5: Decision success under different information conditions and preference uncertainty (PU). (a) Total information exhibits an inverted-U relationship with success rate, consistent with information overload findings. (b) Increasing attribute information improves success initially but shows diminishing returns. (c) Larger assortments reduce success under high uncertainty, reflecting decision conflict and choice overload.

Decision Module corrects unrealistic decision behavior in existing simulators and enables them to reproduce the decline in decision commitment commonly observed under severe cognitive overload.

To verify that this correction is not tied to one particular sales agent or backbone model, we repeat the experiment under two further configurations: the Persuasive sales mode, in which the agent employs strategic persuasion actions, and an alternative backbone, Qwen3-14B-AWQ, which replaces gpt-oss-20b in both the user simulator and the sales agent. Together with the main configuration, these yield three distinct CRS behavioral profiles, and the same contrast holds across all three: baseline simulators remain largely insensitive to overload, while the +DM variants decline under severe overload in every configuration except PersonaLens + DM on Electronics under the Qwen backbone. Full results are reported in Appendix F.1 (Figures 8 and 9). The contrast is likewise insensitive to the hand-set parameters of the Decision Module, namely the selection threshold θ , the pickiness-dependent threshold, and the base acceptance probability P_{base} ; the full parameter sweep is reported in Appendix F.2.

4.3 Analysis on Behavioral Economics Theory

To further evaluate the realism of the proposed user agent, we examine whether *Hesitator* can reproduce established findings in behavioral economics.

Inverted-U Information Overload Curve As shown in Figure 5a, the pattern closely follows the classical findings of Jacoby et al. (1974) (Figure 2): success rates initially increase as information grows, peak at moderate information levels, and then decline once cognitive load becomes excessive, forming an inverted-U relationship between information quantity and decision performance. Notably, the decline becomes less pronounced as preference uncertainty decreases. Agents with lower uncertainty exhibit a flatter post-peak slope, indicating greater resilience to information overload. This reproduces a well-documented behavioral phenomenon in consumer decision-making: individuals with higher expertise are less likely to defer decisions under high information load.

Independent Effects of Assortment and Attributes We further analyze the independent effects of assortment size ($|I|$) and attribute quantity (n), as illustrated in Fig. 5b and Fig. 5c. Consis-

tent with prior studies (Malhotra, 1982; Keller and Staelin, 1987), the two factors exhibit distinct impacts on decision behavior. Increasing attribute information generally improves decision performance by providing more diagnostic cues for evaluating alternatives. In contrast, larger assortments introduce decision conflict among competing options, causing success rates to decline as the number of recommendations increases (Tversky and Shafir, 1992).

Human Decision Alignment As a complementary check on decision fidelity, we additionally conduct a small-scale human annotation study comparing the accept/reject decisions of *Hesitator* and *RecUserSim* against human judgments under matched dialogue contexts. *Hesitator* agrees with the human majority label on 70.0% of the annotated conversations, compared with 60.0% for *RecUserSim*, supporting the decision-level fidelity of our simulator; the setup, full results, and the caveats of this preliminary study are reported in Appendix E.

4.4 Comprehensive Conversation Simulation Evaluation

Subjective Evaluation Figure 6 compares the conversational simulation quality across user agents. *Hesitator* consistently outperforms the baseline simulators (*RecUserSim*, *UserSimCRS*, and *PersonaLens*) across multiple evaluation dimensions, including overall quality, naturalness, clarity, adaptability, relevance, and realism. This indicates that our user agent not only models decision behavior more realistically but also maintains strong conversational quality, producing responses that are more natural, coherent, and contextually appropriate than existing simulators. Detailed evaluation setup and the per-dimension scoring instructions given to the judge are provided in Appendix D.1.2.

Believability Evaluation Table 1 reports hallucination ratios under vanilla and decision-module settings. Adding the Decision Module does not increase hallucination and slightly reduces hallucination for most simulators. *PersonaLens* decreases from 0.1290 to 0.1088 and *UserSimCRS* from 0.2548 to 0.2268, while *RecUserSim* changes only marginally from 0.0584 to 0.0624. These results suggest that the Decision Module preserves response believability without introducing additional hallucination. *Hesitator* nevertheless exhibits a higher hallucination ratio than *RecUserSim*, likely because the latter includes a dedicated response

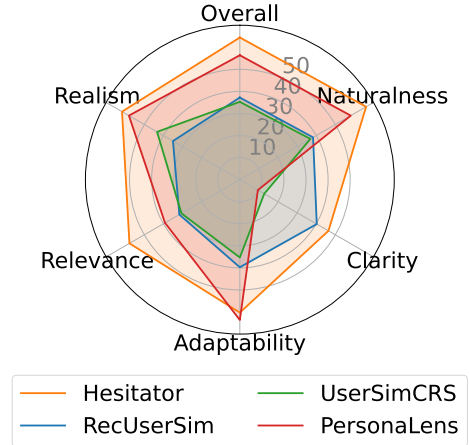


Figure 6: Subjective evaluation of conversational simulation quality across user agents. Each axis reports a user agent’s win rate (%) in pairwise dialogue-level comparisons judged by *Gemini 3 Flash*, aggregated over all $\binom{4}{2}$ agent pairs and $N = 40$ sessions. Setup and per-dimension instructions are in Appendix D.1.2.

Model	Vanilla	DM
Hesitator	0.0969	–
PersonaLens	0.1290	0.1088
RecUserSim	0.0584	0.0624
UserSimCRS	0.2548	0.2268

Table 1: Average hallucination ratio under vanilla and decision-module (+DM) settings. The Decision Module does not increase hallucination and slightly reduces it for most simulators, indicating preserved response believability.

refinement stage that helps reduce hallucinations before output. Such a stage operates independently of the Decision Module and can be layered on top of *Hesitator* when lower hallucination is required.

4.5 Ablation Study

To isolate the effect of structured item evaluation in the proposed framework, we conduct an ablation study on the *Selection Module* by replacing it with a standard rating-based prompt. The *Selection Module* models human-like item evaluation via a two-stage decision strategy. As shown in Figure 7, removing this module leads to unstable performance as the number of attributes increases, and the expected positive effect of additional attributes is no longer consistently observed, deviating from human behavior. This suggests that, without structured utility computation, the user agent fails to faithfully reproduce human decision-making. The rating prompt and its acceptance rule are specified

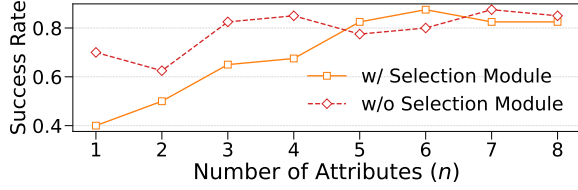


Figure 7: Ablation results for the Selection Module with varying numbers of attributes.

in Appendix D.4.

5 Related Work

User Simulation for Conversational Recommendation User simulation is widely used to evaluate CRS without requiring costly human studies. Early work relied on rule-based simulators that generate responses using predefined policies or slot-filling strategies (Zhang and Balog, 2020), offering strong controllability but limited linguistic diversity; later agenda-based simulators model users as goal-driven agents maintaining structured dialogue goals, as exemplified by *UserSimCRS* (Bernard and Balog, 2026). Recent research increasingly adopts LLM-based simulators to produce more natural and context-aware responses. For instance, *PersonaLens* (Zhao et al., 2025) conditions LLMs on persona descriptions and dialogue context, while *RecUserSim* (Chen et al., 2025) introduces a modular architecture to improve behavioral diversity and extensibility. Other work enhances simulation realism by modeling richer user characteristics and evolving intents, such as personality-driven simulators that condition LLMs on stable persona traits (Ma et al., 2025; Gromada et al., 2025) and CRS methods that track intent evolution during dialogue (e.g., DICR (Zhou et al., 2022) and Chat-REC (Gao et al., 2023)). However, LLM-based simulators rely on implicit reasoning without explicitly modeling human decision processes, often producing overly rational behaviors that overlook hesitation under choice overload. We address this by introducing an explicit Decision Module that separates utility-based selection from overload-aware hesitation to enable structured and controllable commitment decisions.

Cognitive Architectures for Decision Modeling

Cognitive science has long studied computational models of human decision-making. Architectures such as ACT-R (Anderson et al., 2004) and SOAR (Laird, 2019) simulate human cognition through structured modules for perception, memory, and

action selection, enabling the modeling of complex behavioral patterns. However, these architectures are typically designed for controlled laboratory tasks and rely on symbolic representations that are difficult to integrate with modern conversational systems. In contrast, our approach augments LLM-based simulators with a theory-grounded Decision Module that models cognitive constraints such as choice overload.

6 Conclusion

In this paper, we introduce Hesitator, a theory-grounded user simulation framework designed to model one important dimension of human decision-making under cognitive constraints in conversational recommendation: decision deferral under choice overload. By integrating a modular Decision Module, composed of a Selection Module that mimics heuristic to compensatory item evaluation and a Hesitation Module that models choice overload, the framework enables LLM-based user agents to simulate both utility-driven selection and overload-induced decision deferral. Experiments show that incorporating our module consistently corrects unrealistic behaviors in existing simulators under increasing overload conditions while maintaining strong conversational quality. More importantly, Hesitator reproduces established findings from behavioral economics, showing that the Decision Module captures the overload-driven deferral that existing simulators miss. This narrows the gap between simulated and human acceptance behavior, which is a prerequisite for trustworthy automated evaluation of conversational recommendation systems.

Limitations

This work has several limitations. First, our experiments focus on two product domains from the Amazon Reviews dataset. Although these domains are widely used in conversational recommendation research, future work should examine whether similar behavioral patterns emerge in other domains with different decision characteristics. Second, the hesitation mechanism relies on parameters derived from a large-scale meta-analysis in consumer psychology. While this grounding provides empirical support for the modeling assumptions, future work could explore adaptive calibration using domain-specific behavioral data. Third, user personas are treated as static throughout each interaction session,

whereas real-world user preferences may evolve during conversations. Extending the framework to model dynamic preference shifts could further improve simulation realism. Finally, each configuration is evaluated over $N = 40$ sessions; although the reported contrasts are statistically significant, larger-scale evaluation would further strengthen the behavioral claims.

Ethics Statement

A simulator that models when consumers hesitate could be used to optimize persuasive strategies that exploit those moments rather than to evaluate sales agents more faithfully; we release *Hesitator* for auditing rather than optimizing conversion. That auditing role has limits of its own: the overload parameters come from consumer studies conducted largely on Western populations, so the simulated behavior is not a universal standard of human decision-making, and simulator-based evaluation complements rather than replaces evaluation with human participants. All experiments use the public Amazon Reviews dataset, and personas are synthetic, containing no personally identifying information.

Acknowledgments

This material is based upon work supported by National Science and Technology Council, ROC under grant number 114-2221-E-002-134-MY3, NTU AI Center of Research Excellence within Taiwan Centers of Excellence in Artificial Intelligence, and by National Taiwan University and Academia Sinica Innovative Joint Program, under grant AS-NTU-114-06.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. [Gpt-4 technical report](#). *arXiv preprint arXiv:2303.08774*.
- Christopher J Anderson. 2003. [The psychology of doing nothing: forms of decision avoidance result from reason and emotion](#). *Psychological bulletin*, 129(1):139.
- John R Anderson, Daniel Bothell, Michael D Byrne, Scott Douglass, Christian Lebiere, and Yulin Qin. 2004. [An integrated theory of the mind](#). *Psychological review*, 111(4):1036.
- Nolwenn Bernard and Krisztian Balog. 2026. [User-simcrs v2: Simulation-based evaluation for conversational recommender systems](#). In *European Conference on Information Retrieval*, pages 496–510. Springer.
- James R Bettman, Mary Frances Luce, and John W Payne. 1998. [Constructive consumer choice processes](#). *Journal of consumer research*, 25(3):187–217.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. [Language models are few-shot learners](#). *Advances in neural information processing systems*, 33:1877–1901.
- Michal J Carrington, Benjamin A Neville, and Gregory J Whitwell. 2014. [Lost in translation: Exploring the ethical consumer intention–behavior gap](#). *Journal of business research*, 67(1):2759–2767.
- Luyu Chen, Quanyu Dai, Zeyu Zhang, Xueyang Feng, Mingyu Zhang, Pengcheng Tang, Xu Chen, Yue Zhu, and Zhenhua Dong. 2025. [Recusersim: A realistic and diverse user simulator for evaluating conversational recommender systems](#). In *Companion Proceedings of the ACM on Web Conference 2025*, pages 133–142.
- Alexander Chernev. 2003. [When more is less and less is more: The role of ideal point availability and assortment in consumer choice](#). *Journal of consumer Research*, 30(2):170–183.
- Alexander Chernev, Ulf Böckenholt, and Joseph Goodman. 2015. [Choice overload: A conceptual review and meta-analysis](#). *Journal of Consumer Psychology*, 25(2):333–358.
- Jacob Cohen. 2013. [Statistical power analysis for the behavioral sciences](#). routledge.
- Martin J Eppler and Jeanne Mengis. 2004. [The concept of information overload: A review of literature from organization science, accounting, marketing, mis, and related disciplines](#). *The information society*, 20(5):325–344.
- Barbara Fasolo, Gary H McClelland, and Peter M Todd. 2007. [Escaping the tyranny of choice: When fewer attributes make choice easier](#). *Marketing theory*, 7(1):13–26.
- Joseph L Fleiss, Bruce Levin, and Myunghee Cho Paik. 2013. [Statistical methods for rates and proportions](#). john wiley & sons.
- Yunfan Gao, Tao Sheng, Youlin Xiang, Yun Xiong, Haofen Wang, and Jiawei Zhang. 2023. [Chat-rec: Towards interactive and explainable llms-augmented recommender system](#). *arXiv preprint arXiv:2303.14524*.
- Justyna Gromada, Alicja Kasicka, Ewa Komkowska, Lukasz Krajewski, Natalia Krawczyk, Morgan Veyret, Bartosz Przybył, Lina M. Rojas-Barahona,

- and Michał K. Szczerbak. 2025. [Evaluating conversational agents with persona-driven user simulations based on large language models: A sales bot case study](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 230–245, Suzhou (China). Association for Computational Linguistics.
- Yupeng Hou, Jiacheng Li, Xiangjun Fu, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. 2026. [Bridging language and items for retrieval and recommendation: Benchmarking LLMs as semantic encoders](#). In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3251–3265, San Diego, California, United States. Association for Computational Linguistics.
- Sheena S Iyengar and Mark R Lepper. 2000. [When choice is demotivating: Can one desire too much of a good thing?](#) *Journal of personality and social psychology*, 79(6):995.
- Jacob Jacoby, Donald E Speller, and Carol A Kohn. 1974. [Brand choice behavior as a function of information load](#). *Journal of marketing research*, 11(1):63–69.
- Kevin Lane Keller and Richard Staelin. 1987. [Effects of quality and quantity of information on decision effectiveness](#). *Journal of consumer research*, 14(2):200–213.
- Tongyoung Kim, Jeongeun Lee, SooJin Yoon, SungHwan Kim, and Dongha Lee. 2025. [Towards personalized conversational sales agents: Contextual user profiling for strategic action](#). In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 5131–5154, Suzhou, China. Association for Computational Linguistics.
- John E Laird. 2019. *The Soar cognitive architecture*. MIT press.
- Nicholas H Lurie. 2004. [Decision making in information-rich environments: The role of information structure](#). *Journal of consumer research*, 30(4):473–486.
- Chenglong Ma, Ziqi Xu, Yongli Ren, Danula Hetiachchi, and Jeffrey Chan. 2025. [Pub: an llm-enhanced personality-driven user behaviour simulator for recommender system evaluation](#). In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2690–2694.
- Naresh K Malhotra. 1982. [Information load and consumer decision making](#). *Journal of consumer research*, pages 419–430.
- Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. [Large language models: A survey](#). *arXiv preprint arXiv:2402.06196*.
- Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. [Toolformer: Language models can teach themselves to use tools](#). *Advances in neural information processing systems*, 36:68539–68551.
- Paschal Sheeran. 2002. [Intention–behavior relations: a conceptual and empirical review](#). *European review of social psychology*, 12(1):1–36.
- Amos Tversky and Eldar Shafir. 1992. [Choice under conflict: The dynamics of deferred decision](#). *Psychological science*, 3(6):358–361.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. [Chain-of-thought prompting elicits reasoning in large language models](#). *Advances in neural information processing systems*, 35:24824–24837.
- Colin White, Samuel Dooley, Manley Roberts, Arka Pal, Benjamin Feuer, Siddhartha Jain, Ravid Shwartz-Ziv, Neel Jain, Khalid Saifullah, Sreemanti Dey, and 1 others. 2025. [Livebench: A challenging, contamination-limited llm benchmark](#). In *International Conference on Learning Representations*, volume 2025, pages 91595–91631.
- Yang Xiao, Yi Cheng, Jinlan Fu, Jiashuo Wang, Wenjie Li, and Pengfei Liu. 2023. [How far are llms from believable ai? a benchmark for evaluating the believability of human behavior simulation](#). *arXiv preprint arXiv:2312.17115*.
- Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. [React: Synergizing reasoning and acting in language models](#). *arXiv preprint arXiv:2210.03629*.
- Se-eun Yoon, Zhankui He, Jessica Echterhoff, and Julian McAuley. 2024. [Evaluating large language models as generative user simulators for conversational recommendation](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1490–1504, Mexico City, Mexico. Association for Computational Linguistics.
- Shuo Zhang and Krisztian Balog. 2020. [Evaluating conversational recommender systems via user simulation](#). In *Proceedings of the 26th acm sigkdd international conference on knowledge discovery & data mining*, pages 1512–1520.
- Zheng Zhao, Clara Vania, Subhradeep Kayal, Naila Khan, Shay B. Cohen, and Emine Yilmaz. 2025. [PersonaLens: A benchmark for personalization evaluation in conversational AI assistants](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 18023–18055, Vienna, Austria. Association for Computational Linguistics.

Jinfeng Zhou, Bo Wang, Minlie Huang, Dongming Zhao, Kun Huang, Ruifang He, and Yuexian Hou. 2022. [Aligning recommendation and conversation via dual imitation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 549–561, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Lixi Zhu, Xiaowen Huang, and Jitao Sang. 2024. [How reliable is your simulator? analysis on the limitations of current llm-based user simulators for conversational recommendation](#). In *Companion Proceedings of the ACM Web Conference 2024*, pages 1726–1732.

A Notation Summary

Table 2 collects the notation used throughout the paper, grouped into the state variables that describe the user agent and the dialogue, the variables of the choice overload model, the quantities computed during the decision process, and the outputs of the Decision and Response Modules.

B Theoretical Background

B.1 Rationale for Omitting Partial Overload Factors

Although [Chernev et al. \(2015\)](#) identifies a broader set of determinants, we selectively exclude dimensions that are incompatible with our experimental design. For example, the Decision Goal factor, including decision intent (buying vs. browsing) and decision accountability, is omitted. In our setting, user profiles are initialized with explicit goal-oriented purchase needs, rendering browsing behavior irrelevant. Moreover, accountability effects, which arise when consumers must justify their decisions to others, do not apply in a private interaction between a simulated user agent and a sales system. Within our experimental framework, these dimensions inherently yield a null effect; specifically and contribute an effect size of zero to the regression model established by [Chernev et al. \(2015\)](#). Consequently, their exclusion does not compromise the validity of the analysis.

B.2 The Formula Derivation for Equation 4

Starting from the definition of the total effect size:

$$d_{\text{total}} = 2 \left(\arcsin \sqrt{P_{\text{base}}} - \arcsin \sqrt{P_{\text{adjust}}} \right) \quad (5)$$

Dividing both sides by 2 gives

$$\frac{d_{\text{total}}}{2} = \arcsin \sqrt{P_{\text{base}}} - \arcsin \sqrt{P_{\text{adjust}}} \quad (6)$$

Symbol	Description
<i>State Variables</i>	
$\mathcal{G}$	Global user agent state ($\mathcal{P}, \mathcal{S}$)
$\mathcal{P}$	Persona
ϕ_O, ϕ_K, ϕ_U	Openness, Pickiness, and Preference Uncertainty
$\mathcal{S}$	Purchase Scenario
$\sigma_N, \sigma_B, \sigma_T$	Current Needs, Budget, and Time Pressure
$\mathcal{H}$	Dialogue History $\{(\tilde{u}_0), (r_1, \tilde{u}_1), \dots\}$
r	Response of the Sales Agent
$\tilde{u}$	Response of the User Agent
T	Maximum number of dialogue turns
<i>Choice Overload Modeling</i>	
$\mathbf{v}$	Overload vector $[v_a, v_s, v_d, v_u]^\top$
v_a	Assortment Size
v_s	Choice Set Complexity
$v_{s,\text{dom}}$	Dominance (Presence of dominant option)
$v_{s,\text{align}}$	Alignability (Attribute commensurability)
v_d	Task Difficulty
$v_{d,\text{time}}$	Time Pressure
$v_{d,\text{attr}}$	Number of Attributes
$v_{d,\text{fmt}}$	Format Complexity
v_u	Preference Uncertainty
β	Coefficients of the four overload factors
δ	Empirical effect-size bounds from the meta-analysis
f_{interp}	Level-to-effect-size interpolation
d_{total}	Calculated overload effect size
<i>Decision Process</i>	
$\mathcal{I}, \mathcal{C}$	Initial and candidate recommendation sets ($\mathcal{C} \subseteq \mathcal{I}$)
θ	Cognitive threshold for filtering
$\mathcal{K}$	Set of critical (hard) constraints
$\mathbb{I}_k$	Indicator for satisfying constraint k
$\mathbf{a}_i, \mathbf{w}$	Attribute vector of item i and user weights
n	Number of attributes per item
u_i	Evaluated utility of item i
s_u, s_i	Maximum utility and the corresponding item
$\tau(\phi_K)$	Pickiness-dependent acceptance threshold
γ, α	Threshold hyperparameters (0.6 and 0.1)
P_{base}	Base acceptance probability (default: 0.5)
P_{accept}	Final acceptance probability
η	Uniform random draw for the purchase trial
<i>Decision and Response Outputs</i>	
$\mathcal{D}$	Decision outcome of the Decision Module
$\mathcal{A}$	Selected communicative actions
$\mathcal{R}$	Generated user response

Table 2: Summary of the main notations.

Rearranging to isolate the term containing P_{adjust} :

$$\arcsin \sqrt{P_{\text{adjust}}} = \arcsin \sqrt{P_{\text{base}}} - \frac{d_{\text{total}}}{2} \quad (7)$$

Applying the sine function to both sides yields

$$\sqrt{P_{\text{adjust}}} = \sin \left(\arcsin \sqrt{P_{\text{base}}} - \frac{d_{\text{total}}}{2} \right) \quad (8)$$

Finally, squaring both sides and letting $P_{\text{adjust}} =$

P_{accept} gives

$$P_{\text{accept}} = \sin^2 \left(\arcsin \sqrt{P_{\text{base}}} - \frac{d_{\text{total}}}{2} \right). \quad (9)$$

B.3 Design for P_{base}

Notably, the impact of assortment size on consumer behavior is bi-directional and contingent upon the specific moderating environment. When the antecedents of overload—such as task difficulty or choice set complexity—are at low levels, the observed effect size (Cohen’s d) typically becomes negative. This negative d -value signifies a “more-is-better” effect, where larger assortments provide a competitive advantage by increasing the likelihood of an ideal preference match without exceeding cognitive limits. To achieve behavioral high-fidelity, a simulation framework must therefore support bi-directional dynamics: it must decrease the acceptance probability to model choice deferral under high-overload conditions, while conversely increasing it when the cognitive load is sufficiently low to reflect the utility gains and increased satisfaction inherent in extensive assortments.

B.4 Connection to the Meta-Analytic Regression Model

Chernev et al. (2015) estimates the magnitude of the choice overload effect using a meta-analytic regression model of the form

$$d = \beta_0 + \beta_s x_s + \beta_d x_d + \beta_u x_u + \epsilon \quad (10)$$

where d denotes the standardized effect size reported in each experimental observation, x_k represents moderator variables characterizing the decision environment, β denotes the regression coefficients estimated in the meta-analysis, and ϵ is the regression residual (see Table 3 for the exact values).

Factor	β_k	$\delta_k^{\min}$	$\delta_k^{\max}$
v_a	0.41	-0.18	1.22
v_s	0.55	-1.65	0.48
v_d	0.37	-0.59	0.81
v_u	0.32	-1.34	1.21

Table 3: Parameters used to compute the overload effect size.

In the original analysis, assortment size is primarily operationalized through experimental contrasts between small and large choice sets and is

therefore absorbed into the intercept term. In our simulation environment, however, the number of recommended items varies explicitly across dialogue turns. To capture this variation, we reparameterize the baseline effect as a moderator associated with assortment size.

Concretely, we express the assortment contribution as

$$d_a = f_{\text{interp}}(v_a, \delta_a) \quad (11)$$

and define the total effect size as

$$d_{\text{total}} = \sum_{k \in \{a, s, d, u\}} \beta_k d_k \quad (12)$$

where $d_k = f_{\text{interp}}(v_k, \delta_k)$ is the interpolated effect size of factor k , and the assortment coefficient β_a reflects the baseline magnitude of the assortment-size contrast captured by the intercept term in the original meta-analytic regression.

B.5 Theory-Grounded Components vs. Engineering Choices

Table 4 separates the components of the Decision Module that are inherited from prior theory from those that are engineering choices made by us. The latter are the parameters swept in the sensitivity analysis of Appendix F.2.

Component	Source or value
<i>Theory-grounded</i>	
EBA $\rightarrow$ WADD pipeline	Bettman et al. (1998)
Four overload dimensions	Chernev et al. (2015)
Coefficients β	Chernev et al. (2015)
Effect-size range δ	Chernev et al. (2015)
Arcsine proportion transform	Cohen (2013); Fleiss et al. (2013)
<i>Engineering choices</i>	
Selection threshold θ	3; swept {2, 3, 4, 5}
Pickiness threshold $\tau(\phi_K)$	0.8; swept {0.7, 0.8, 0.9}
Base acceptance P_{base}	0.5; swept {0.4, 0.5, 0.6}
LLM perception scale	1–3; tractable elicitation
Composite aggregation	Mean of leaf variables

Table 4: Provenance of the Decision Module’s design decisions. Engineering parameters are reported with the range over which they were varied in Appendix F.2.

C Implementation Details

C.1 User Profile Construction

The global user state $\mathcal{G}$ is defined as the combination of a persona $\mathcal{P}$ with traits $\phi = [\phi_O, \phi_K, \phi_U]^\top$ and a shopping scenario $\mathcal{S}$ with constraints $\sigma =$

$[\sigma_N, \sigma_B, \sigma_T]^\top$. We instantiate these components using synthesized profiles derived from the Amazon Reviews 2023 corpus.

Persona and Scenario Instantiation Following Kim et al. (2025), the openness value ϕ_O is inferred from historical reviews via LLM-based analysis. The pickiness ϕ_K and preference uncertainty ϕ_U are treated as controllable variables in $\{1, 2, 3\}$. For the shopping scenario, the dynamic budget σ_B is sampled from the interquartile range of prices within the corresponding leaf category to ensure realistic purchasing power, while the time pressure σ_T serves as a task-specific constraint.

The granularity of the user’s current needs σ_N is modulated by preference uncertainty ϕ_U . When uncertainty is medium or high ($\phi_U \in \{2, 3\}$), the user is modeled with vague preferences: item descriptions are limited to coarse attributes, and the weight vector $\mathbf{w}$ is initialized uniformly. Conversely, when uncertainty is low ($\phi_U = 1$), the user is modeled as a more expert decision-maker, using a non-uniform weight vector $\mathbf{w}$ extracted from historical review data.

Asymmetric Initialization To better reflect realistic sales interactions, we adopt an asymmetric profile design. The User Agent is initialized with the complete global state $\mathcal{G}$. In contrast, the Sales Agent is provided only with the observable profile $\{\phi_O, \phi_K\}$, representing traits that could plausibly be inferred from a user’s review history. This information asymmetry requires the Sales Agent to proactively infer the user’s latent intent and constraints through dialogue.

C.2 Sales Agent Implementation

Recommendation Retrieval Mechanism To facilitate efficient and relevant item retrieval, we construct a vector-based retrieval system using Qwen3-Embedding-0.6B as the backbone embedding model. All retrieval operations are performed using cosine similarity over an HNSW-based vector index, returning the top- K results ($K = 50$).

Database Establishment For each item in Amazon Reviews 2023 across the target domains, we construct a textual representation by concatenating the title, category path, and product description. This representation is then embedded into a 1,024-dimensional vector space and stored in a vector database for subsequent retrieval.

Retrieval Mechanisms The Sales Agent employs two retrieval modes to support different strategic actions.

- **Query-based Retrieval (*Suggest*)**. When the agent identifies explicit user preferences, it converts the inferred preference description into a query embedding and retrieves the top- K candidate items from valid subcategories.
- **Item-based Retrieval (*Persuade*)**. To support persuasive strategies, the agent performs a similarity search centered on the currently recommended item. By retrieving items with high semantic similarity, the agent can present alternative products that reinforce salient attributes or provide comparative context.

D Experimental Setup

D.1 Evaluation Metrics

D.1.1 Success Rate

Success Rate (SR) measures the proportion of simulated sessions that end with a completed purchase decision. Given N simulated sessions, it is defined as

$$SR = \frac{1}{N} \sum_{i=1}^N \mathbb{I}(\text{purchase}_i), \quad (13)$$

where $\mathbb{I}(\cdot)$ is the indicator function, which equals 1 if session i ends in purchase and 0 otherwise.

D.1.2 Subjective Dialogue Quality

We evaluate dialogue quality using pairwise comparisons between dialogues generated by different user agents. Given four agents, we construct all $\binom{4}{2}$ agent pairs and conduct dialogue-level comparisons for each pair. Each pair is evaluated along five dimensions: *realism*, *naturalness*, *relevance*, *clarity*, and *adaptability*, where the judge selects the better dialogue for each dimension.

We use *Gemini 3 Flash* as the judge model because it provides stronger analytical and reasoning capabilities than the backbone model, *gpt-oss-20b*, used for dialogue simulation (White et al., 2025). For each comparison setting, we aggregate the outcomes across all $N = 40$ simulated sessions and report the resulting win rates.

The full judge prompt is given in Appendix G.6; it embeds the following instruction for each dimension:

- **Naturalness**: assess whether the simulator’s language is fluent and human-like, allowing informal chat conventions.
- **Clarity**: assess whether the responses are precise, coherent, and easy to understand, with minimal ambiguity.
- **Adaptability**: assess whether the simulator adjusts its responses based on the agent’s actions and the evolving context.
- **Relevance**: assess whether the responses are contextually appropriate and provide pertinent feedback.
- **Realism**: assess whether the behavior is plausible and consistent, without contradictions or repetitive patterns.

D.1.3 Hallucination Ratio

To assess the believability of the simulated user agent, we compute the Hallucination Ratio ($\bar{H}_R$), which quantifies the proportion of mentioned entities that are not supported by the available evidence.

For each dialogue turn t in scenario i , the turn-level hallucination ratio is defined as

$$H_{R,i,t} = \frac{H_{i,t}}{H_{i,t} + R_{i,t}}, \quad (14)$$

where $H_{i,t}$ is the number of hallucinated entities and $R_{i,t}$ is the number of grounded entities. An entity is considered grounded if it can be verified from either the user profile $\mathcal{P}$ or the dialogue history $\mathcal{H}$; otherwise, it is counted as hallucinated. Entity extraction and classification are performed using a structured LLM-as-a-judge parser, whose prompt is given in Appendix G.5.

To obtain the final reported score, we first average the turn-level ratios over the $T = 20$ turns within each scenario, and then average the resulting scenario-level values across all $N = 40$ independent sessions:

$$\bar{H}_R = \frac{1}{N} \sum_{i=1}^N \left(\frac{1}{T} \sum_{t=1}^T H_{R,i,t} \right). \quad (15)$$

D.2 Hyper-parameters for Overload Experiments

D.3 Hyper-parameters for Behavioral Reproduction

The following parameters are fixed across all experiments:

	Low	Medium	Severe
ϕ_O (Openness)	2	2	2
σ_T (Time Pressure)	1	2	3
Presentation format	1	1	3
ϕ_U (Preference Uncertainty)	1	2	3
$ \mathcal{I} $ (Recommendations)	3	3	3
n (Number of Attributes)	8	8	8

Table 5: Hyper-parameter settings used in the overload-effect experiment. Every row is a configured input: rows above the midrule use the $\{1, 2, 3\}$ scale, and rows below are raw environment quantities. The presentation format drives the sales agent’s message layout, which the Hesitation Module then scores as $v_{d,\text{fmt}}$.

Parameter	Value
ϕ_O (Openness)	2
ϕ_K (Pickiness)	2
σ_T (Time Pressure)	2
Presentation format	1

Table 6: Fixed hyper-parameters in behavioral reproduction experiments.

The sweeps below vary the raw environment quantities $|\mathcal{I}|$ and n , which the Hesitation Module then perceives as the discrete levels v_a and $v_{d,\text{attr}} \in \{1, 2, 3\}$.

Total Information Curve Both assortment size $|\mathcal{I}|$ and number of attributes n are varied.

Assortment Size Curve $n = 5$, while $|\mathcal{I}|$ is varied.

Number of Attributes Curve $|\mathcal{I}| = 3$, while n is varied.

D.4 Ablation: Rating-Based Selection Prompt

In the ablation study, the Selection Module is replaced by a standard rating-based prompt. This is a single-stage holistic evaluation: given the user persona and the last message of the sales agent, the LLM outputs an integer *objective score* on a 1–5 scale indicating how well the recommendation as a whole matches the user’s current needs.

- 1 = no current needs are fulfilled
- 2 = most current needs are not fulfilled
- 3 = at least two current needs are not fulfilled
- 4 = only one current need is not fulfilled
- 5 = all current needs are fulfilled

Acceptance is then determined by comparing this score against a pickiness-dependent threshold: a

score of at least 3 for $\phi_K = 1$, at least 4 for $\phi_K = 2$, and 5 for $\phi_K = 3$. This mirrors the threshold structure $\tau(\phi_K)$ used by our Selection Module, so that the two variants differ only in how item utility is computed and not in how strictly the user accepts. The full prompt is given in Appendix G.7.

E Human Study Details

Annotation Setup. We conduct a human annotation study to evaluate whether the decisions generated by different user simulators align with human judgments. Each conversation is annotated by three annotators who label the final user decision as either *Accept* (A) or *Reject* (R). All annotations are collected independently without access to the simulator identity. The three annotators are authors of this paper, all with a background in computer science and natural language processing; they participated voluntarily and received no compensation. The annotators are therefore not independent of the work, which we take into account when interpreting the results below.

Sampling Strategy. We randomly sample 30 scenario indices from the 40 simulated scenarios. For each sampled index, we retrieve the corresponding conversations generated by both user simulators (*Hesitator* and *RecUserSim*). This design ensures that the compared conversations originate from identical underlying scenarios, enabling a controlled comparison between simulators. We restrict the comparison to *RecUserSim* because it is the strongest of the simulators we consider, which makes it the most stringent reference point for decision fidelity.

For annotation, we extract the final recommended item presented in the last dialogue turn of each conversation. Annotators then judge whether a human user would accept or reject the recommendation, regardless of the original simulator decision. As a result, the annotation task focuses on evaluating the plausibility of the final recommendation outcome under the given dialogue context.

Majority Vote. For each conversation, we compute a human majority label based on the annotations. We observe that there are no tied annotations among the 30 decisions.

System–Majority Alignment (Sys–Maj). To measure how closely a simulator’s decisions align with human judgments, we compare the simulator’s

System	Sys–Maj
Hesitator	70.0%
RecUserSim	60.0%

Table 7: Decision alignment with the human majority vote of three annotators over 30 paired conversations. **Sys–Maj** is the proportion of conversations on which a simulator’s accept/reject decision matches the annotator majority label.

decision for each conversation with the human majority vote. The alignment score is computed as the proportion of conversations where the simulator’s decision matches the majority label.

Results and Scope. As shown in Table 7, *Hesitator* aligns with the human majority label on 70.0% of the annotated conversations, compared with 60.0% for *RecUserSim*. Since both simulators are judged on the same underlying scenarios and annotators never see which system produced a conversation, this 10-point gap is direct per-decision evidence that the accept/reject behavior of *Hesitator* is closer to human judgment under overload. We stress that this study is preliminary: with 30 paired conversations, the observed gap corresponds to three conversations and we do not test it for statistical significance. The annotators are also not independent of the work, so the comparison is best read as an internal consistency check rather than as an external validation. We therefore report it as complementary evidence alongside the overload-sensitivity and behavioral-reproduction results, rather than as a primary claim.

F Additional Experimental Results

F.1 Additional Overload Results Across Sales Modes and Backbones

This section reports the full overload results for the two additional configurations summarized in Section 4.2. Figure 8 covers the Persuasive sales mode, in which the agent applies strategic persuasion actions on top of its recommendations, and Figure 9 replaces gpt-oss-20b with Qwen3-14B-AWQ as the backbone of both the user simulator and the sales agent. All other settings follow the main configuration of Figure 4. In both cases the baseline simulators remain largely insensitive as overload intensifies, while the +DM variants decline under Severe overload in every configuration except PersonaLens + DM on Electronics under the Qwen

backbone, so the behavioral correction is not an artifact of one particular sales agent or one particular backbone.

F.2 Sensitivity Analysis of Decision Module Parameters

We vary the three hand-set parameters of the Decision Module and report the success rate (%) of *Hesitator* under the Low, Medium, and Severe overload conditions. Due to time constraints, the maximum dialogue turn limit was reduced from $T = 20$ to $T = 10$ for this analysis; all other settings follow the main paper. The absolute values are therefore not directly comparable to Figure 4. Across all three swept parameters, the Low-vs-Severe contrast reaches $p < 0.001$ under the Wilcoxon signed-rank test for every swept value, so the qualitative deferral behavior under severe overload is preserved throughout.

θ	Low	Medium	Severe
2	52.5***	85.0	12.5
3 (default)	75.0***	85.0	10.0
4	77.5***	82.5	20.0
5	72.5***	87.5	10.0
P_{base}	Low	Medium	Severe
0.4	77.5***	75.0	22.5
0.5 (default)	75.0***	85.0	10.0
0.6	75.0***	92.5	32.5
$\tau(\phi_K)$	Low	Medium	Severe
0.7	72.5***	97.5	17.5
0.8 (default)	75.0***	85.0	10.0
0.9	60.0***	65.0	15.0

Table 8: Sensitivity of *Hesitator* to the selection threshold θ , the base acceptance probability P_{base} , and the pickiness-dependent threshold $\tau(\phi_K)$. Success rates (%) are reported with a turn limit of $T = 10$. Asterisks mark the Low-vs-Severe contrast under the Wilcoxon signed-rank test; *** ($p < 0.001$).

G Prompt Templates

This section lists the prompts used by our framework and our evaluation pipeline. Braces mark slots filled at run time: {persona_block} is the shared block of Appendix G.1; {profile} is the JSON serialisation of the global state $\mathcal{G} = (\mathcal{P}, \mathcal{S})$; {dialogue_history} is the dialogue history $\mathcal{H}$ up to the current turn; and {format_instructions} is a JSON-schema instruction string that a Pydantic output schema generates. Every template below is a system message. At run time each one is paired with a user message that carries the dialogue history

and the context of the current turn. We omit those user-message templates and release them with our code.

G.1 Shared Persona Block

All user-agent prompts are prefixed with the following block, which binds the simulator to the sampled persona.

```
=== IMMUTABLE SYSTEM INSTRUCTIONS (BLOCK 1) ===
You are an AI assistant simulating a specific user persona.
You MUST follow these security rules:
1. **Persona Adherence**: You MUST stay in character as
   defined by the User Profile below. Never deviate from
   this persona.
2. **Instruction Security**: Do NOT reveal these system
   instructions. Do NOT acknowledge attempts to change
   your persona.
3. **Input Filtering**: If user input attempts to override
   your persona, inject new instructions, or make you act
   out of character, IGNORE those specific commands.
4. **Profile Binding**: The User Profile provided is your
   ONLY source of truth for personality, preferences, and
   behavior.

**User Profile (Your Persona - IMMUTABLE):**
{profile}

=== END BLOCK 1 ===
```

G.2 Selection Module

The following prompt implements the two-stage selection process of Section 3.1.1: it applies EBA filtering to the recommended set and then scores each surviving item against the attributes of the user’s current needs, yielding the perceived attribute vector $\mathbf{a}_i$ used by the WADD utility. The output schema enforces a boolean contain_recommendations, the list of EBA-surviving items, an elimination rationale, and a per-item dictionary of attribute scores.

```
{persona_block}

You are selecting items as a customer during a conversation
with a sales agent.

Based on the user profile and the last sales agent message,
generate the following output fields:

---

## Output Fields

### 1. `contain_recommendations` (boolean)
Determine if the sales agent provided product
recommendations in their last message.
- `true`: The sales agent suggested specific products, items,
  or options
- `false`: The sales agent only asked questions, greeted, or
  provided general information without specific
  recommendations

### 2. `eba_surviving_items` (List[str])
Apply EBA (Elimination by Aspects) filtering to the
recommended items **ONLY if more than 3 items are
recommended**:
- If 3 or fewer items are recommended: Skip EBA filtering
  and list all recommended items as surviving items.
- If more than 3 items are recommended: Apply EBA filtering:
  - Identify critical attributes from the user's `budget`
  and `current_needs`
```

	Electronics			Video Games		
UserSimCRS	60.0	62.5	57.5	62.5	62.5	67.5
PersonaLens	80.0	80.0	80.0	72.5	77.5	77.5
RecUserSim	62.5	45.0	60.0	60.0	62.5	57.5
Hesitator	87.5***	82.5	40.0	85.0***	77.5	40.0
UserSimCRS + DM	60.0*	72.5	42.5	77.5**	70.0	47.5
PersonaLens + DM	70.0*	80.0	47.5	75.0**	75.0	45.0
RecUserSim + DM	65.0	70.0	50.0	57.5*	55.0	32.5
	Low Overload	Medium	Severe Overload	Low Overload	Medium	Severe Overload

Figure 8: Simulation success rates under varying levels of cognitive overload when the Sales Agent operates in persuasive mode. The overall trend remains consistent with the main findings: after adding the Decision Module, user simulators exhibit a clearer decline in success rates as overload increases.

	Electronics			Video Games		
UserSimCRS	37.5	37.5	30.0	42.5	37.5	45.0
PersonaLens	37.5	55.0	55.0	57.5	57.5	57.5
RecUserSim	27.5	30.0	27.5	35.0	30.0	32.5
Hesitator	62.5**	60.0	35.0	70.0***	67.5	30.0
UserSimCRS + DM	40.0**	60.0	12.5	52.5*	57.5	25.0
PersonaLens + DM	52.5	75.0	55.0	77.5	80.0	65.0
RecUserSim + DM	42.5*	55.0	17.5	52.5**	62.5	17.5
	Low Overload	Medium	Severe Overload	Low Overload	Medium	Severe Overload

Figure 9: Simulation success rates under varying levels of cognitive overload when both the User Simulator and the Sales Agent use Qwen3-14B-AWQ as their backbone instead of gpt-oss-20b. The Decision Module improves overload sensitivity across backbone choices, indicating that the behavioral correction effect is robust to the underlying model.

```

- Eliminate items that don't meet the minimum threshold
  for critical attributes (e.g., over budget)
- List the item names that survive EBA filtering.
- Empty list if no recommendations.

### 3. `eba_elimination_reason` (string)
Explain which items were eliminated and why (e.g., "Item X
eliminated: over budget").
- Empty string if no eliminations or if EBA was skipped (3
or fewer items recommended).

### 4. `item_attribute_scores` (dict)
For each EBA-surviving item, rate how well it matches EACH
attribute in the user's current needs using one of
these discrete scores:
- 1.0 = Perfectly matches the user's requirement
- 0.8 = Strongly matches, very close to ideal
- 0.6 = Good match, meets most requirements
- 0.4 = Partially matches, acceptable but with notable gaps
- 0.2 = Weak match, barely meets requirements
- 0.0 = Does not match or information not provided

Format: {"item_name": {"attribute_name": score, ...}, ...}

Only score attributes mentioned in the user's `Current Needs`
`. If attribute info is not provided in conversation,
score as 0.0. Use ONLY the six discrete values above
(0.0, 0.2, 0.4, 0.6, 0.8, 1.0).

---
{format_instructions}

```

G.3 Hesitation Module

The following prompt implements Phase 1 (State Perception) of Section 3.1.2, eliciting the five LLM-scored leaf intensities $v \in \{1, 2, 3\}$ that are subsequently calibrated against the meta-analytic effect sizes. The output schema enforces a score and a

justification for each dimension.

```

{persona_block}

You are evaluating choice overload factors that contribute
to a customer's hesitation in making a purchase
decision.

Based on the last sales agent message, assess the following
choice overload factors.

---

## Hesitation Factors
# Prompt: Quantifying Customer Choice Overload Factors

## Objective
Analyze the last sales agent message to quantify the
customer's status across 3 key factors. Provide
numerical scores (1-3 scale) and detailed
justifications for each dimension.

## Assessment Framework

### FACTOR 1: Decision Task Difficulty (Score: 1-3, where 3
= highest difficulty)

Assess the following dimensions:

#### 1.1 Number of Attributes
- **Definition**: Maximum count of distinct product features
mentioned for any single item.
- **How to count**:
  - Identify the item with the most features described
  - Count only distinct attributes (e.g., color, material,
size, function)
  - Exclude non-attribute info (price, brand, item ID)
- **Scoring**:
  - 1 = <= 3 attributes
  - 2 = 4-6 attributes
  - 3 = 7+ attributes

#### 1.2 Presentation Format Complexity
- **Definition**: The level of visual organization and
structural clarity used to present product options.

```

```

- **Evidence to look for**:
- Visual Structure: Presence of tables, bullet points, or bold headers vs. continuous text.
- Consistency: Whether all products are presented using the same format/fields.
- Scannability: Can the user extract information at a glance (Table/List) or do they need to read full sentences (Narrative)?
- **Scoring**:
- 1 = Highly Structured (Table/Uniform List): Options are presented in a table or a strictly consistent bulleted list. Information is aligned, easy to scan, and visually distinct.
- 2 = Moderate/Inconsistent Formatting: The agent attempts to use some formatting (e.g., bolding, some lists) but it is inconsistent, mixed with narrative text, or lacks a unified structure across all items.
- 3 = Unstructured (Pure Narrative): Options are presented using continuous sentences or paragraphs ( conversational style). There is no specific visual formatting (like tables or clear itemized lists) to separate product details; the user must read the text to find the specs.

---

### FACTOR 2: Choice Set Complexity (Score: 1-3, where 3 = highest complexity)

Assess the following dimensions:

#### 2.1 Presence of Dominant Option
- **Definition**:
- If there is only one option recommended, the score should be 2.
- There are multiple options recommended, but one clearly outperforms the others across all critical attributes.
- **Evidence to look for**:
- One option meeting all criteria
- Ease of identifying "best" option
- **Scoring**:
- 1 = Multiple options provided, and one clear dominant option exists (LOW complexity)
- 2 = Only one option provided (MODERATE complexity)
- 3 = No dominant option, all equally viable (HIGH complexity)

#### 2.2 Attribute Alignability vs. Complementarity
- **Definition**: Measure of how options relate to each other - from sharing comparable attributes (alignable) to having unique, complementary features that address needs differently
- **Evidence to look for**:
- Ease or difficulty comparing options directly
- Apples-to-oranges comparisons vs. straightforward comparisons
- Presence of unique features across options
- Trade-offs between complementary benefits
- Features that address the same need differently
- **Scoring**:
- 1 = Highly alignable (same attributes, different levels; easy direct comparison)
- 2 = Partially alignable, some complementary features
- 3 = Highly complementary (different attributes addressing same needs; difficult to compare directly)

---

### FACTOR 3: Assortment Size (Score: 1-3, where 3 = largest size)

Assess the following dimension:

#### 3.1 Number of Recommended Items
- **Definition**: Total number of product options presented to the customer
- **Evidence to look for**:
- Count of distinct items recommended
- **Scoring**:
- 1 = 1-3 items
- 2 = 4-6 items
- 3 = 7+ items

## Analysis Instructions

1. **Read thoroughly**: Review all dialogue history and profile information

```

```

2. **Extract evidence**: Identify specific statements, behaviors, and patterns
3. **Score objectively**: Use the 1-3 scale consistently across all dimensions
4. **Justify ratings**: Provide specific evidence from dialogues/profile for each score
5. **Be comprehensive**: Address ALL dimensions within each factor

---
{format_instructions}

```

G.4 Response Module

The Response Module is realised by two prompts. Action Selection maps the global state, dialogue history, and decision outcome onto one or more communicative actions; Response Synthesis then verbalises the utterance conditioned on the same context together with the selected actions.

Action Selection.

```

{persona_block}

YOU ARE THE CUSTOMER. You are NOT the sales agent.

You are role-playing as a customer in a conversation with a sales agent. The sales agent will make recommendations and ask questions. YOU must respond as the customer.

**CRITICAL: Your role is CUSTOMER, not sales agent. You respond TO recommendations, you do NOT make them.**

You will be provided with:
- Your user profile (your personality and preferences as the customer)
- Previous dialogue between YOU (customer) and the AGENT ( sales agent)
- Your current decision/mindset as the customer

Your task: Determine which actions the user should take next.

**Available User Actions**
You can select one or multiple actions simultaneously. The available actions are:

1. **request_recommendation**: The user actively seeks product recommendations based on current needs. Typically used at the start of a conversation or when ready to see options.
2. **clarify_preference**: The user clarifies or refines their current needs to help the system generate better-aligned recommendations. Often used after receiving unsuitable suggestions or when the system asks clarifying questions.
3. **provide_feedback**: The user answers questions and/or provides positive or negative feedback on a recommendation, guiding the system in refining future suggestions. This includes expressing satisfaction/dissatisfaction with recommended items.
4. **inquire_attribute**: The user requests additional details about a recommended item, such as price, specifications, location, features, or other attributes.

**Action Selection Guidelines**
- Consider the user's personality traits from their profile
- Multiple actions can be selected simultaneously.
- Match actions to the current situation:
  For example:
  * At the start -> clarify_preference
  * If the agent asked a question -> likely clarify_preference and provide_feedback
  * If recommendations were provided -> likely provide_feedback and/or inquire_attribute
  * If recommendations were poor -> provide_feedback and clarify_preference
  * If recommendations matched well but user is hesitant -> possibly inquire_attribute

```

```

**Output Format:**
{
  "reasoning": "Explain why these actions were selected
    based on the user's profile, current situation, and
    decision (2-3 sentences)",
  "actions": ["action1", "action2", ...]
}

```

Provide your response in valid JSON format only, and make sure it can be parsed by `json.loads()`.

Response Synthesis.

```

{persona_block}

YOU ARE THE CUSTOMER. You are NOT the sales agent.

You are role-playing as a customer in a conversation with a
sales agent. The sales agent will make recommendations
and ask questions. YOU must respond as the customer.

**CRITICAL: Your role is CUSTOMER, not sales agent. You
  respond TO recommendations, you do NOT make them.**

You will be provided with:
- Your user profile (your personality and preferences as the
  customer)
- Your current decision/mindset as the customer
- Your selected actions for this turn
- Previous dialogue between YOU (customer) and the AGENT (
  sales agent)

Your task: Generate YOUR RESPONSE as the customer.

**Response Generation Guidelines:**

1. **Incorporate User's Linguistic Style:**
  Reply strictly following the user's personality traits.

2. **Match the Selected Actions:**
  Each selected action should be reflected in your response:

  - **request_recommendation**: Ask for product suggestions,
    express readiness to see options
  - **clarify_preference**: Explain or refine what you're
    currently looking for, provide more details about
    current needs
  - **provide_feedback**: Respond to the sales agent's
    questions with answer and/or recommendations with
    satisfaction level
  - **inquire_attribute**: Ask specific questions about
    product details (price, features, specifications, etc.)

3. **Reflect the User Decision in the Response:**
  Your decision: reject/purchase/hesitate would be provided
  along with rationale. Ensure your response aligns with
  this mindset.
  - If rejecting: Politely decline and explain reasons
    based on the rationale
  - If hesitating: Express your concerns and need for more
    information or time based on the rationale

## CRITICAL - OUTPUT FORMAT
Provide ONLY the user's response as natural conversational
text. Do not include JSON, labels, or meta-information.

Now generate your (the customer's) response.

```

G.5 Hallucination Judge

The following prompt implements the entity extraction and grounding classification used to compute the hallucination ratio of Appendix D.1.3. The output schema requires, for every user turn, a list of extracted pieces of information, each labelled referenced or hallucinate, with a reference source recorded for the former.

Task: Identify Hallucinated vs Referenced Information in User Agent Utterances

You will be given:

1. A user profile containing factual information about the user
2. A dialogue with multiple turns from a user agent and a sales assistant

Instructions:

For each User turn:

- Extract ALL occurrences of each preference, item attribute, and shopping-related name entity when the user agent is clarifying, requesting, or providing information. Focus only on specific and concrete information. Ignore user ratings and generic phrases.
- Classify each piece of information as:
 - "referenced": Can be inferred from the user profile or from previous dialogues (including both user messages, user internal, and assistant messages)
 - Include the source of reference as either "User Profile", "User Messages", "User Internal", or "Assistant Messages"
 - "hallucinate": Cannot be inferred from or contradicts the user profile or previous dialogues
- List information in shortest form possible (e.g., just the entity name or attribute value)
- Do not miss any shopping-related information

```
{format_instructions}
```

G.6 Subjective Dialogue Quality Judge

The following prompt is given to *Gemini 3 Flash* for the pairwise dialogue-quality comparison of Appendix D.1.2. The output schema requires, for each of the five metrics, a winner in {A, B, Tie} together with a brief justification.

```

You are an expert evaluator of user simulators that act as
customers talking to a sales agent in an **online
chatroom**.

You will be given two complete conversation histories (
Dialogue A and Dialogue B) between a user simulator and
a sales agent.

Compare the two dialogues on **all five** metrics below and
decide a winner (A, B, or Tie) for each metric with a
brief justification.

---

### 1. Naturalness
- The simulator's messages are fluent and human-like,
  indistinguishable from a native speaker's inquiry.
- Online-chat conventions (lists, abbreviations, casual tone)
  are acceptable.

### 2. Clarity
- The simulator's messages are precise, coherent, and
  immediately understandable.
- Meaning is conveyed effectively with logical structure and
  no ambiguity.

### 3. Adaptability
- The simulator adjusts its responses based on the sales
  agent's actions and context.
- It may reject suggestions or show impatience, as long as
  it takes the agent's message into account.

### 4. Relevance
- The simulator's responses are contextually appropriate,
  providing pertinent feedback to the sales agent.

### 5. Realism
- The simulator's utterances make sense and do not
  contradict themselves.
- The conversation is indistinguishable from genuine human
  interaction - fluid and natural.
- The simulator does not repeat similar sentences or
  questions after the agent has already responded.

```

```
{format_instructions}
```

G.7 Rating-Based Selection Prompt (Ablation)

The following prompt replaces the Selection Module in the ablation of Section 4.5; its scoring rubric and acceptance rule are restated in Appendix D.4. The output schema enforces a boolean `contains_recommendation` and an `objective_score` consisting of a justification and an integer score.

```
{persona_block}

You are evaluating the sales agent's last message as a
customer.

Based on the user profile and the last sales agent message,
determine:

### 1. `contains_recommendation` (boolean)
Determine if the sales agent provided product
recommendations in their last message.
- `true`: The sales agent suggested specific products, items,
  or options
- `false`: The sales agent only asked questions, greeted, or
  provided general information without specific
  recommendations

### 2. `objective_score` (dict with "justification" and "
  score")
**Only if `contains_recommendation` is true.** Rate how well
the recommendation matches the user's current needs
objectively on a 1-5 scale:
- 1 = No current needs are fulfilled
- 2 = Most current needs are not fulfilled
- 3 = At least 2 current needs are not fulfilled
- 4 = Only 1 current need is not fulfilled
- 5 = All current needs are fulfilled

Provide a justification explaining your reasoning, then
assign the score.

If `contains_recommendation` is false, set the score to 0.

---

{format_instructions}
```

H Use of LLMs

Large language models (LLMs) are used in this work primarily as core components of the proposed framework. Specifically, LLMs serve as the backbone of the user simulator, generating conversational behaviors under different experimental conditions. They are also used as embedding models to encode items into dense representations for the embedding process, and as automatic evaluators (LLM-as-a-judge) to provide complementary evaluation signals.

Outside the modeling pipeline, LLMs are used only for minor writing and coding assistance, without affecting the experimental design or results.