Skill-CMIB: Multimodal Agent Skill for Consistent Action via Conditional Multimodal Information Bottleneck

Anonymous ACL submission

Abstract

LLM-based multimodal agents often produce inconsistent action trajectories across trials, limiting their reliability in long-horizon environments. Improving consistency requires reusable skills that preserve task-relevant invariants from trial-and-error trajectories while suppressing trajectory-specific noise. Yet multimodal skills must reconcile interpretable, verbalizable procedures with dense perceptual evidence that text alone cannot capture: text-only skills discard such cues, whereas naive two-stream storage introduces redundancy and noise. We introduce the **Conditional Multimodal Information Bottleneck (CMIB)**, a principled framework that decomposes a joint multimodal skill bottleneck into an interpretable text-stage skill card and a conditional multimodal bottleneck retaining only residual perceptual information predictive beyond text. By conditioning the multimodal latent on the text skill, CMIB reduces cross-modal redundancy and separately controls textual and perceptual compression. A tractable variational objective instantiates CMIB, yielding reusable multimodal skills that improve execution stability without multi-sample inference overhead.

1 Introduction

Multimodal LLM agents increasingly act in environments where successful decisions depend on both language and perception, including web navigation (Deng et al., 2023; Zhou et al., 2024), GUI control (Rawles et al., 2023; Zhang et al., 2024a; Nguyen et al., 2025), and multimodal decision making (Driess et al., 2023; Brohan et al., 2023; Wu et al., 2024b,d, 2025d). Yet the same task can yield markedly different action sequences across trials under an unchanged policy (Shinn et al., 2023; Xia et al., 2025). Self-consistency reduces such variability by sampling and aggregating multiple trajectories (Wang et al., 2023; Wu et al., 2024c, 2025a, 2026; Yu et al., 2025), but incurs substantial decoding cost and does not produce reusable procedures (Wang et al., 2025b). We therefore seek compact multimodal skills that distill task-relevant invariants from trial-and-error experience and improve consistency without repeated resampling.

Agent skills provide a natural interface for reusable control because they condition planning, tool use, and action selection without modifying the underlying task model (Zhang et al., 2025; Xu and Yan, 2026a; Wu and Zhang, 2026). However, multimodal skill construction must separate reusable procedural structure from trajectory-specific noise (Yan et al.; Li et al., 2025; Wu et al., 2025e) while handling signals distributed unevenly across text and vision. Text-only skill cards are interpretable and easy to index (Zhang et al., 2025, 2026; Huang et al., 2026; Wang et al., 2026), but may discard dense visual evidence that cannot be faithfully verbalized. Naive two-stream storage, by contrast, lacks a principled criterion for what belongs in the symbolic channel versus the latent multimodal channel, leading to redundant, noisy, or poorly transferable skills (Stepputtis et al., 2020).

We introduce **Skill-CMIB**, a multimodal skill construction framework grounded in a *Conditional Multimodal Information Bottleneck (CMIB)*. As illustrated in Figure 1, CMIB starts from a joint bottleneck over multimodal skills and derives a sequential decomposition into a text-stage bottleneck for an interpretable skill card and a conditional multimodal bottleneck for residual information predictive beyond the card. The text card serves as the symbolic interface for retrieval, indexing, and explanation, while the multimodal latent is injected as soft control tokens to preserve complementary perceptual evidence (Li et al., 2023; Nguyen et al., 2024; Radford et al., 2021; Goyal et al., 2019b). Because the latent is conditioned on the text skill, CMIB discourages information already explained by text and yields skills that are sufficient for task prediction, minimal under compression, and com-

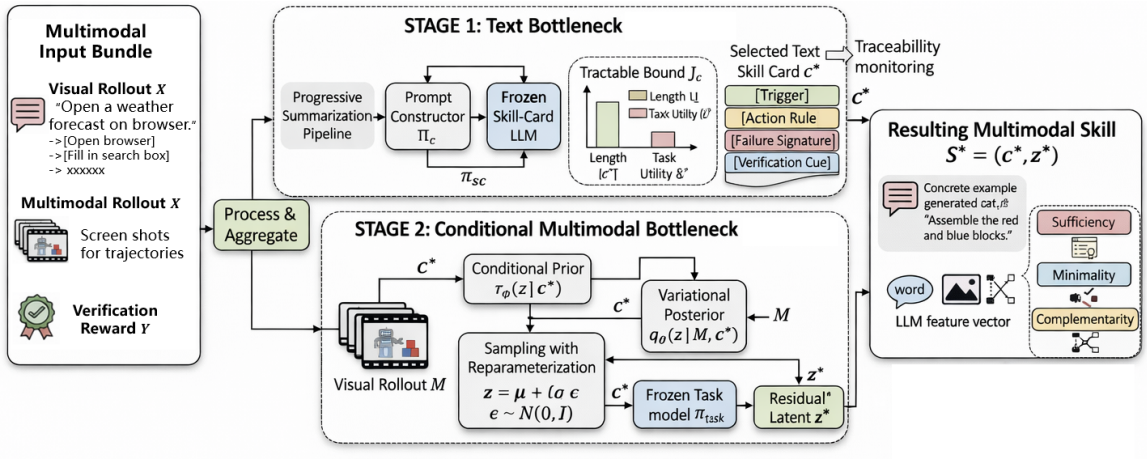

Figure 1: Illustration of the Skill-CMIB pipeline. Given a multimodal dataset, Skill-CMIB first collects trial-and-error trajectories, then applies **Stage 1 Text Bottleneck** to extract and select text skill cards. **Stage 2 Conditional Multimodal Bottleneck** further derives a residual multimodal latent conditioned on the text card, and the two stages are combined into a CMIB multimodal skill library.

plementary across modalities.

We further derive tractable information bounds for this formulation (Wu et al., 2022; Liu et al., 2025): the text stage uses prompted skill-card generation under a utility–length trade-off, and the conditional multimodal stage uses a variational posterior, a card-conditioned prior, and a lightweight projection into the frozen task model (Tishby et al., 2000; Poole et al., 2019; Mahabadi et al., 2021; Goyal et al., 2019a; Wu et al., 2023; Huang et al., 2025d). Following the scope of Anthropic-style skills, we evaluate Skill-CMIB on `Multimodal-Mind2Web`, `ScreenSpot`, and `AndroidControl`; across direct prompting, self-consistency, and text-only skill baselines, it improves task success and action consistency without repeated multi-sample decoding at deployment.

Our contributions are:

- We formulate *multimodal agent skills* information-theoretically, organizing reusable procedural structure and task-relevant perceptual evidence into complementary channels.
- We propose **CMIB**, a conditional decomposition of a joint bottleneck into an interpretable text skill card and a residual multimodal latent, with a tractable variational realization for frozen backbone agents.
- We validate **Skill-CMIB** on web, GUI, and mobile-control benchmarks, improving success rate and action consistency over prompting, self-consistency, and text-only skill baselines.

2 Related Works

2.1 Agent Skills

Recent work formalizes *agent skills* as modular, reusable procedures that extend agents beyond atomic tool calls, spanning product-oriented descriptions (Zhang et al., 2025; Nguyen et al., 2025; Wu et al., 2025d), systematic perspectives on skills versus tools (Jiang et al., 2026b), and surveys on architecture, acquisition, and procedural-memory views of skills (Xu and Yan, 2026a; Wu and Zhang, 2026; Huang et al., 2025b,a; Wu et al., 2024a). A parallel line learns skills or procedural memory from interaction traces via hierarchical memory, reinforcement learning over skill libraries, non-parametric procedural memory, or personalized adaptation of LLMs (Fang et al., 2025; Wang et al., 2025a; Mi et al., 2026; Jiang et al., 2026a; Zhang et al., 2024c; Xie et al., 2025; Ni et al., 2026; Wang et al., 2025c). These efforts establish *what* a skill library is and how skills are acquired, but typically expose skills as text or opaque parameters (Wu et al., 2024d,b), leaving the joint compression of interpretable language and complementary multimodal perceptual evidence underexplored.

2.2 Behavioral Reliability in LLM-Based Agents

LLM-based agents exhibit trial-to-trial variability and measurable self-disagreement (Mehta, 2026; Xia et al., 2025; Shinn et al., 2023; Wang et al., 2025b). The standard remedy, self-consistency and its variants, samples and aggregates multiple outputs (Wang et al., 2023, 2022; Aggarwal et al.,

2023; Wang et al., 2024), which improves robustness but multiplies decoding cost in sequential settings.

Alternatives *internalize* consistency via post-training (Wu et al., 2024c; Samanta et al., 2026; Kveton et al., 2025; Wu et al., 2025a), while the information bottleneck (Wu et al., 2022; Liu et al., 2025; Tishby et al., 2000) and variational surrogates (Poole et al., 2019; Mahabadi et al., 2021) instead compress experience into sufficient minimal representations, with instantiations in RL and soft control (Kveton et al., 2025; Wu et al., 2025b; Huang et al., 2025c; Surana et al., 2026; Mundada et al., 2026; Goyal et al., 2019b; Wu et al., 2023; Huang et al., 2025d). Skill-CMIB follows this philosophy for multimodal skills: it avoids repeated voting at deployment and structures compression so a text skill card and a conditional multimodal latent remain complementary (Wang et al., 2026; Wu et al., 2021; Li et al., 2025; Radford et al., 2021; Stepputtis et al., 2020).

3 Skill-CMIB: Multimodal Skills via Conditional Multimodal Information Bottleneck

We consider a multimodal agentic system that produces trial-and-error trajectories through interaction with an environment. We extend the agent skill rollout process in Equation (17) to multimodality:

$$\tau^{(k)} = \big(x_t^{(k)},\ m_t^{(k)},\ a_t^{(k)}, o_t^{(k)}\big)_{t=1}^{T_k}, \tau^{(k)} \in \mathcal{B} \quad (1)$$

where x_t denotes textual prompts, m_t denotes visual pixel information, a_t is the agent's action, o_t is the environment response.

Definition 3.1 (Multimodal Agent Skill)**.** *Let X and M denote the aggregated textual and multimodal content extracted from a set of rollout trajectories, and let Y denote the supervision target corresponding to the verifiable reward. A* reusable skill *is a structured representation of (X, M) defined as*

$$S = (c, z) \sim p_\psi(S \mid X, M),\ c \in \mathcal{C}, z \in \mathbb{R}^d, \quad (2)$$

where c is a natural-language text skill card *used for retrieval, indexing, and coarse control guidance, and z is a* multimodal latent vector *that retains dense perceptual details beyond what can be faithfully verbalized. In this sense, $S = (c, z)$ is a reusable latent skill distilled from (X, M) and intended to retain the task-relevant information needed to predict Y.*

The multimodal skill pair (c, z), at deployment, is fused and injected into a frozen task LLM as soft tokens (Li et al., 2023; Nguyen et al., 2024). Intuitively, the trajectory signal decomposes into: (1) the **invariant skill mechanism**, which is a reusable, transferable structure determining what should be done as this skill; (2) and **nuisance variation**, which is instance-specific details and noise from both modalities. The desired multimodal skill should satisfy $Y \perp\!\!\!\perp (X, M) \mid S$ (sufficiency), depend primarily on the invariant task skill information, and expose a text interface for retrieval. Thus, the multimodal skill optimization is to enable the two modalities of S jointly achieve sufficiency and invariance while remaining complementary.

3.1 Conditional Multimodal Information Bottleneck (CMIB)

We extend the information bottleneck in Equation (18) to a joint bottleneck over the multimodal skill $S = (c, z)$ defined in Equation (2):

$$\mathcal{L}_{\text{joint}} = I\big((X, M);\ (c, z)\big) - \beta\, I\big((c, z);\ Y\big). \quad (3)$$

This objective compresses the rollout content (X, M) while preserving the task-relevant information needed to predict the verifiable target Y (illustrated in Figure 2). The following lemma shows that Equation (3) admits an exact two-stage factorization, separating a text-stage bottleneck in c from a conditional multimodal bottleneck in z given c.

Lemma 3.2 (Factorization underlying CMIB)**.** *The objective in Equation (3) admits an exact decomposition into a text-stage term involving c and a conditional multimodal term involving z given c. Motivated by this factorization, we define the Conditional Multimodal Information Bottleneck (CMIB) by introducing stage-specific trade-off coefficients:*

$$\mathcal{L}_{\text{CMIB}} = \Big[I\big((X, M); c\big) - \beta_c\, I(c; Y)\Big] + \Big[I\big((X, M); z \mid c\big) - \beta_z\, I(z; Y \mid c)\Big]. \quad (4)$$

When $\beta_c = \beta_z = \beta$, Equation (4) reduces to the exact factorization of Equation (3).

The proof is a direct consequence of the chain rule of mutual information as in Appendix D.

Equation (4) directly encodes the three design requirements of multimodal skill construction: **Sufficiency** is captured by the relevance terms $I(c; Y)$ and $I(z; Y \mid c)$, which preserve the information

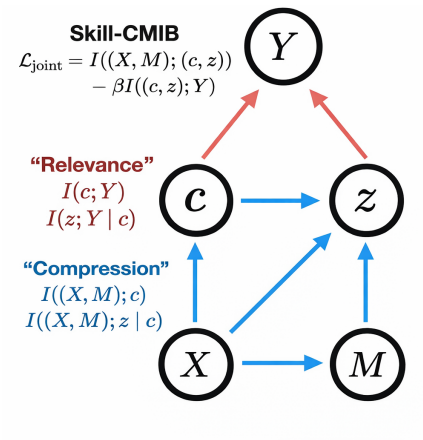

Figure 2: Skill-CMIB illustration.

needed to predict Y, with the conditional term measuring the residual predictive value beyond the text card. **Minimality** is imposed by the compression terms $I((X, M); c)$ and $I((X, M); z \mid c))$, which suppress instance-specific noise, with the latter compressing only information not already encoded in c. **Complementarity** follows from conditioning the multimodal latent on c. The text card c, optimized through the unconditional bottleneck $I((X, M); c)) - \beta_c I(c; Y)$, captures reusable semantics of the rollout and serves as the interface for indexing, retrieval, and explanation in skill libraries. In turn, the conditional bottleneck $I((X, M); z \mid c)) - \beta_z I(z; Y \mid c)$ drives z to retain only residual multimodal evidence that remains useful once c is known (Li et al., 2026b; Xu and Yan, 2026b).

CMIB also suggests a natural representation strategy for the agentic workflow. At construction time, c and z are stored separately because they are produced by different encoders and serve different roles: c provides the symbolic interface for skill management, whereas z preserves perceptual detail that is inefficient to verbalize. At inference time, the multimodal information z is projected in to skill prefix,

$$u = g_\omega(z), \tag{5}$$

where g_ω is a lightweight projection module. This separation makes the bottleneck traceable: the statistics of the text and the conditional multimodal stream can be monitored independently, while the frozen model still receives unified control signals at inference time.

We now instantiate this view through **tractable bounds on information** in a multimodal LLM agent. Because the information bottleneck terms in Equation (4) are not directly tractable in practice (Tishby et al., 2000; Poole et al., 2019; Mahabadi et al., 2021; Huang et al., 2025d), we replace them with computable surrogates that track each component of CMIB (Goyal et al., 2019a; Wu et al., 2023). We begin with the **text bottleneck**, then derive the **conditional multimodal bottleneck**, and combine the two into a unified training objective.

3.2 Tractable Bound for the Text Bottleneck

The text bottleneck aims to extract a compact skill card c that preserves the task-relevant procedural content of the rollout bundle while remaining short enough to act as the symbolic interface of the skill library. Formally, the text-stage term of CMIB is

$$\mathcal{L}_c = I((X, M); c) - \beta_c I(c; Y). \tag{6}$$

Here, the compression term discourages unnecessary dependence of c on the rollout content, while the relevance term encourages c to retain information predictive of the verifiable target Y. Since c is generated by a frozen LLM rather than a trainable probabilistic encoder, we do not optimize Equation (6) directly. Instead, we define the feasible candidate set and use card length only as a rate proxy, not as a direct estimate of $I((X, M); c)$. Under a fixed tokenizer and prompt format, the token length $|c|$ upper-bounds the description length of the selected text card up to a constant bits-per-token factor. This follows recent retrieval and agentic frameworks that make information-seeking objectives tractable through variational bounds or discrete candidate selection (van den Oord et al., 2019; Yao et al., 2023; Wu et al., 2025c), rather than direct mutual-information optimization.

$$\begin{aligned} \mathcal{J}_c(c; \mathcal{B}) &= |c| - \beta_c \widehat{U}(c; \mathcal{B}), \\ c^* &= \arg \min_{c \in \mathcal{C}_{L_c}(X)} \mathcal{J}_c(c; \mathcal{B}), \end{aligned} \tag{7}$$

where $|c|$ denotes the length of c (e.g., number of tokens) and $\widehat{U}(c; \mathcal{B})$ is the task-specific utility score of card c evaluated on the rollout bundle $\mathcal{B}$ in Equation (1). Let $\mathcal{C}_{L_c}(X) \subseteq \mathcal{C}$ denote the set of candidate skill cards obtainable from the aggregated rollout text X under length budget L_c. In this formulation, the objective $\mathcal{J}_c$ tracks the relevance term $I(c; Y)$, while restricting c to $\mathcal{C}_{L_c}(X)$ enforces the compression budget associated with $I((X, M); c)$.

Because Equation (7) optimizes over discrete skill cards produced by a frozen LLM, we realize it

through prompting rather than direct optimization. In practice, the optimization is implemented as selecting a prompted card from candidate set.

Let $\Pi_c(X, L_c)$ denote the prompt constructor that maps the aggregated rollout text X and the length budget L_c to a skill-card generation prompt, and let

$$\begin{aligned} c^* &\sim \pi_{\text{sc}}(\cdot \mid X, L_c) \\ &= \pi_{\text{sc}}(\cdot \mid \Pi_c(X, L_c)). \end{aligned} \tag{8}$$

be the induced generation distribution under the frozen LLM. The prompt $\Pi_c(X, L_c)$ is instantiated by a **progressive summarization pipeline**: trajectory-level evidence is first extracted from each rollout, then aggregated across the K rollouts, and finally formatted into a structured card. The selected output c^* from Equation (8) is thus the tractable approximation to the original text bottleneck in Equation (6), and serves as the discrete interface used later for retrieval, indexing, and explanation.

3.3 Tractable Bound for the Conditional Multimodal Bottleneck

Since the text stage has already produced the discrete component c^*, the remaining problem is to construct the residual multimodal latent z conditioned on that fixed skill card. Under this realization, the conditional multimodal term in Equation (4) becomes

$$\mathcal{L}_z = I((X, M); z \mid c^*) - \beta_z I(z; Y \mid c^*). \tag{9}$$

The following lemma shows that this objective admits a tractable variational surrogate based on a text-conditioned posterior encoder and prior.

Lemma 3.3 (Variational surrogate of the conditional multimodal bottleneck)**.** *Fix a text card c^*, and suppose the latent variable z is generated by the encoder $q_\theta(z \mid M, c^*)$, so that $Z \perp X \mid (M, c^*)$. Let g_ω be the projection map introduced in Equation (5), which maps the multimodal latent z into the control space of the frozen task model π_{tsk}. For any conditional prior $r_\phi(z \mid c^*)$, define*

$$\begin{aligned} \mathcal{J}_z(\theta, \phi; c^*) = \mathop{\mathbb{E}}_{\substack{(M,Y)\sim p(\cdot,\cdot\mid c^*) \\ z\sim q_\theta(\cdot\mid M, c^*)}} \Bigg[\log \frac{q_\theta(z \mid M, c^*)}{r_\phi(z \mid c^*)} \\ - \beta_z \log \pi_{\text{tsk}}\left(Y \mid [\, g_\omega(z);\ c^*;\ \mathcal{B}\,]\right) \Bigg]. \end{aligned} \tag{10}$$

Then

$$\mathcal{L}_z \ \le\ \mathcal{J}_z(\theta, \phi; c^*) - \beta_z H(Y \mid c^*). \tag{11}$$

Equivalently, up to the additive constant $\beta_z H(Y \mid c^*)$, the surrogate $\mathcal{J}_z(\theta, \phi; c^*)$ upper-bounds the original conditional multimodal bottleneck.

By Lemma 3.3, the KL term controls the conditional compression term, while the predictive log-likelihood term provides a variational lower bound on the conditional relevance term up to the additive constant $H(Y \mid c^*)$. The full proof is deferred to Appendix E. The posterior and prior are parameterized as

$$\begin{aligned} q_\theta(z \mid M, c^*) &= \mathcal{N}\big(\mu_\theta(M, c^*),\, \Sigma_\theta(M, c^*)\big), \\ r_\phi(z \mid c^*) &= \mathcal{N}\big(\mu_\phi(c^*),\, \Sigma_\phi(c^*)\big) \end{aligned} \tag{12}$$

where Σ_θ and Σ_ϕ are diagonal, $\sigma_\theta(M, c^*)$ is the elementwise standard-deviation vector, the posterior conditions multimodal rollout features on the fixed text card, and the prior represents the default multimodal expectation induced by c^* alone. During training, we use the standard reparameterization

$$\begin{aligned} z &= \mu_\theta(M, c^*) + \sigma_\theta(M, c^*) \odot \epsilon, \epsilon \sim \mathcal{N}(0, I), \\ \Sigma_\theta(M, c^*) &= \operatorname{diag}\big(\sigma_\theta(M, c^*) \odot \sigma_\theta(M, c^*)\big) \end{aligned} \tag{13}$$

Finally, the realized latent z is fused with the text card through the control map already introduced in Equation (5). Concretely, the projected latent $g_\omega(z)$ is prepended together with the fixed card c^* and the rollout bundle $\mathcal{B}$ to the frozen task model π_{tsk}, so that the prediction term in Equation (10) measures how much additional task-relevant information remains in z once the symbolic interface c^* has already been fixed.

3.4 Overall CMIB Objective

Combining the text-stage surrogate with the conditional multimodal surrogate yields the overall tractable realization of CMIB. Since the textual skill card is produced by discrete prompting rather than direct gradient-based optimization, we first the best textual skill card c^* and then optimize the continuous multimodal stage conditioned on the selected card. The resulting overall objective is

$$\widetilde{\mathcal{L}}_{\text{CMIB}}(\theta, \phi; c^*) = \mathcal{J}_c(c^*; \mathcal{B}) + \mathcal{J}_z(\theta, \phi; c^*). \tag{14}$$

Equation (14) is the trackable realization of the proposed information bottleneck Equation (4) that makes explicit how each information term in CMIB is realized in practice: the text-stage bottleneck is tracked by card length and task utility, while

the conditional multimodal bottleneck is realized by the variational KL term and the predictive log-likelihood term. All trainable components are confined to the posterior encoder q_θ, the prior r_ϕ, and the projection map g_ω, while the frozen task model π_{tsk} is never updated.

The final multimodal skill is a concrete realization of Equation (2): after the text stage selects c^* from Equation (8), we instantiate the multimodal skill as

$$\begin{aligned} S^* &= (c^*, z^*), \\ c^* &\sim \pi_{\text{sc}}(\cdot \mid \Pi_c(X, L_c)), \\ z^* &\sim q_\theta(\cdot \mid M, c^*), \end{aligned} \tag{15}$$

which is the realized form of $p_\psi(S \mid X, M)$ under the two-stage CMIB construction in Section 3.1. In this way, the text card c^* supplies an interpretable and retrievable procedural interface, while the latent z^* injects complementary multimodal evidence that cannot be faithfully compressed into text alone. The task model thus consumes the learned skill through both an explicit symbolic prompt and a fused soft control state.

4 Experiments

We evaluate whether CMIB achieves the information-theoretic goals of sufficiency, minimality, and complementarity while improving task performance, action consistency, and inference efficiency in multimodal agentic environments. This investigation is guided by the following research questions:

RQ1: CMIB Effectiveness. To what extent does CMIB improve task success rate and trajectory consistency of multimodal agents compared to state-of-the-art baselines?

RQ2: Action Consistency. How does CMIB affect action-level consistency across repeated trials compared with vanilla inference and self-consistency?

RQ3: Ablation Study. Does the conditional latent vector successfully capture residual perceptual information, and does it exhibit the theoretical collapse property when text is sufficient?

RQ4: Inference Efficiency and Computational Cost. Does CMIB mitigate the "prohibitively expensive" nature of sequential action generation compared to self-consistency sampling?

Dataset. We evaluate CMIB within the domain of Anthropic Agent Skill on `Multimodal-Mind2Web` (Deng et al., 2023), `ScreenSpot` (Cheng et al., 2024), and `AndroidControl` (Rawles et al., 2023), covering real-world web tasks, fine-grained GUI grounding, and mobile-device control.

Metrics. We report standard metrics across the three datasets, including element accuracy (Ele. Acc), operation F1 (Op. F1), step success rate (Step SR), and task success rate (SR), and assess action and trajectory stability with Step Consistency (StepCons) and EarlyDiv.

Model. We evaluate CMIB with `Qwen2.5-VL-7B-Instruct` (Team, 2025) and `InternVL3.5-4B`. CMIB augments the backbone with a skill library, using a `Qformer` (Zhang et al., 2024b) and `MLP` to encode multimodal trajectories into a latent vector decoded into soft prompts. We compare against *Vanilla Agent*, *Text-Only Skill Card*, and *Self-Consistency* (Wang et al., 2022) with up to $K{=}5$ multi-sample decoding, using identical splits, candidate sets, and prompting.

4.1 (RQ1) CMIB Effectiveness

Table 1 compares CMIB with in-context learning (ICL), supervised fine-tuning (SFT), and data-synthesis-plus-SFT baselines. Without task-specific tuning on LLMs, CMIB consistently improves over the Qwen2.5-VL-7B-instruction baseline across splits, showing that the conditional multimodal information bottleneck extracts complementary visual-textual cues into the skill library.

Among methods without task-specific backbone fine-tuning, CMIB outperforms ICL baselines such as SeeAct and GPT-4; on InternVL3.5-4B, Skill-CMIB improves average Step SR from 38.5 to 41.5, while stronger trained baselines such as AgentTrek-7B and Explorer-7B retain higher absolute Step SR.

Table 2 shows that Qwen-based Skill-CMIB is superior or competitive on `ScreenSpot`, especially Mobile Icon/Widget and Web Text. InternVL3.5-4B remains weak, likely because it is not tuned for GUI grounding; Skill-CMIB elicits capabilities already present in the backbone rather than replacing task-specific tuning.

Table 3 shows modest AndroidControl gains, suggesting that the Skill-CMIB skill library also benefits mobile-control scenarios.

4.2 (RQ2) Action Consistency

Task success alone does not capture the stability of intermediate decisions, so we run each setting for $N = 3$ independent repeats on the same evaluation

Method	Cross-Task			Cross-Website			Cross-Domain			Avg. Step SR
	Ele. Acc	Op. F1	Step SR	Ele. Acc	Op. F1	Step SR	Ele. Acc	Op. F1	Step SR	
In-Context Learning										
GPT-3.5 (Pahuja et al., 2025)	19.4	59.2	16.8	14.9	56.5	14.1	25.2	57.9	24.1	18.3
GPT-4 (Pahuja et al., 2025)	40.8	63.1	32.3	30.2	61.0	27.0	35.4	61.9	29.7	29.7
SeeAct (Zheng et al., 2024)	46.4	73.4	40.2	38.0	67.8	32.4	42.4	69.3	36.8	36.5
Supervised Fine-Tuning										
Generation (Deng et al., 2023)	20.2	52.0	17.5	13.9	44.7	11.0	14.2	44.7	11.9	13.5
Explorer-4B (Pahuja et al., 2025)	48.1	88.0	44.8	49.1	87.2	45.0	46.9	87.7	44.6	44.8
Explorer-7B (Pahuja et al., 2025)	51.8	88.0	48.3	56.3	89.7	52.0	50.9	88.9	48.1	49.5
Data Synthesis+Supervised Fine-Tuning										
SeeClick-9.6B (Cheng et al., 2024)	26.3	86.2	23.7	21.9	82.9	18.8	22.1	84.1	20.2	20.9
MiniCPM-GUI-3.1B (Chen et al., 2025)	23.8	86.8	20.8	20.3	81.7	17.3	17.9	74.5	14.6	17.6
Scribe-Agent-L.-32B (Shen et al., 2024)	38.0	52.9	35.6	34.1	52.7	32.5	39.4	54.7	37.3	35.1
AgentTrek-7B (Xu et al., 2025)	60.8	88.9	55.7	57.6	88.1	51.4	56.0	87.5	52.6	53.2
Explorer-4B (Pahuja et al., 2025)	53.4	88.1	50.7	55.6	89.5	51.4	49.8	88.8	47.2	49.8
Explorer-7B (Pahuja et al., 2025)	56.5	90.3	53.2	60.5	90.7	56.7	55.7	90.4	53.0	54.3
Skill Based										
InternVL3.5-4B	55.5	80.3	40.2	51.5	78.5	36.3	53.2	80.0	39.0	38.5
InternVL3.5-4B + CMIB	**60.5**	**82.5**	**45.6**	**55.6**	**79.8**	**37.2**	55.5	**82.5**	41.7	**41.5**
Qwen2.5-VL-7B-Inst.	51.9	63.5	33.4	44.7	57.5	25.5	46.8	66.3	32.4	30.4
Qwen2.5-VL-7B-Inst. + CMIB	57.0	59.8	39.0	55.4	60.8	34.1	**57.4**	69.5	**42.9**	38.7

Table 1: Results across different settings on Multimodal-Mind2Web (Deng et al., 2023). Shaded rows denote finetuned baselines; bold indicates the best result among non-finetuned methods.

LVLMs (Size)	Mobile		Desktop		Web		Avg.
	Text	Icon/Widget	Text	Icon/Widget	Text	Icon/Widget	
MiniGPT-v2-7B	8.4	6.6	6.2	2.9	6.5	3.4	5.7
Qwen-VL-9.6B	9.5	4.8	5.7	5.0	3.5	2.4	5.2
GPT-4V	22.6	24.5	20.2	11.8	9.2	8.8	16.2
Fuyu-8B	41.0	1.3	33.0	3.6	33.9	4.4	19.5
CogAgent-18B	67.0	24.0	74.2	20.0	70.4	28.6	47.4
SeeClick-9.6B	78.0	52.0	72.2	30.0	55.7	32.5	53.4
InternVL3.5-4B	1.5	0.4	4.1	5.0	2.2	1.9	2.3
InternVL3.5-4B + CMIB	2.2	0.9	3.1	5.7	1.7	1.9	2.4
Qwen2.5-VL-7B-Inst.	60.8	53.3	**57.7**	42.1	51.7	46.6	53.0
Qwen2.5-VL-7B-Inst. + CMIB	**64.8**	**65.1**	54.1	**46.4**	**60.0**	**49.5**	**57.9**

Table 2: Action Success Rate (%) compared with baselines on ScreenSpot (Cheng et al., 2024).

Model	AC-L			AC-H		
	Type	GR	SR	Type	GR	SR
GPT-4o	74.33	38.67	28.39	63.06	30.90	21.17
Qwen2.5-3B	62.03	**74.07**	59.32	47.81	46.51	38.90
Qwen2.5-VL-7B-Inst.	91.54	69.40	67.53	75.15	47.84	43.47
Qwen2.5-VL-7B-Inst.+CMIB	**91.66**	70.90	**68.19**	**75.52**	**48.97**	**44.61**
InternVL-4B	86.84	0.69	20.26	70.84	0.82	13.34
InternVL-4B+CMIB	87.15	0.69	21.44	71.82	0.75	13.91

Table 3: AndroidControl results. AC-L/AC-H denote AndroidControl-Low/High. Type denotes action-type accuracy, GR denotes grounding rate, and SR denotes step success rate. (Luo et al., 2025)

split and report **Step Consistency** (StepCons) with EarlyDiv@3.

$$\text{StepCons} = \frac{1}{\binom{N}{2}} \sum_{i<j} \frac{1}{|\mathcal{S}_{ij}|} \sum_{t \in \mathcal{S}_{ij}} \mathbf{1}\left[\tilde{a}_t^{(i)} = \tilde{a}_t^{(j)}\right].$$

$$d_e^{(i,j)} = \min\{t : \tilde{a}_{e,t}^{(i)} \neq \tilde{a}_{e,t}^{(j)}\},$$

$$\text{EarlyDiv@3} = \frac{1}{\binom{N}{2}} \sum_{i<j} \frac{1}{|\mathcal{E}_{ij}|} \sum_{e \in \mathcal{E}_{ij}} \mathbf{1}\left[d_e^{(i,j)} \leq 3\right]. \tag{16}$$

Here $\mathcal{E}_{ij}$ denotes common episodes for repeats i, j, and $d_e^{(i,j)}$ is the first normalized-action mismatch step; lower EarlyDiv@3 indicates later divergence. Table 4 shows that CMIB achieves higher StepCons and lower EarlyDiv@3 than the best self-consistency baseline, indicating more stable step- and trajectory-level action selection.

As k increases from 1 to 5, self-consistency raises average Step SR from 33.77% to 35.98% and StepCons from 0.0866 to 0.1789, consistent with prior observations that larger k reduces variance in multi-step reasoning (Mehta, 2026). However, even when self-consistency with $k = 5$ achieves comparable Step SR on some splits, its StepCons remains far below CMIB (0.1789 vs. 0.4144), showing that CMIB maintains competitive task performance while producing more consistent intermediate actions.

4.3 (RQ3) Ablation Study

To validate the theoretical underpinnings of CMIB, we ablate the multimodal stage by comparing full

Setting	Metric	CMIB (Full)	SC (k=1)	SC (k=2)	SC (k=3)	SC (k=4)	SC (k=5)
Cross-Task	Ele. Acc	51.35	52.98	52.98	54.43	56.23	**56.78**
	Op. F1	61.08	67.13	67.13	67.94	**68.27**	68.12
	Step SR	39.16	38.39	38.39	40.51	**41.49**	41.48
	StepCons	**43.67**	11.33	11.33	11.00	12.00	17.33
	EarlyDiv@3 ↓	**89.74**	100.00	100.00	100.00	100.00	100.00
Cross-Website	Ele. Acc	**44.68**	38.77	38.77	39.80	40.63	41.78
	Op. F1	62.84	61.78	61.78	61.10	62.73	**64.94**
	Step SR	**28.14**	24.48	24.48	24.87	25.31	26.54
	StepCons	**42.33**	6.33	6.33	10.00	12.00	16.00
	EarlyDiv@3 ↓	**92.16**	100.00	100.00	100.00	100.00	100.00
Cross-Domain	Ele. Acc	**69.88**	57.02	57.02	58.02	59.23	59.88
	Op. F1	**75.13**	67.71	67.71	67.14	68.70	68.80
	Step SR	**49.42**	38.45	38.45	38.25	40.07	39.91
	StepCons	**38.33**	8.33	8.33	12.00	16.33	20.33
	EarlyDiv@3 ↓	**92.16**	100.00	100.00	100.00	100.00	100.00
Average	Avg. Step SR	**38.91**	33.77	33.77	34.54	35.62	35.98
Average	Avg. StepCons	**41.44**	8.66	8.66	11.00	13.44	17.89
Average	Avg. EarlyDiv@3 ↓	**91.35**	100.00	100.00	100.00	100.00	100.00

Table 4: Model Performance on Multimodal-Mind2Web. Results are percentages; ↓ lower is better. Bold: best in each row.

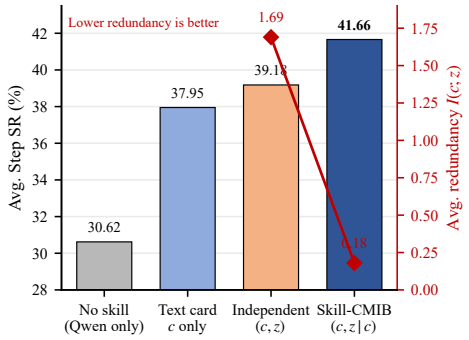

Figure 3: Component ablation on Mind2Web. Skill-CMIB improves Step SR while reducing redundancy between c and z.

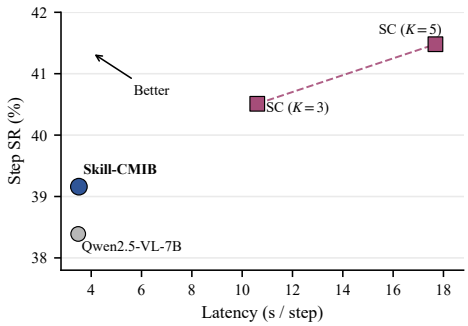

Figure 4: Inference efficiency trade-off. Skill-CMIB achieves competitive Step SR with substantially lower latency than self-consistency.

CMIB with text-card-only, independent c and z, and Qwen2.5-VL-7B variants, measuring Step Success Rate and Information Redundancy $I(c; z)$ through the average KL divergence between $q_\theta(z \mid M, c^*)$ and $r_\phi(z \mid c^*)$. Figure 3 shows that CMIB achieves the highest Step SR and much lower redundancy, indicating a more minimal representation.

Component results show the same pattern: removing the redundancy constraint raises $I(c; z)$ and drops Step SR to 39.18, omitting z (*Text Card c only*) reduces Step SR to 37.95, and removing the CMIB skill library (*No skill*) yields the largest drop to 30.62. Thus, z captures perceptual skill information beyond c, and each CMIB component contributes to performance.

4.4 (RQ4) Inference Efficiency and Computational Cost

Figure 4 shows the performance–cost trade-off: Self-Consistency improves performance with $K\times$ inference overhead, whereas CMIB achieves better or comparable results with minimal additional inference cost through a lightweight Q-former, MLP projector, and reusable multimodal skill library.

5 Conclusion

In this paper, we introduced Skill-CMIB, a principled framework for multimodal agent skill construction that leverages the Information Bottleneck principle to enhance agent action consistency across trials. By employing a sequential decomposition, Skill-CMIB partitions skills into interpretable text-stage bottlenecks and conditional multimodal bottlenecks, effectively distilling symbolic skill cards while capturing essential residual perceptual evidence. Theoretical analysis and empirical evaluations on benchmarks such as Multimodal-Mind2Web, ScreenSpot, and Android-Control demonstrate that our approach significantly improves task success rates and consistency without the prohibitive overhead of self-consistency. By ensuring sufficiency, minimality, and cross-modal complementarity, Skill-CMIB provides a robust foundation for building reliable multimodal agent Skill Library.

Limitations

This work focuses on multimodal agent skills for web, GUI, and mobile-control environments, which cover a broad set of visually grounded interaction tasks but do not exhaust all possible real-world deployment settings. Skill-CMIB also assumes access to trial-and-error trajectories for skill construction and therefore introduces an offline skill-building cost before deployment; this cost is amortized when the resulting skill library is reused across related tasks. Finally, because Skill-CMIB augments rather than replaces the underlying task model, its gains depend on the backbone model's basic perception, grounding, and action-generation capabilities. Extending the framework to broader domains, larger skill libraries, and stronger state-aware skill retrieval remains a promising direction for future work.

LLM usage disclosure: AI writing tools were used to assist in drafting and verifying the theoretical proofs in this paper. AI tools were used to assist creating the illustrative figures.

References

Pranjal Aggarwal, Aman Madaan, Yiming Yang, and Mausam. 2023. Let's sample step by step: Adaptive-consistency for efficient reasoning and coding with LLMs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12375–12396, Singapore. Association for Computational Linguistics.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *Preprint*, arXiv:2307.15818.

Wentong Chen, Junbo Cui, Jinyi Hu, Yujia Qin, Junjie Fang, Yue Zhao, Chongyi Wang, Jun Liu, Guirong Chen, Yupeng Huo, Yuan Yao, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2025. Guicourse: From general vision language models to versatile gui agents. *Preprint*, arXiv:2406.11317.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. 2024. Seeclick: Harnessing GUI grounding for advanced visual GUI agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 9313–9332. Association for Computational Linguistics.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samual Stevens, Boshi Wang, Huan Sun, and Yu Su. 2023. Mind2web: Towards a generalist agent for the web. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. 2023. Palm-e: An embodied multimodal language model. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, Proceedings of Machine Learning Research, pages 8469–8488. PMLR.

Runnan Fang, Yuan Liang, Xiaobin Wang, Jialong Wu, Shuofei Qiao, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. 2025. Memp: Exploring agent procedural memory. *arXiv preprint arXiv:2508.06433*.

Saman Forouzandeh, Wei Peng, Parham Moradi, Xinghuo Yu, and Mahdi Jalili. 2025. Learning hierarchical procedural memory for llm agents through bayesian selection and contrastive refinement. *arXiv preprint arXiv:2512.18950*.

Anirudh Goyal, Riashat Islam, Daniel Strouse, Zafarali Ahmed, Matthew Botvinick, Hugo Larochelle, Yoshua Bengio, and Sergey Levine. 2019a. Infobot: Transfer and exploration via the information bottleneck. *arXiv preprint arXiv:1901.10902*.

Anirudh Goyal, Riashat Islam, Daniel Strouse, Zafarali Ahmed, Hugo Larochelle, Matthew M. Botvinick, Yoshua Bengio, and Sergey Levine. 2019b. Infobot: Transfer and exploration via the information bottleneck. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Chengkai Huang, Hongtao Huang, Tong Yu, Kaige Xie, Junda Wu, Shuai Zhang, Julian Mcauley, Dietmar Jannach, and Lina Yao. 2025a. A survey of foundation model-powered recommender systems: From feature-based, generative to agentic paradigms. *arXiv preprint arXiv:2504.16420*.

Chengkai Huang, Junda Wu, Yu Xia, Zixu Yu, Ruhan Wang, Tong Yu, Ruiyi Zhang, Ryan A Rossi, Branislav Kveton, Dongruo Zhou, et al. 2025b. Towards agentic recommender systems in the era of multimodal large language models. *arXiv preprint arXiv:2503.16734*.

Chengkai Huang, Junda Wu, Zhouhang Xie, Yu Xia, Rui Wang, Tong Yu, Subrata Mitra, Julian McAuley, and Lina Yao. 2025c. Pluralistic off-policy evaluation and alignment. *arXiv preprint arXiv:2509.19333*.

Zihan Huang, Xintong Li, Rohan Surana, Tong Yu, Rui Wang, Julian McAuley, Jingbo Shang, and Junda Wu. 2026. Amps: Adaptive modality preference steering via functional entropy. *arXiv preprint arXiv:2602.12533*.

Zihan Huang, Junda Wu, Rohan Surana, Raghav Jain, Tong Yu, Raghavendra Addanki, David Arbour, Sungchul Kim, and Julian McAuley. 2025d. Traceable and explainable multimodal large language models: An information-theoretic view. In *Second Conference on Language Modeling*.

Guanyu Jiang, Zhaochen Su, Xiaoye Qu, et al. 2026a. Xskill: Continual learning from experience and skills in multimodal agents. *arXiv preprint arXiv:2603.12056*.

Yanna Jiang, Delong Li, Haiyu Deng, Baihe Ma, Xu Wang, Qin Wang, and Guangsheng Yu. 2026b. Sok: Agentic skills–beyond tool use in llm agents. *arXiv preprint arXiv:2602.20867*.

Branislav Kveton, Xintong Li, Julian McAuley, Ryan Rossi, Jingbo Shang, Junda Wu, and Tong Yu. 2025. Active learning for direct preference optimization. *arXiv preprint arXiv:2503.01076*.

Hao Li, Chunjiang Mu, Jianhao Chen, Siyue Ren, Zhiyao Cui, Yiqun Zhang, Lei Bai, and Shuyue Hu. 2026a. Organizing, orchestrating, and benchmarking agent skills at ecosystem scale. *arXiv preprint arXiv:2603.02176*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. 2023. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, Proceedings of Machine Learning Research, pages 19730–19742. PMLR.

Xiangyi Li, Wenbo Chen, Yimin Liu, Shenghan Zheng, Xiaokun Chen, Yifeng He, Yubo Li, Bingran You, Haotian Shen, Jiankai Sun, et al. 2026b. Skillsbench: Benchmarking how well agent skills work across diverse tasks. *arXiv preprint arXiv:2602.12670*.

Xintong Li, Junda Wu, Tong Yu, Rui Wang, Yu Wang, Xiang Chen, Jiuxiang Gu, Lina Yao, Julian McAuley, and Jingbo Shang. 2025. Commit: Coordinated multimodal instruction tuning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 11533–11547.

George Ling, Shanshan Zhong, and Richard Huang. 2026. Agent skills: A data-driven analysis of claude skills for extending large language model functionality. *arXiv preprint arXiv:2602.08004*.

Xiaoyu Liu, Paiheng Xu, Junda Wu, Jiaxin Yuan, Yifan Yang, Yuhang Zhou, Fuxiao Liu, Tianrui Guan, Haoliang Wang, Tong Yu, et al. 2025. Large language models and causal inference in collaboration: A comprehensive survey. *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 7668–7684.

Run Luo, Lu Wang, Wanwei He, Longze Chen, Jiaming Li, and Xiaobo Xia. 2025. Gui-r1 : A generalist r1-style vision-language action model for gui agents. *Preprint*, arXiv:2504.10458.

Rabeeh Karimi Mahabadi, Yonatan Belinkov, and James Henderson. 2021. Variational information bottleneck for effective low-resource fine-tuning. *arXiv preprint arXiv:2106.05469*.

Aman Mehta. 2026. When agents disagree with themselves: Measuring behavioral consistency in llm-based agents. *Preprint*, arXiv:2602.11619.

Qirui Mi, Zhijian Ma, Mengyue Yang, Haoxuan Li, Yisen Wang, Haifeng Zhang, and Jun Wang. 2026. Procmem: Learning reusable procedural memory from experience via non-parametric ppo for llm agents. *arXiv preprint arXiv:2602.01869*.

Gagan Mundada, Zihan Huang, Rohan Surana, Sheldon Yu, Jennifer Yuntong Zhang, Xintong Li, Tong Yu, Lina Yao, Jingbo Shang, Julian McAuley, et al. 2026. Ws-grpo: Weakly-supervised group-relative policy optimization for rollout-efficient reasoning. *arXiv preprint arXiv:2602.17025*.

Dang Nguyen, Jian Chen, Yu Wang, Gang Wu, Namyong Park, Zhengmian Hu, Hanjia Lyu, Junda Wu, Ryan Aponte, Yu Xia, et al. 2025. Gui agents: A survey. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 22522–22538.

Thao Nguyen, Haotian Liu, Yuheng Li, Mu Cai, Utkarsh Ojha, and Yong Jae Lee. 2024. Yo'llava: Your personalized language and vision assistant. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Bo Ni, Yu Wang, Leyao Wang, Branislav Kveton, Franck Dernoncourt, Yu Xia, Hongjie Chen, Reuben Luera, Samyadeep Basu, Subhojyoti Mukherjee, et al. 2026. A survey on llm-based conversational user simulation. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4266–4301.

Vardaan Pahuja, Yadong Lu, Corby Rosset, Boyu Gou, Arindam Mitra, Spencer Whitehead, Yu Su, and Ahmed Hassan Awadallah. 2025. Explorer: Scaling

exploration-driven web trajectory synthesis for multimodal web agents. In *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, Findings of ACL, pages 6300–6323. Association for Computational Linguistics.

Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. 2019. On variational bounds of mutual information. In *International conference on machine learning*, pages 5171–5180. PMLR.

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. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, Proceedings of Machine Learning Research, pages 8748–8763. PMLR.

Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, and Timothy P. Lillicrap. 2023. Android in the wild: A large-scale dataset for android device control. *CoRR*, abs/2307.10088.

Ankur Samanta, Akshayaa Magesh, Runzhe Wu, Ayush Jain, Youliang Yu, Daniel Jiang, Boris Vidolov, Paul Sajda, Yonathan Efroni, and Kaveh Hassani. 2026. Self-improvement of language models by post-training on multi-agent debate. *Preprint*, arXiv:2509.15172.

Junhong Shen, Atishay Jain, Zedian Xiao, Ishan Amlekar, Mouad Hadji, Aaron Podolny, and Ameet Talwalkar. 2024. Scribeagent: Towards specialized web agents using production-scale workflow data. *Preprint*, arXiv:2411.15004.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Preprint*, arXiv:2303.11366.

Simon Stepputtis, Joseph Campbell, Mariano J. Phielipp, Stefan Lee, Chitta Baral, and Heni Ben Amor. 2020. Language-conditioned imitation learning for robot manipulation tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Rohan Surana, Junda Wu, Xintong Li, Sheldon Yu, Yiran Jenny Shen, Chuhan Wang, Tong Yu, Prithviraj Ammanabrolu, Jingbo Shang, and Julian McAuley. 2026. MASS-DPO: Multi-negative active sample selection for direct policy optimization.

Qwen Team. 2025. Qwen2.5-vl.

Naftali Tishby, Fernando C Pereira, and William Bialek. 2000. The information bottleneck method. *arXiv preprint physics/0004057*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding. *Preprint*, arXiv:1807.03748.

Chuhan Wang, Xintong Li, Jennifer Yuntong Zhang, Junda Wu, Chengkai Huang, Lina Yao, Julian McAuley, and Jingbo Shang. 2026. Scenealign: Aligning multimodal reasoning to scene graphs in complex visual scenes. *arXiv preprint arXiv:2601.05600*.

Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. 2024. Soft self-consistency improves language models agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 287–301.

Jiongxiao Wang, Qiaojing Yan, Yawei Wang, Yijun Tian, Soumya Smruti Mishra, Zhichao Xu, Megha Gandhi, Panpan Xu, and Lin Lee Cheong. 2025a. Reinforcement learning for self-improving agent with skill library. *arXiv preprint arXiv:2512.17102*.

Ruoyu Wang, Junda Wu, Yu Xia, Tong Yu, Ryan A Rossi, Julian McAuley, and Lina Yao. 2025b. Dice: Dynamic in-context example selection in llm agents via efficient knowledge transfer. *arXiv preprint arXiv:2507.23554*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Yu Wang, Xinshuang Liu, Xiusi Chen, Sean OBrien, Junda Wu, and Julian McAuley. 2025c. Self-updatable large language models by integrating context into model parameters. In *International Conference on Learning Representations*, volume 2025, pages 16961–16979.

Junda Wu, Cheng-Chun Chang, Tong Yu, Zhankui He, Jianing Wang, Yupeng Hou, and Julian McAuley. 2024a. Coral: Collaborative retrieval-augmented large language models improve long-tail recommendation. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pages 3391–3401.

Junda Wu, Xintong Li, Ruoyu Wang, Yu Xia, Yuxin Xiong, Jianing Wang, Tong Yu, Xiang Chen, Branislav Kveton, Lina Yao, et al. 2025a. Ocean: Offline chain-of-thought evaluation and alignment in large language models. In *International Conference on Learning Representations*, volume 2025, pages 100570–100589.

Junda Wu, Hanjia Lyu, Yu Xia, Zhehao Zhang, Joe Barrow, Ishita Kumar, Mehrnoosh Mirtaheri, Hongjie Chen, Ryan A Rossi, Franck Dernoncourt, et al. 2024b. Personalized multimodal large language models: A survey. *arXiv preprint arXiv:2412.02142*.

Junda Wu, Rohan Surana, Zhouhang Xie, Yiran Shen, Yu Xia, Tong Yu, Ryan A Rossi, Prithviraj Ammanabrolu, and Julian McAuley. 2025b. In-context ranking preference optimization. In *Second Conference on Language Modeling*.

Junda Wu, Rui Wang, Tong Yu, Ruiyi Zhang, Handong Zhao, Shuai Li, Ricardo Henao, and Ani Nenkova. 2022. Context-aware information-theoretic causal de-biasing for interactive sequence labeling. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3436–3448, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Junda Wu, Yu Xia, Tong Yu, Xiang Chen, Sai Sree Harsha, Akash V Maharaj, Ruiyi Zhang, Victor Bursztyn, Sungchul Kim, Ryan A. Rossi, Julian McAuley, Yunyao Li, and Ritwik Sinha. 2025c. Doc-react: Multi-page heterogeneous document question-answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 67–78, Vienna, Austria. Association for Computational Linguistics.

Junda Wu, Yu Xia, Tong Yu, Xiang Chen, Sai Sree Harsha, Akash V Maharaj, Ruiyi Zhang, Victor Bursztyn, Sungchul Kim, Ryan A Rossi, et al. 2025d. Doc-react: Multi-page heterogeneous document question-answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 67–78.

Junda Wu, Yuxin Xiong, Xintong Li, Yu Xia, Ruoyu Wang, Yu Wang, Tong Yu, Sungchul Kim, Ryan A Rossi, Lina Yao, et al. 2025e. Mitigating visual knowledge forgetting in mllm instruction-tuning via modality-decoupled gradient descent. *arXiv preprint arXiv:2502.11740*, 8.

Junda Wu, Yuxin Xiong, Xintong Li, Sheldon Yu, Zhengmian Hu, Tong Yu, Rui Wang, Xiang Chen, Jingbo Shang, and Julian McAuley. 2026. Ctrls: Chain-of-thought reasoning via latent state-transition. In *The 29th International Conference on Artificial Intelligence and Statistics*.

Junda Wu, Tong Yu, Xiang Chen, Haoliang Wang, Ryan Rossi, Sungchul Kim, Anup Rao, and Julian McAuley. 2024c. Decot: Debiasing chain-of-thought for knowledge-intensive tasks in large language models via causal intervention. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14073–14087.

Junda Wu, Tong Yu, and Shuai Li. 2021. Deconfounded and explainable interactive vision-language retrieval of complex scenes. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2103–2111.

Junda Wu, Tong Yu, Rui Wang, Zhao Song, Ruiyi Zhang, Handong Zhao, Chaochao Lu, Shuai Li, and Ricardo Henao. 2023. Infoprompt: Information-theoretic soft prompt tuning for natural language understanding. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Junda Wu, Zhehao Zhang, Yu Xia, Xintong Li, Zhaoyang Xia, Aaron Chang, Tong Yu, Sungchul Kim, Ryan A Rossi, Ruiyi Zhang, et al. 2024d. Visual prompting in multimodal large language models: A survey. *arXiv preprint arXiv:2409.15310*.

Yaxiong Wu and Yongyue Zhang. 2026. Agent skills from the perspective of procedural memory: A survey. *Authorea Preprints*.

Yu Xia, Yiran Jenny Shen, Junda Wu, Tong Yu, Sungchul Kim, Ryan A Rossi, Lina Yao, and Julian McAuley. 2025. Sand: Boosting llm agents with self-taught action deliberation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 3062–3077.

Zhouhang Xie, Junda Wu, Yiran Shen, Raghav Jain, Yu Xia, Xintong Li, Aaron Chang, Ryan A. Rossi, Tong Yu, Sachin Kumar, Bodhisattwa Prasad Majumder, Jingbo Shang, Prithviraj Ammanabrolu, and Julian McAuley. 2025. A survey on personalized and pluralistic preference alignment in large language models. In *Second Conference on Language Modeling*.

Renjun Xu and Yang Yan. 2026a. Agent skills for large language models: Architecture, acquisition, security, and the path forward. *Preprint*, arXiv:2602.12430.

Renjun Xu and Yang Yan. 2026b. Agent skills for large language models: Architecture, acquisition, security, and the path forward. *arXiv preprint arXiv:2602.12430*.

Yangjie Xu, Lujun Li, Lama Sleem, Niccolo Gentile, Yewei Song, Yiqun Wang, Siming Ji, Wenbo Wu, and Radu State. 2026. Agent skill framework: Perspectives on the potential of small language models in industrial environments. *arXiv preprint arXiv:2602.16653*.

Yiheng Xu, Dunjie Lu, Zhennan Shen, Junli Wang, Zekun Wang, Yuchen Mao, Caiming Xiong, and Tao Yu. 2025. Agenttrek: Agent trajectory synthesis via guiding replay with web tutorials. *Preprint*, arXiv:2412.09605.

An Yan, Zhengyuan Yang, Junda Wu, Wanrong Zhu, Jianwei Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Julian McAuley, Jianfeng Gao, et al. List items one by one: A new data source and learning paradigm for multimodal llms. In *First Conference on Language Modeling*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. *Preprint*, arXiv:2210.03629.

Sheldon Yu, Yuxin Xiong, Junda Wu, Xintong Li, Tong Yu, Xiang Chen, Ritwik Sinha, Jingbo Shang, and Julian McAuley. 2025. Explainable chain-of-thought reasoning: An empirical analysis on state-aware reasoning dynamics. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 16660–16667, Suzhou, China. Association for Computational Linguistics.

Barry Zhang, Keith Lazuka, and Mahesh Murag. 2025. Equipping agents for the real world with agent skills. *Anthropic Engineering Blog*.

Danyang Zhang, Zhennan Shen, Rui Xie, Situo Zhang, Tianbao Xie, Zihan Zhao, Siyuan Chen, Lu Chen, Hongshen Xu, Ruisheng Cao, and Kai Yu. 2024a. Mobile-env: Building qualified evaluation benchmarks for llm-gui interaction. *Preprint*, arXiv:2305.08144.

Haozhen Zhang, Quanyu Long, Jianzhu Bao, Tao Feng, Weizhi Zhang, Haodong Yue, and Wenya Wang. 2026. Memskill: Learning and evolving memory skills for self-evolving agents. *arXiv preprint arXiv:2602.02474*.

Qiming Zhang, Jing Zhang, Yufei Xu, and Dacheng Tao. 2024b. Vision transformer with quadrangle attention. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(5):3608–3624.

Zhehao Zhang, Ryan A Rossi, Branislav Kveton, Yijia Shao, Diyi Yang, Hamed Zamani, Franck Dernoncourt, Joe Barrow, Tong Yu, Sungchul Kim, et al. 2024c. Personalization of large language models: A survey. *arXiv preprint arXiv:2411.00027*.

Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. 2024. Gpt-4v(ision) is a generalist web agent, if grounded. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, Proceedings of Machine Learning Research, pages 61349–61385. PMLR / OpenReview.net.

YanZhao Zheng, ZhenTao Zhang, Chao Ma, YuanQiang Yu, JiHuan Zhu, Baohua Dong, and Hangcheng Zhu. 2026. Skillrouter: Retrieve-and-rerank skill selection for llm agents at scale. *Preprint*, arXiv:2603.22455.

Huichi Zhou, Siyuan Guo, Anjie Liu, Zhongwei Yu, Ziqin Gong, Bowen Zhao, Zhixun Chen, Menglong Zhang, Yihang Chen, Jinsong Li, et al. 2026. Memento-skills: Let agents design agents. *arXiv preprint arXiv:2603.18743*.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. 2024. Webarena: A realistic web environment for building autonomous agents. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

A Preliminaries

A.1 Agent Skills

Let π_{tsk} denote a frozen task LLM that interacts with an environment over multiple decision steps. An *agent skill* is a reusable procedure $s_i \in \mathcal{S}$, represented by a skill card $c_i \in \mathcal{C}$, that specifies how to solve a class of tasks rather than a single instance (Li et al., 2026b; Xu et al., 2026; Xu and Yan, 2026b). A *skill library* is a finite collection $\mathcal{L} = \{s_i\}_{i=1}^{N}$ of such skills, as commonly maintained in agentic systems (Ling et al., 2026; Li et al., 2026a); in many settings, the library is updated across episodes or task chains rather than within a trajectory (Fang et al., 2025; Forouzandeh et al., 2025). At step t, the task LLM conditions on the current state $h_t = (x_t, a_{<t}, o_{<t})$, together with an active skill subset $\mathcal{L}_t \subseteq \mathcal{L}$ selected from the larger library (Zheng et al., 2026; Li et al., 2026a; Zhou et al., 2026), and outputs $a_t \sim \pi_{\text{tsk}}(\cdot \mid h_t, \mathcal{L}_t)$. The environment then returns an observation o_t, yielding the next state h_{t+1}. Iterating this process produces a rollout

$$\tau^{(k)} = \big(x_t^{(k)},\, a_t^{(k)},\, o_t^{(k)}\big)_{t=1}^{T_k}, \tag{17}$$

and K related trial-and-error rollouts form the bundle $\mathcal{B} = \{\tau^{(k)}\}_{k=1}^{K}$ (Wang et al., 2025a; Mi et al., 2026; Jiang et al., 2026a).

A.2 Information Bottleneck

The Information Bottleneck (IB) principle (Tishby et al., 2000) provides a foundational framework for learning compressed representations. Given a source random variable W and a target Y, the IB seeks a representation Z by solving

$$\min_{p(z|w)} \; I(W;\, Z) - \beta\, I(Z;\, Y), \tag{18}$$

where $I(\cdot;\cdot)$ denotes mutual information and $\beta \geq 0$ controls the trade-off between compression $I(W;Z)$ and relevance $I(Z;Y)$. Setting $W = (X, M)$ and $Z = S$, with X and M the aggregate textual and multimodal content associated with rollout bundles (formalized below), recovers a naive multimodal skill bottleneck, but this ignores the heterogeneous nature of discrete text and continuous multimodal features, offers no retrieval interface, and provides no mechanism for ensuring cross-modal complementarity.

B Skill-card generation prompt and examples

We generate the text skill card c from a consolidated bundle summary using the following prompt. The prompt is applied after per-trajectory summaries are merged across related trials, and the final output is constrained to four reusable fields:

Exact skill-card generation prompt. Based on the consolidated skill summary below, produce a structured skill card with exactly four fields. Be concise (each field ≤ 2 sentences).

Summary: {summary}

Requirements:

- Keep each field reusable across similar tasks and websites.
- Include high-level context and decision cues, not just low-level click/type instructions.
- Avoid overfitting details (exact entity names, brittle selectors, unique IDs, one-off URLs).
- Preserve enough situational context to improve agent reasoning under partial observability.

Response format:

```
[Trigger] <when to apply this skill>
[Action] <what the agent should do>
[Failure] <common failure modes to
watch for>
[Verify] <how to confirm the skill
worked>
```

The following representative examples are selected from the generated skill-card cache on the Multimodal-Mind2Web training split. We choose examples that cover common web-agent patterns, including multi-field booking, location-based filtering, structured form filling, progressive search refinement, and checkout-like workflows.

Multi-field vehicle booking

[Trigger] Apply this skill when initiating a search by accurately inputting the location to narrow down relevant options.

[Action] Utilize dropdown menus and search functionalities to select specific criteria like location, time, and vehicle type, employing a methodical approach with sequential clicking and typing.

[Failure] Watch for a lack of specificity in search criteria and excessive clicking without progress, which can lead to incomplete task execution.

[Verify] Confirm success by reviewing selections to ensure all criteria, such as car type and insurance options, are correctly applied and align with initial instructions.

Location-based inventory filtering

[Trigger] Apply this skill when searching for inventory based on geographic relevance.

[Action] Begin by inputting a specific location (ZIP code) and selecting a distance radius to filter results effectively.

[Failure] Watch for failures in filtering specific car models and colors, which can lead to overly broad searches.

[Verify] Confirm success by ensuring all relevant filters (location, distance, model, and color) are applied, yielding a focused set of results.

Sequential financial form filling

[Trigger] Apply this skill when specific financial details need to be inputted and options selected from dropdown menus to tailor calculations.

[Action] Break down tasks into smaller steps for sequential data entry and use interface elements like dropdowns and hover/click actions to navigate the platform effectively.

[Failure] Watch for incorrect data entry or incomplete task execution, such as failing to proceed to subsequent steps after calculations.

[Verify] Confirm the skill worked by ensuring all required fields are filled, verifying each input, and using structured approaches and interface guidance for successful task execution.

Progressive search, sorting, and filter refinement

[Trigger] Apply this skill when entering a specific location or ZIP code early in the search process to ensure geographically relevant results.

[Action] Use sorting features like "Price - Lowest" and apply filters sequentially (make, model, year, features) to streamline the search and meet task requirements.

[Failure] Watch for redundant or missed application of filters, which can cause inefficiencies or irrelevant results, and ensure the final selection is verified for accuracy.

[Verify] Regularly check and adjust filters to meet all task criteria, and verify the final selection details to confirm it meets specifications and is the best option.

Gift card purchase workflow

[Trigger] Apply this skill when navigating to the gift card section.

[Action] The agent should execute tasks sequentially: select card type, enter information, and proceed to checkout.

[Failure] Watch for the absence of a confirmation step, which can lead to errors like incorrect email addresses or missing cart additions.

[Verify] Confirm the skill worked by implementing a confirmation step to review all details and ensure items are correctly added to the cart.

C Case study on Mind2Web action grounding

Although Skill-CMIB improves average Step SR and substantially improves StepCons over self-consistency (SC), aggregate metrics do not explain

how the two methods differ at the level of individual web actions. We therefore inspect aligned Mind2Web predictions from the same split, run, task id, and step. Below we show two representative CMIB success cases and one underperformance case where SC is correct but CMIB is not.

CMIB success: workflow-state grounding for location editing

Task. The user asks: "see Nissan and Honda cars for sale near Kentwood, MI 49512." At this step, the page is not yet ready to accept the new location value; the agent first needs to open the location editing interface.

GT	`CLICK`
CMIB	`CLICK` on `Change Location` button **(correct)**
SC	`SELECT` with value `Kentwood, MI 49512` **(wrong)**

Interpretation. SC captures the semantic target location, but applies it too early. CMIB instead grounds the current workflow state: before entering or selecting `Kentwood, MI 49512`, the agent must click the `Change Location` control. This illustrates a key benefit of skill memory: it can stabilize multi-step UI procedures, not only recognize task-relevant text.

CMIB success: choosing the executable UI operation, not just the target text

Task. The user asks: "Print a recipe containing shrimp." At this step, a search result for a shrimp recipe is visible. The correct next action is to open the recipe page by clicking the result link.

GT	`CLICK`
CMIB	`CLICK` on recipe result `Mediterranean shrimp pasta` **(correct)**
SC	`SELECT` with value `Mediterranean shrimp pasta` **(wrong)**

Interpretation. Both methods identify the right semantic object, but only CMIB predicts the executable browser action. SC's output is semantically close yet operationally invalid for this UI state. This explains why Step SR can differ even when both methods appear to understand the target: web action grounding requires selecting the correct affordance and operation type.

Underperformance: CMIB over-regularizes toward an abstract filter policy

Task. The user asks: "Find the cheapest snow colored Google Pixel 7 with 128GB storage that can be bought on Google." The task requires filtering and sorting, but at this particular step the page expects a click on the current control.

GT	`CLICK`
CMIB	`SELECT` with value `Price - low to high` **(wrong)**
SC	`CLICK` on the expected control **(correct)**

Interpretation. CMIB predicts a directionally reasonable high-level intent, namely sorting by low price, but applies it with the wrong operation type at the wrong step. SC succeeds by following the immediate UI affordance. This is a representative failure mode: skill cards can sometimes overemphasize a learned task policy or future constraint, causing the model to lose synchronization with the current page state.

Overall, these cases suggest that CMIB is most helpful when the next action depends on workflow state and executable UI affordances, such as opening a location editor or clicking a result link. The failure case also clarifies why SC can outperform CMIB on selected Step SR or element-accuracy rows: when the current step is a simple local click and CMIB's retrieved skill is too abstract, SC may better match the immediate UI state. Thus, CMIB improves consistency and many long-horizon grounding decisions, while still leaving room for better state-aware gating of skill guidance.

D Factorization underlying CMIB

Proof of Lemma 3.2. Applying the chain rule of mutual information to the first term in Equation (3) gives

$$I\big((X,M);\,(c,z)\big) = I\big((X,M);c\big) + I\big((X,M);z \mid c\big). \tag{19}$$

Likewise, applying the chain rule to the second term gives

$$I\big((c,z);Y\big) = I(c;Y) + I(z;Y \mid c). \tag{20}$$

Substituting Equations (19) and (20) into Equation (3), the objective can be written as

$$\Big[I\big((X,M);c\big) - \beta\, I(c;Y)\Big] + \Big[I\big((X,M);z \mid c\big) - \beta\, I(z;Y \mid c)\Big]. \tag{21}$$

This yields the claimed exact two-stage factorization. The generalized objective in Equation (4) is then obtained by replacing the shared coefficient β with stage-specific coefficients β_c and β_z. When $\beta_c = \beta_z = \beta$, Equation (4) reduces to Equation (21). □

E Variational surrogate of the conditional multimodal bottleneck

Proof of Lemma 3.3. We first control the conditional compression term. Since $Z \perp X \mid (M, c^*)$, we have

$$I((X,M);z \mid c^*) = I(M;z \mid c^*). \tag{22}$$

Now define the aggregated posterior

$$q_\theta(z \mid c^*) = \int q_\theta(z \mid M, c^*)\, p(M \mid c^*)\, dM. \tag{23}$$

Using the standard variational decomposition,

$$\begin{aligned}
&\mathop{\mathbb{E}}_{M\sim p(\cdot|c^*)}\big[\mathrm{KL}\big(q_\theta(z \mid M, c^*) \,\|\, r_\phi(z \mid c^*)\big)\big] \\
&\quad = I(M; z \mid c^*) \\
&\qquad + \mathrm{KL}\big(q_\theta(z \mid c^*) \,\|\, r_\phi(z \mid c^*)\big) \\
&\quad \geq I(M; z \mid c^*). \qquad (24)
\end{aligned}$$

Combining Equations (22) and (24) yields

$$I((X, M); z \mid c^*) \;\leq\; \mathop{\mathbb{E}}_{M\sim p(\cdot|c^*)}\Bigg[\mathrm{KL}\big(q_\theta(z \mid M, c^*) \,\|\, r_\phi(z \mid c^*)\big)\Bigg]. \qquad (25)$$

Next we lower-bound the conditional relevance term. By the definition of conditional mutual information,

$$I(z; Y \mid c^*) = H(Y \mid c^*) - H(Y \mid z, c^*). \quad (26)$$

For any predictive distribution $\pi_{\mathrm{tsk}}(Y \mid [\,g_\omega(z);\ c^*;\ \mathcal{B}\,])$, the conditional cross-entropy upper-bounds the conditional entropy:

$$H(Y \mid z, c^*) \;\leq\; \mathop{\mathbb{E}}_{\substack{(M,Y)\sim p(\cdot,\cdot|c^*)\\ z\sim q_\theta(\cdot|M,c^*)}}\Bigg[-\log \pi_{\mathrm{tsk}}(Y \mid [\,g_\omega(z);\ c^*;\ \mathcal{B}\,])\Bigg]. \qquad (27)$$

Substituting Equation (27) into Equation (26) gives

$$I(z; Y \mid c^*) \;\geq\; H(Y \mid c^*) + \mathop{\mathbb{E}}_{\substack{(M,Y)\sim p(\cdot,\cdot|c^*)\\ z\sim q_\theta(\cdot|M,c^*)}}\Bigg[\log \pi_{\mathrm{tsk}}(Y \mid [\,g_\omega(z);\ c^*;\ \mathcal{B}\,])\Bigg]. \qquad (28)$$

Finally, combining Equations (25) and (28) with the definition of $\mathcal{L}_z$ in Equation (9), we obtain

$$\begin{aligned}
\mathcal{L}_z &= I((X, M); z \mid c^*) - \beta_z I(z; Y \mid c^*) \\
&\leq \mathop{\mathbb{E}}_{\substack{(M,Y)\sim p(\cdot,\cdot|c^*)\\ z\sim q_\theta(\cdot|M,c^*)}}\Bigg[\log \frac{q_\theta(z \mid M, c^*)}{r_\phi(z \mid c^*)} \qquad (29) \\
&\quad - \beta_z \log \pi_{\mathrm{tsk}}(Y \mid [\,g_\omega(z);\ c^*;\ \mathcal{B}\,])\Bigg] - \beta_z H(Y \mid c^*) \\
&= \mathcal{J}_z(\theta, \phi; c^*) - \beta_z H(Y \mid c^*), \qquad (30)
\end{aligned}$$

which proves the claim. □