EARLYKNOWBENCH: WHEN DOES A GUI ASSISTANT KNOW WHAT THE USER WANTS, AND DOES IT KEEP KNOWING?

Anonymous authors
Paper under double-blind review

ABSTRACT

A proactive mobile GUI agent has to infer the user's goal from a partial trajectory, early enough to intervene, and sustain that prediction as the interaction continues. Existing GUI benchmarks score only a single point, usually the final state of a completed trajectory, which cannot separate a model that recognized the goal early and held it from one that reached the same answer too late to act on. EarlyKnowBench elicits the hidden session goal at every observation of a recorded session and scores the prediction sequence against an annotated *full evidence support time* t^*, the earliest step where the observation's screenshots and completed actions fully specify the goal. Grounding t^* in the trajectory rather than model outputs makes early recognition and subsequent retention directly comparable across both sessions and models. We annotate t^* across 1,503 sessions and 29,630 interaction steps from FingerTip 20K, evaluating 12 models across every step. Our findings reveal that models struggle to correctly infer the user's goal before t^*, and even when they succeed early, they systematically fail to retain that prediction over time. Annotated goals improve early action success for 10 of 12 models, but inferred goals often underdeliver. These insights indicate that future proactive agents must look beyond raw scaling to focus explicitly on early intent recognition, temporal prediction retention, and downstream execution alignment.[1]

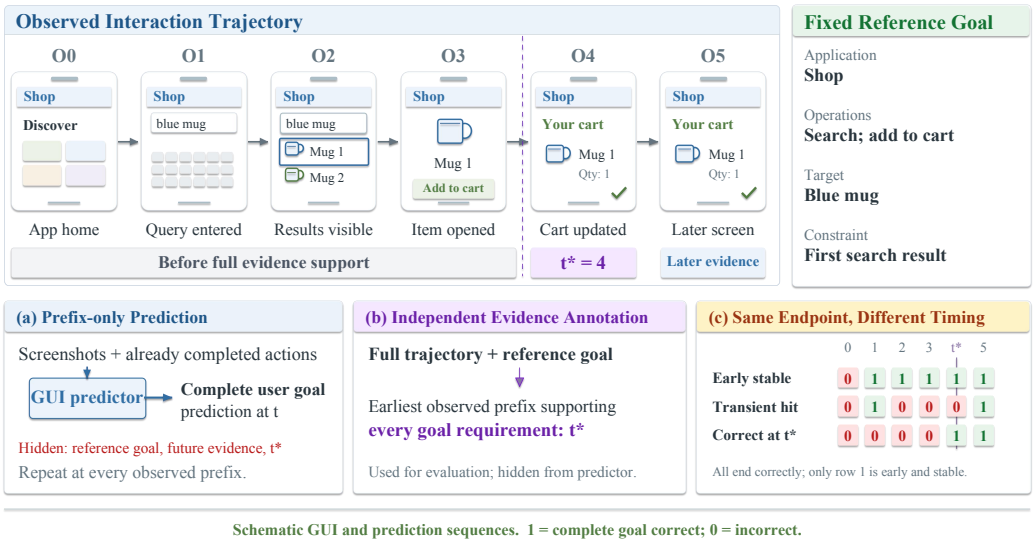

Figure 1: EarlyKnowBench overview. Predicting the hidden session goal at every observation yields a sequence rather than a single score. A separately annotated full support time (here, $t^* = 4$) remains hidden from the model. This reveals how sequences with identical endpoint scores can differ drastically in early recognition and persistence.

[1]The code and data are available for review at `https://anonymous.4open.science/r/EarlyKnowBench-188A/`.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Mobile GUI agents are typically evaluated in a reactive setting: given an explicit instruction, they execute a sequence of actions (Rawles et al., 2025; Li et al., 2024; Qin et al., 2025). Proactive assistance inverts this paradigm, requiring the agent to infer an unstated, eventual goal from an ongoing interaction trajectory (Lu et al., 2025; Berkovitch et al., 2025). As established in Bayesian user modeling and mixed-initiative interfaces, turning uncertain goal inference into effective assistance depends on both *what* the model predicts and *when* (Horvitz et al., 1998; Horvitz, 1999; Chen et al., 2025). An assistant that figures out the user's intent only after the task is finished misses the window to help; it can no longer pre-fill forms, prepare information, or prevent errors along the way. For a GUI assistant, useful anticipation therefore requires two capabilities: predicting a complete goal early while visual evidence is still incomplete, and stably retaining that prediction as new screens arrive.

Existing proactive GUI benchmarks primarily evaluate intent using endpoint, segment-level, or fixed-ratio scores (Yang et al., 2026a;b; Chai et al., 2026; Yan et al., 2026; Pawar et al., 2025). However, an endpoint score cannot distinguish a sustained early prediction from a goal recognized only at the final step, when it is too late to help. Similarly, fixed observation ratios fail to reveal when a goal is first recognized relative to screen evidence, or whether that recognition persists. While early classification formalizes the trade-off between prediction error and waiting cost (Dachraoui et al., 2015), GUI goal inference requires an objective reference for when observations fully support the intent. A global step index cannot provide this reference because evidence accumulates at different rates across sessions; for instance, opening a product page could precede either item comparison or purchase, requiring subsequent actions to resolve ambiguity.

We introduce EarlyKnowBench, a streaming evaluation grounded by the *full support time* (t^*): the earliest observation where recorded screenshots and completed actions support every goal component (application, operations, entities, and constraints). Derived from the interaction record rather than model output, t^* provides a model-agnostic boundary between extrapolation from partial evidence and persistence under later observations. We evaluate 12 models on 1,503 FingerTip sessions and 29,630 prefixes (Yang et al., 2026a), using the same cohort for goal recognition and downstream action prediction. To address ambiguity before t^*, we solicit five ranked hypotheses per prefix and judge them independently so that valid alternatives are measured rather than penalized. The downstream next-action arm conditions on the prefix alone, the annotated goal, or the predicted goal to quantify the actionable utility of early recognition.

Our contributions are summarized as follows:

- We formulate proactive goal inference as a streaming, ranked prediction task over 29,630 prefixes, introducing the *full support time* (t^*) to objectively disentangle early anticipation from temporal persistence across varying session lengths.
- We introduce a paired downstream next-action framework evaluated before t^* that measures the actionable utility of inferred goals, decoupling element grounding from stop-decision policies.
- Evaluating 12 diverse models, we reveal a critical misalignment: scale significantly improves early goal anticipation but fails to improve intent retention.

2 RELATED WORK

Table 1 compares the inputs and temporal evaluation of intent datasets and GUI benchmarks.

2.1 INSTRUCTION-FOLLOWING AGENTS AND INTENT RECOGNITION

Instruction-following benchmarks cover mobile control (Rawles et al., 2023; 2025; Li et al., 2024), web interaction (Deng et al., 2023; Zhou et al., 2024), and desktop tasks (Xie et al., 2024). SeeClick and OS-ATLAS develop visual grounding capabilities (Cheng et al., 2024; Wu et al., 2025), while ScreenSpot-Pro tests grounding in professional, high-resolution interfaces (Li et al., 2025). GUI agents further learn multistep control through reinforcement learning (Wang et al., 2025; Zhang et al., 2025). These settings emphasize executing a supplied goal; we study when that goal can be inferred from user activity.

Table 1: Benchmark comparison. No instruction means no explicit user task request is supplied. FingerTip and AndroidIntent use their suggestion tracks; Act2Intention uses understanding and prediction, and our row uses the primary setting. GUIDE supplies candidate goals. Evidence time independently marks full intent support; Early recognition measures correctness before it, and Stable recognition measures correctness retained through the endpoint.

Benchmark	Input	No instruction	Evidence time	Early recognition	Stable recognition
SNIPS	Utterance	✗	✗	✗	✗
CLINC150	Utterance	✗	✗	✗	✗
BANKING77	Utterance	✗	✗	✗	✗
AndroidControl	GUI + instruction	✗	✗	✗	✗
FingerTip 20K	GUI + context	✓	✗	✗	✗
AndroidIntent	User state	✓	✗	✗	✗
GUIDE	GUI clip	✓	✗	✗	✗
PIRA-Bench	GUI + profile	✓	✗	✗	✗
Act2Intention	Actions / intents	✓	✗	✗	✗
EarlyKnowBench	GUI prefix	✓	✓	✓	✓

Classical plan recognition infers goals from partially observed actions under a symbolic domain model (Ramírez & Geffner, 2009; 2010). Utterance intent classification (Coucke et al., 2018; Larson et al., 2019; Casanueva et al., 2020) and usage-based action or preference prediction (Agrawal et al., 2018; Zhao et al., 2019; Zhang et al., 2022) use different targets. We track a complete natural-language session goal, measuring both the onset and persistence of correctness.

2.2 Proactive Assistance from Partial Observations

Proactive Agent learns when and what assistance to offer from event histories (Lu et al., 2025), while AppAgent-Pro expands user instructions with cross-application information retrieval (Zhao et al., 2025). GUI benchmarks study task suggestion (Yang et al., 2026a), preference inference (Lyu et al., 2026), and future intentions (Chai et al., 2026; Yang et al., 2026b; Yan et al., 2026). Act2Intention predicts current or next intentions; we keep one session goal fixed so correctness is comparable over time. Early classification balances error and delay, including through learned halting policies (Dachraoui et al., 2015; Hartvigsen et al., 2019). Here, t^* is an annotation-derived evidence boundary, independent of a model's decision to predict. Prefix-level evaluation separates anticipation from later recognition, measures persistence, and connects inferred goals to action prediction.

3 EarlyKnowBench

3.1 Problem Formulation

A proactive assistant receives a growing prefix rather than a full trajectory, so it must predict at every observation. Using zero-based indices, the input and prediction at observation t are:

$$H_{i,t} = (O_{i,0}, A_{i,0}, \ldots, A_{i,t-1}, O_{i,t}), \qquad \hat{Y}_{i,t} = f(H_{i,t}). \tag{1}$$

Session i has N_i observations, $O_{i,t}$ is a screenshot, $A_{i,t-1}$ the preceding completed action, and f the evaluated goal predictor. The reference goal Y_i is fixed across all prefixes, so $\mathrm{Acc}_{i,t}$ measures the same target at every t. Each prefix is queried independently, with future observations, annotations, and earlier predictions hidden, yielding a sequence $(\mathrm{Acc}_{i,0}, \ldots, \mathrm{Acc}_{i,N_i-1})$ that evaluates the trajectory's shape rather than its endpoint alone.

3.2 Sources and Dataset Construction

EarlyKnowBench is built from FingerTip 20K (Yang et al., 2026a) (user-driven interactions) and AndroidControl (Li et al., 2024) (instruction-driven demonstrations). Sessions are selected via four criteria: usable ordered observations, goal-trajectory agreement, verifiable evidence for all goal components, and a full support time strictly after the first observation ($t^* > 0$). Source instructions and typed strings in metadata are withheld from the predictor.

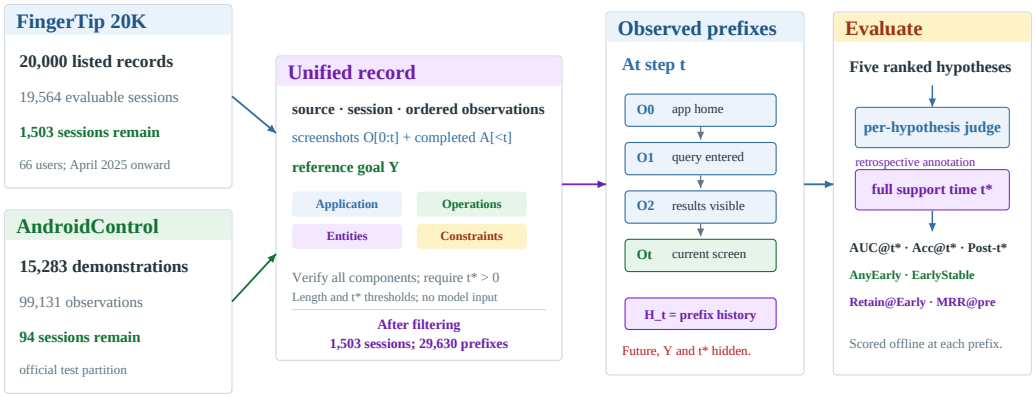

Figure 2: Construction of EarlyKnowBench from FingerTip 20K and AndroidControl. Each record pairs an ordered interaction with a reference user goal and component evidence annotations, and an eligible session supports every required goal component and has at least one observation before t^*. A prefix contains the current screenshot and all preceding executed actions and screenshots; the predictor receives only that, while the reference goal and temporal annotations are used for evaluation.

A strict calendar boundary (April 1, 2025) defines the FingerTip test set, retaining all 83 users without activity filtering; AndroidControl retains its official test partition. The cohort filters of Section 3.5 then select the 1,503 FingerTip sessions and 29,630 prefixes used in the primary results. Before action-cohort filtering, FingerTip sessions average 13.64 observations (median 11) with 15.69% exceeding 20 observations, whereas AndroidControl averages 6.40 (median 6) with 0.36% exceeding 20.

Human agreement on what the benchmark adds. Construction adds two judgements to the recorded sessions: where t^* falls, which decides how every metric is split, and whether a predicted goal matches the reference, which decides whether a prefix counts as correct. Both audits are complete: two annotators independently reviewed all 150 timing sessions and all 300 semantic-judgement pairs, with the stored annotations and judge verdicts hidden during review. For full-support timing, model annotations match the final human reference exactly in 94.00% of sessions and within one frame in 97.33%, with a mean absolute difference of 0.093 frames. The two annotators agree exactly in 97.33% of sessions and within one frame in 98.67%, with a mean absolute difference of 0.047 frames; the median absolute difference is zero for both comparisons. Because a one-observation deviation shifts a session's split by a single prefix, the interval metrics of Section 3.6 are session means rather than pooled over prefixes. For semantic judgements, the annotators agree on 298 of 300 pairs (99.33%), and the LLM judge agrees with each annotator on 297 of 300 pairs (99.00%). These audits establish agreement on both the timing annotations and the semantic verdicts used to score prediction sequences; Appendix A gives the sampling protocol and complete results.

3.3 Reference Goal and the Full Support Time

A fixed step index is not a reliable temporal reference, because evidence arrives at widely different points across sessions. The reference goal is $Y_i = (a_i, \mathbf{o}_i, \mathbf{e}_i, \mathbf{c}_i)$: the required application, operations, entities with roles, and constraints. The *full support time* t_i^* is the earliest observation where the recorded trajectory supports every required component $r \in R(Y_i)$:

$$t_i^* = \min\{t : V(r, H_{i,t}) = 1 \text{ for every } r \in R(Y_i)\}. \tag{2}$$

Support requires completed actions or traceable state changes, not merely visible on-screen elements. Computed independently of model predictions, t^* splits each session: predicting correctly before t^* requires extrapolating ahead of the evidence, while correctness from t^* onward tests state retention.

3.4 Ranked Goal Hypotheses

Because pre-t^* prefixes are ambiguous, the predictor emits five ranked hypotheses per prefix:

$$f(H_{i,t}) = \big(\hat{Y}_{i,t}^{(1)}, \ldots, \hat{Y}_{i,t}^{(K_{i,t})}\big), \qquad \hat{Y}_{i,t} = \hat{Y}_{i,t}^{(1)}. \tag{3}$$

Near-identical hypotheses are deduplicated, yielding a realized count $K_{i,t}$. A judge returns an independent 0/1 verdict on every candidate, so ambiguity is measured without penalizing paraphrases.

3.5 Downstream Action Utility

To quantify what an inferred goal is worth downstream, we evaluate next-action prediction under three conditions, altering only the goal information provided to the action predictor g:

$$\hat{A}_{i,t}^{\text{none}} = g(H_{i,t}), \qquad \hat{A}_{i,t}^{\text{gold}} = g(H_{i,t}, Y_i), \qquad \hat{A}_{i,t}^{\text{pred}} = g(H_{i,t}, \hat{Y}_{i,t}). \tag{4}$$

Correctness is scored against the recorded user trajectory $A_{i,t}$. To prevent post-t^* information leakage from skewing the results, we report the gap ($\text{Act}^{\text{gold}} - \text{Act}^{\text{none}}$) separately before and after t^*. We further isolate terminal prefixes (stop decisions) from non-terminal ones (element grounding).

Action cohort and matching. The action cohort keeps sessions with a readable pre-t^* interval: ≥ 11 observations, ≥ 4 pre-t^* prefixes, and for FingerTip a date in April 11–30, 2025, giving 1,503 FingerTip sessions (29,630 prefixes; 16,800 pre-t^*) over 66 users and 94 AndroidControl sessions (1,273 prefixes). Actions match when type and arguments agree; element-based actions are scored as *Bounds*, *Node* (primary, requiring the same accessibility-tree node), or *Distance* (< 0.14). We read *Node* because *Bounds* rewards clicks anywhere inside a large container.

3.6 Evaluation Metrics

Complete-goal correctness and temporal AUC. Accuracy ($\text{Acc}_{i,t} \in \{0, 1\}$) requires all four slots (application, operation, target, constraints) to match the reference, and End is accuracy at the final observation. We summarize temporal area under the curve (AUC) up to t^* and the endpoint:

$$\text{AUC@}t^* = \text{Avg}_{i \in \mathcal{I}} \left[\frac{1}{t_i^*} \sum_{t=1}^{t_i^*} \frac{\text{Acc}_{i,t-1} + \text{Acc}_{i,t}}{2} \right], \tag{5}$$

$$\text{AUC@End} = \text{Avg}_{i \in \mathcal{I}} \left[\frac{1}{N_i - 1} \sum_{t=1}^{N_i - 1} \frac{\text{Acc}_{i,t-1} + \text{Acc}_{i,t}}{2} \right]. \tag{6}$$

Early correctness and retention. We separate early anticipation from stable retention:

$$\text{AnyEarly} = \text{Avg}_{i \in \mathcal{I}}[\mathbf{1}\{\exists t < t_i^* : \text{Acc}_{i,t} = 1\}], \tag{7}$$

$$\text{EarlyStable} = \text{Avg}_{i \in \mathcal{I}}[\mathbf{1}\{\exists t < t_i^* : \text{ Acc}_{i,v} = 1 \text{ for all } v = t, \ldots, N_i - 1\}]. \tag{8}$$

AnyEarly bounds how often a model could theoretically help, EarlyStable measures when an assistant acts on a prediction without later contradicting itself, and their ratio $\text{Retain@Early} = \text{EarlyStable}/\text{AnyEarly}$ isolates a model's capacity to maintain early hits.

Evidence-aligned and ranked metrics. We report recognition at full support ($\text{Acc@}t^*$), its trapezoidal integral up to full support ($\text{AUC@}t^*$), and average maintenance afterwards (Post-t^*). For ranked outputs, $\text{Recall@}k$ is the share of prefixes with a correct hypothesis at rank k or better; a prefix has exactly one reference goal, so rank-k correctness is a recall over that single target rather than an accuracy over a label set. Complete-goal comparisons read rank 1, and the Recall@5 gain tests ambiguity resolution against the realized candidate count $K_{i,t}$. From the credited rank $\rho_{i,t}$ we report MRR@pre, restricted to the prefixes before t^*.

Action utility. For interval $\mathcal{T}$, action accuracy $\text{Act}^c(\mathcal{T})$ is the session-mean match rate for condition c. We evaluate the theoretical headroom (Δ_{head}) and the realized delivery (Δ_{real}):

$$\Delta_{\text{head}}(\mathcal{T}) = \text{Act}^{\text{gold}}(\mathcal{T}) - \text{Act}^{\text{none}}(\mathcal{T}), \qquad \Delta_{\text{real}}(\mathcal{T}) = \text{Act}^{\text{pred}}(\mathcal{T}) - \text{Act}^{\text{none}}(\mathcal{T}). \tag{9}$$

Comparisons use paired McNemar tests with continuity correction on shared prefixes. A match requires the reference gesture exactly: unlike the published matcher we adopt, we do not accept a short press against a long press, worth at most 0.04 points of success on any row. Beside type accuracy and success, the published evaluator's own quantities, we report one it does not define: the *false-finish rate*, the share of non-terminal prefixes at or after t^* where the predictor declares the task finished (lower is better). It adds no criterion to the matcher, which already charges such a completion as a type mismatch, and separates a stopping failure from a grounding one (Appendix B).

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

We evaluate all models on the shared test set (Figure 2), weighting each session equally, with 95% confidence intervals from 2,000 session-bootstrap resamples; configurations with incomplete scoring are omitted so every table row reflects identical interactions.

Models and baselines. Two families share a common prompt and deterministic decoding. General-multimodal VLMs (Qwen3.5 0.8B–9B; Qwen Team 2026) isolate the effect of parameter scale. GUI-specialized models are UI-Voyager (Lin et al., 2026), GUI-Owl-1.5-2B (Xu et al., 2026), MAI-UI-2B (Zhou et al., 2025), UI-Venus-1.5 2B/8B (Venus Team et al., 2026b), UI-TARS-1.5-7B (Qin et al., 2025), UI-Mate-9B (Ding et al., 2026), and UI-Venus-2-9B (Venus Team et al., 2026a); they test whether interface-grounding pretraining translates to session-level goal persistence.

Protocols and scoring. We run the ranked and action protocols (Sections 3.4 and 3.5) unmodified, with the action predictor consuming the rank-1 goal the same model inferred at the current prefix, which is what a deployed assistant knows in real time. A fixed prompt compares predicted and reference goals, and all models share one Qwen3.8-27B judge so no row is scored by a different grader; temporal annotations are independent of the semantic judgments. Before aggregation we validate sequence completeness, deduplicate prefix scores, and check target agreement.

4.2 OVERALL BENCHMARK RESULTS

Table 2: Early goal recognition and the downstream value of the goal a model infers for itself, on all 29,630 prefixes of the 1,503-session action cohort. Both halves use the same cohort; action columns condition on the rank-1 prediction. Success pools all prefixes, while node accuracy is restricted to $t < t^*$. Recognition columns are equal-weight session means in percent under one Qwen3.8-27B judge, defined in Section 3.6. Bold marks the best value in a column, underline the second best. Cohort and denominators: Appendix B.

Model	EARLY GOAL RECOGNITION							DOWNSTREAM ACTION	
	AUC @t^*	Acc @t^*	Post-t^*	Any Early	Early Stable	Retain @Early	MRR @pre	Success all t	Node acc. $t < t^*$
General-multimodal models									
Qwen3.5-0.8B	0.77	2.21	2.38	3.15	0.33	10.64	0.75	3.76	14.86
Qwen3.5-2B	3.72	10.22	8.18	16.11	2.03	12.61	3.68	6.20	17.09
Qwen3.5-4B	7.57	21.76	17.25	32.14	4.52	**14.08**	8.04	17.84	16.19
Qwen3.5-9B	7.85	21.62	17.14	33.60	4.46	13.27	8.45	17.62	19.25
GUI-specialized models									
UI-Voyager	0.53	1.09	0.89	2.38	0.14	5.71	0.48	4.79	16.18
GUI-Owl-1.5-2B	2.63	8.32	6.35	11.24	1.40	12.43	2.34	4.24	8.21
MAI-UI-2B	3.25	10.05	8.66	13.77	1.86	<u>13.53</u>	3.22	6.05	19.29
UI-Venus-1.5-2B	3.99	12.32	11.09	18.18	2.06	11.36	3.92	7.56	21.75
UI-TARS-1.5-7B	6.40	15.97	13.06	24.62	2.66	10.81	5.95	13.98	14.76
UI-Venus-1.5-8B	7.13	19.43	15.72	28.94	3.86	13.33	7.17	**21.22**	**23.10**
UI-Mate-9B	<u>9.02</u>	<u>28.28</u>	**21.59**	<u>36.93</u>	<u>4.72</u>	12.79	<u>9.88</u>	15.23	11.75
UI-Venus-2-9B	**9.61**	**29.74**	<u>21.23</u>	**40.92**	**4.79**	11.71	**10.10**	<u>19.29</u>	<u>22.27</u>

Table 2 reports FingerTip rank-1 recognition together with the action value of each model's own inferred goal, both read on every prefix of the action cohort. Its action columns are the full-action match over all prefixes and, before t^*, *Node* grounding on the reference-eligible denominator. Reading a row straight across shows that recognizing a goal and acting on one are distinct abilities: UI-Mate-9B is the second-strongest recognizer at t^* (28.28%) yet grounds second-worst among the GUI-specialized rows (11.75%), while UI-Venus-1.5-8B recognizes nine points less (19.43%) and grounds best (23.10%). Across the twelve rows Acc@t^* ranks with pooled success at Spearman 0.89 but with pre-t^* node accuracy at only 0.30, so recognition predicts how often an actor is right overall and barely predicts whether it uses the goal before the record contains it.

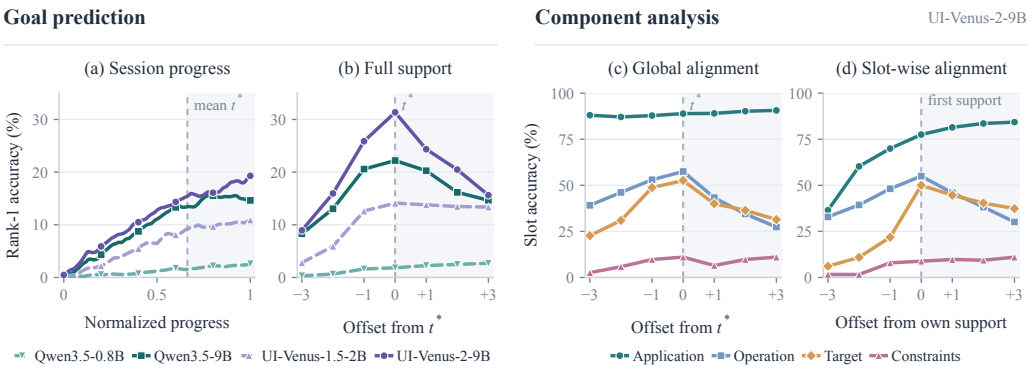

Figure 3: What happens to a goal over a session, on the 1,503-session cohort of Table 2, read from rank 1. (a) Accuracy over normalized session progress, sessions weighted equally, dashed line at the mean full-support position ($t^* = 0.663$). (b) Accuracy by offset from t^* on the 928 sessions supplying a prefix at every offset, so composition is fixed across the curve. (c) and (d) read UI-Venus-2-9B's four slots against the global t^* and against each slot's own first support. Panels (a) and (b) share a vertical scale, (c) and (d) a separate one; shading marks observations at or after the reference point.

5 Analysis

5.1 Endpoint Scores Do Not Recover Temporal Behavior

Endpoint correctness evaluates the final observation alone and says nothing about the path that produced it, so Table 2 reports early recognition and retention, with endpoint scores in Table 8. In every row a model reaches a correct goal before full support far more often than one lasting to the end of the session, by $7.1\times$ to $17.0\times$ (AnyEarly against EarlyStable), and more often than it is correct at the endpoint at all, by $1.21\times$ to $4.41\times$; one endpoint score fits either case. Models ending at similar values also show very different pre-t^* curves (Figure 3a, plotting the extreme scale of each family), and a late-rising curve offers a proactive assistant no actionable window.

The gap is visible even among the strongest rows. UI-Venus-2-9B reaches the goal early in 40.92% of sessions, but sustains an early prediction to the endpoint in 4.79%, a gap of 36.13 percentage points. Its endpoint accuracy is 19.29%, so endpoint success also includes sessions without stable early recognition. Qwen3.5-4B and 9B have identical endpoint accuracy (14.64%) despite different AnyEarly rates (32.14% and 33.60%). The three metrics distinguish finding the goal once, ending with it, and maintaining an early prediction.

5.2 Completing the Evidence Does Not Consolidate the Hypothesis

If post-t^* failures merely reflected insufficient information, accuracy should rise once all goal components are supported. It does not. Read on the 928 sessions matched at every offset from $t^* - 3$ to $t^* + 3$ (Figure 3b), nine of the twelve rows peak exactly at t^* and eleven are lower three observations later than they were at full support; UI-Venus-2-9B falls from 31.36% to 15.62% and Qwen3.5-9B from 22.20% to 14.66%. The one exception, Qwen3.5-0.8B, never rises above 2.69% in the window and has nothing to lose. Because the composition is fixed across the window and later observations only add evidence, this decline is models discarding correctly formed goals rather than a change in which sessions are averaged. Panel (a) shows the same thing on a progress axis, so the loss is not an artifact of aligning on t^*.

Decomposing accuracy by slot isolates which part of the goal is lost (Figure 3c). For UI-Venus-2-9B application accuracy is flat and even rises over the window (88.90% to 90.62%), while operation falls 30.10 points and target 21.19: models keep track of which application is in use and lose the specific operation and its parameters, the part an assistant needs to act. Panel (d) is the control: read against each slot's support rather than the global t^*, every slot steps up exactly at offset zero, most sharply target (+28.16 points), as predicted by the annotations.

This evaluation measures consistency across independently queried prefixes: a later failure occurs despite receiving the earlier observations again. The declining curve therefore identifies a failure to preserve a correct interpretation as evidence accumulates. It does not by itself identify whether the cause is visual attention, context use, or the decoding policy.

5.3 Scale Buys Coverage, Not Retention

Retain@Early distinguishes whether EarlyStable improves through more early hits or better retention, and the scale comparison points to coverage. Within the Qwen3.5 family (Table 2), scaling from 0.8B to 9B grows AnyEarly 10.7× (3.15% to 33.60%), yet Retain@Early moves only from 10.64% to 13.27% and is not monotone, peaking at 4B (14.08%). The model with the highest AnyEarly rate rarely has the highest Retain@Early: models trade early opportunities against retention.

The largest Qwen3.5 model also shows diminishing gains over 4B. Moving from 4B to 9B adds 1.46 points of AnyEarly, while Acc@t^* changes from 21.76% to 21.62% and EarlyStable from 4.52% to 4.46%. These are descriptive differences, rather than evidence of a statistically established reversal. Across families, UI-Venus-2-9B has the highest AnyEarly and EarlyStable, but Qwen3.5-4B has the highest Retain@Early. A larger supply of early hits can thus produce more stable successes without improving the conditional fraction retained.

The retention bottleneck remains large even for the best conditional score: a Retain@Early of 14.08% means that 85.92% of sessions with an early hit do not achieve EarlyStable. Increasing AnyEarly expands the set of opportunities available to an assistant, but the ratio shows how few of those opportunities survive the requirement of a correct suffix beginning before full support.

5.4 Recognition and Action Grounding Rank Models Differently

The distinction extends to downstream actions. UI-Venus-2-9B leads five recognition columns, yet UI-Venus-1.5-8B has the highest predicted-goal action success (21.22%) and pre-t^* node accuracy (23.10%). UI-Mate-9B's strong recognition similarly does not translate into strong grounding: its 11.75% node accuracy is below MAI-UI-2B's 19.29%, despite a much higher Acc@t^* (28.28% versus 10.05%). The recognition ranking is therefore insufficient for choosing an actor that must use a predicted goal before full support.

Table 3: Pre-t^* action success (%, session means). None/Gold/Pred use no goal, the reference goal, or the inferred goal. $\delta = \mathrm{Pred} - \mathrm{Gold}$ (points). Bold: row best; stars: significant differences. Details: Appendix B.1.

Model	None	Gold	Pred	δ
General-multimodal models				
Qwen3.5-0.8B	**5.80**	5.31	5.68	+0.37
Qwen3.5-2B	**10.23**	8.23	9.44	+1.20*
Qwen3.5-4B	16.82	**19.48**	17.38	−2.09
Qwen3.5-9B	18.55	**19.86**	17.03	−2.82*
GUI-specialized models				
UI-Voyager	6.73	**6.88**	6.35	−0.54*
GUI-Owl-1.5-2B	0.90	**8.05**	5.12	−2.93*
MAI-UI-2B	7.77	**10.81**	9.35	−1.46*
UI-Venus-1.5-2B	9.48	**14.22**	11.32	−2.90*
UI-TARS-1.5-7B	9.28	9.72	**10.95**	+1.23*
UI-Venus-1.5-8B	19.38	**20.79**	18.78	−2.01*
UI-Mate-9B	13.22	**16.38**	14.47	−1.91*
UI-Venus-2-9B	20.15	**22.29**	20.62	−1.67

The two action columns answer different questions. Full-action success includes action types and arguments across the session, whereas pre-t^* node accuracy tests element selection where the recorded goal is not yet fully supported. A model may rank well on the former without doing so on the latter: Qwen3.5-4B slightly exceeds 9B in full-action success (17.84% versus 17.62%), but trails it in early node accuracy (16.19% versus 19.25%). Both columns describe the predicted-goal condition; Table 3 adds matched no-goal and annotated-goal controls before full support.

An annotated goal improves pre-t^* action success for 10 of 12 models, including all eight GUI-specialized rows: GUI-Owl-1.5-2B gains 7.15 points over no goal and UI-Venus-1.5-2B gains 4.74. No prefix in this interval has a terminal reference action, so these gains cannot come from correct stopping at the endpoint.

Inferred goals deliver less of this benefit. Pred falls below Gold in nine rows, seven with a marked difference, and below None in five. UI-TARS-1.5-7B is an exception, exceeding Gold by 1.23 points. Thus, providing the reference goal can help an actor, but supplying its own inference does not consistently realize that benefit; recognition and the use of inferred goals must be evaluated together.

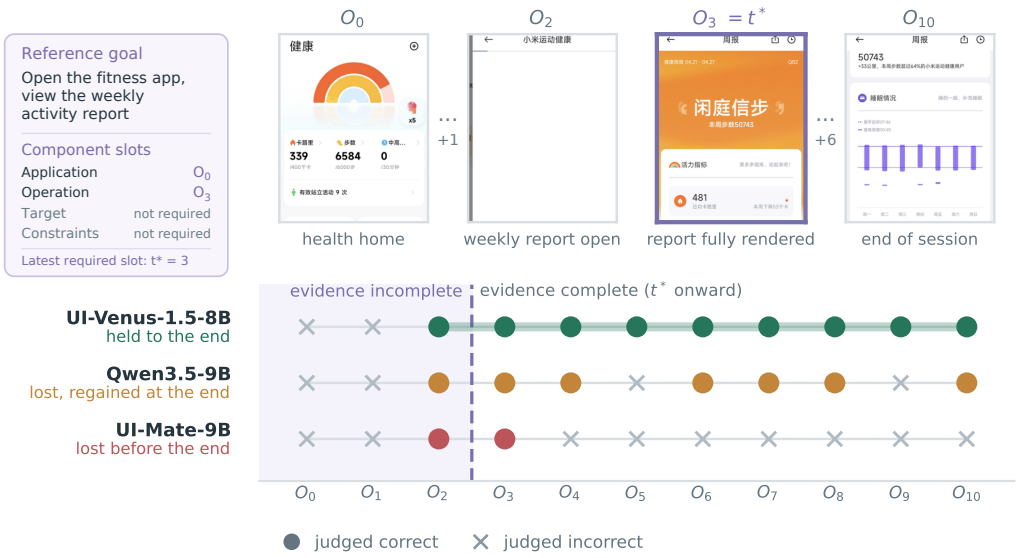

Figure 4: One FingerTip session, 11 observations with $t^* = 3$, read in Section 5.5. Only application and operation are required, and the later of the two, operation at O_3, places t^*. Screens are the recorded screenshots cropped to the application content; four are shown and each ellipsis carries the count skipped. This illustrative session lies outside the action cohort because it has only three pre-t^* prefixes. The session identifier is withheld and released with the frozen cohort manifest.

5.5 What One Session Shows That No Score Does

Figure 4 shows one session, whose reference goal is to open a fitness application and view the weekly activity report. All three models are judged correct at O_2, one observation before full support, then diverge over the screens: UI-Venus-1.5-8B holds the goal to the end, Qwen3.5-9B loses it at O_5, recovers at O_6, loses it again at O_9, and recovers at the final observation, and UI-Mate-9B loses it and does not return. An endpoint score equates the first two, since both are correct at O_{10}, though only one maintains a correct prediction throughout the remaining session; Retain@Early separates them.

On this session, all three AnyEarly indicators are one, whereas EarlyStable is one only for UI-Venus-1.5-8B. Endpoint correctness instead groups Qwen3.5-9B with UI-Venus-1.5-8B and separates UI-Mate-9B. Qwen3.5-9B's two recoveries further distinguish intermittent loss from UI-Mate-9B's persistent loss after O_3. The same reference goal and observation sequence therefore expose three behaviors: maintaining an early interpretation, repeatedly reconstructing it, and failing to recover it. Reading the sequence reveals these differences even when a selected summary score agrees.

6 Conclusion

EarlyKnowBench uses the *full support time* (t^*) to evaluate proactive GUI assistants on early recognition, temporal persistence, and downstream utility. Across 12 models and 29,630 prefixes, early recognition rarely becomes sustained correctness, and accuracy often falls after the observations fully support the goal. Scaling expands early coverage more consistently than it improves retention. Recognition and action grounding also induce different model rankings, so strong recognition scores do not imply equally strong action performance. The paired action analysis makes this distinction concrete: annotated goals improve pre-support action success in 10 of 12 rows, while inferred goals underperform annotated goals in nine and no goal in five. A useful assistant must therefore do more than produce an early correct hypothesis: it must retain that interpretation and translate the available goal information into appropriate actions. Together, these findings identify goal persistence and action grounding as distinct requirements for proactive assistance. The resulting framework supports the study of assistants that maintain and use inferred goals as observations accumulate.

AI USE STATEMENT

Generative AI tools were used to assist with language polishing and improve the clarity and readability of the manuscript. AI model APIs were also used as part of the experimental evaluation described in the paper. All AI-assisted outputs used in the manuscript or experiments were reviewed by the authors, who take responsibility for the final content and reported results.

ETHICS STATEMENT

This work is based on existing publicly released benchmark datasets and does not involve the collection of new user interaction data. We follow the access and redistribution conditions of the original datasets, particularly for GUI screenshots and interaction records that may contain user-related information. We do not identify additional ethical concerns beyond those associated with the responsible use of the source datasets.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the dataset construction, session filtering, evaluation protocols, prompts, metrics, and experimental settings in the main paper and appendix. The experiments use fixed evaluation sets and deterministic decoding where applicable, and the paper documents the procedures used to construct the reported tables and analyses. These details are provided to facilitate reproduction of the benchmark and experimental results.

REFERENCES

Rakshit Agrawal, Anwar Habeeb, and Chih-Hsin Hsueh. Learning user intent from action sequences on interactive systems. In *Proceedings of the AAAI Workshop on Web and Social Media*, 2018.

Omri Berkovitch, Sapir Caduri, Noam Kahlon, Anatoly Efros, Avi Caciularu, and Ido Dagan. Identifying user goals from UI trajectories. In *Companion Proceedings of the ACM Web Conference 2025*, pp. 2381–2390. Association for Computing Machinery, 2025. doi: 10.1145/3701716.3717525.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pp. 38–45, 2020. doi: 10.18653/v1/2020.nlp4convai-1.5.

Yuxiang Chai, Shunye Tang, Han Xiao, Rui Liu, and Hongsheng Li. PIRA-Bench: A transition from reactive GUI agents to GUI-based proactive intent recommendation agents. *arXiv preprint arXiv:2603.08013*, 2026.

Valerie Chen, Alan Zhu, Sebastian Zhao, Hussein Mozannar, David Sontag, and Ameet Talwalkar. Need help? designing proactive AI assistants for programming. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, 2025. doi: 10.1145/3706598.3714002.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. 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)*, pp. 9313–9332, 2024.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Maël Primet, and Joseph Dureau. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*, 2018.

Asma Dachraoui, Alexis Bondu, and Antoine Cornuéjols. Early classification of time series as a non myopic sequential decision making problem. In *Machine Learning and Knowledge Discovery in Databases, ECML PKDD 2015, Part I*, pp. 433–447. Springer, 2015. doi: 10.1007/978-3-319-23528-8_27.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2Web: Towards a generalist agent for the web. In *Advances in Neural Information Processing Systems*, 2023.

Zihan Ding, Longxu Dou, Qi Gao, Xiangwu Guo, Shengchao Hu, Zilong Huang, Zihang Jiang, Lei Ke, Mengcheng Lan, Weixian Lei, Hanxuan Li, Honglin Li, Xiyun Li, Zaitang Li, Leowei Liang, Xin Luo, Haozhe Ma, Jiayi Mao, Zhoujie Pan, Can Qin, Tianyuan Qu, Weiqi Wang, Wenkai Wang, Yonglin Wang, Yuxin Wang, Chenxu Wu, Yingchen Yu, Chenyu Zhang, and Yuhao Zheng. UI-Mate: Advancing open-weight foundation GUI agents with in-context demonstrations. *arXiv preprint arXiv:2608.15930*, 2026.

Thomas Hartvigsen, Cansu Sen, Xiangnan Kong, and Elke A. Rundensteiner. Adaptive-halting policy network for early classification. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 101–110, 2019.

Eric Horvitz. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 159–166. ACM, 1999. doi: 10.1145/302979.303030.

Eric Horvitz, Jack S. Breese, David Heckerman, David Hovel, and Koos Rommelse. The lumière project: Bayesian user modeling for inferring the goals and needs of software users. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pp. 256–265, 1998.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 1311–1316, 2019. doi: 10.18653/v1/D19-1131.

Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Ma, Zhiyong Huang, and Tat-Seng Chua. ScreenSpot-Pro: GUI grounding for professional high-resolution computer use. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 8778–8786, 2025.

Wei Li, William Bishop, Alice Li, Chris Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. On the effects of data scale on computer control agents. *arXiv preprint arXiv:2406.03679*, 2024.

Zichuan Lin, Feiyu Liu, Yijun Yang, Jiafei Lyu, Yiming Gao, Yicheng Liu, Zhicong Lu, Yangbin Yu, Mingyu Yang, Junyou Li, Deheng Ye, and Jie Jiang. UI-Voyager: A self-evolving GUI agent learning via failed experience. *arXiv preprint arXiv:2603.24533*, 2026.

Yaxi Lu, Shenzhi Yang, Cheng Qian, Guirong Chen, Qinyu Luo, Yesai Wu, Huadong Wang, Xin Cong, Zhong Zhang, Yankai Lin, Weiwen Liu, Yasheng Wang, Zhiyuan Liu, Fangming Liu, and Maosong Sun. Proactive agent: Shifting LLM agents from reactive responses to active assistance. In *The Thirteenth International Conference on Learning Representations*. OpenReview.net, 2025.

Yibo Lyu, Gongwei Chen, Rui Shao, Weili Guan, and Liqiang Nie. PersonalAlign: Hierarchical implicit intent alignment for personalized GUI agent with long-term user-centric records. *arXiv preprint arXiv:2601.09636*, 2026.

Shraddha Vijay Pawar, Balavarun Pedapudi, Pramod Kaushik, Sarath Sivaprasad, Mario Fritz, and Shirish Karande. EARL: Early intent recognition in GUI tasks using theory of mind. In *ICML 2025 Workshop on Computer Use Agents*, 2025.

Yujia Qin, Yining Ye, Junjie Fang, Haoming Wang, Shihao Liang, Shizuo Tian, Junda Zhang, Jiahao Li, Yunxin Li, Shijue Huang, Wanjun Zhong, Kuanye Li, Jiale Yang, Yu Miao, Woyu Lin, Longxiang Liu, Xu Jiang, Qianli Ma, Jingyu Li, Xiaojun Xiao, Kai Cai, Chuang Li, Yaowei Zheng, Chaolin Jin, Chen Li, Xiao Zhou, Minchao Wang, Haoli Chen, Zhaojian Li, Haihua Yang, Haifeng Liu, Feng Lin, Tao Peng, Xin Liu, and Guang Shi. UI-TARS: Pioneering automated GUI interaction with native agents. *arXiv preprint arXiv:2501.12326*, 2025.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026.

Miquel Ramírez and Hector Geffner. Plan recognition as planning. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, pp. 1778–1783, 2009.

Miquel Ramírez and Hector Geffner. Probabilistic plan recognition using off-the-shelf classical planners. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, pp. 1121–1126, 2010. doi: 10.1609/aaai.v24i1.7745.

Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, and Timothy P. Lillicrap. AndroidInTheWild: A large-scale dataset for android device control. In *Advances in Neural Information Processing Systems*, 2023.

Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William Bishop, Wei Li, Folawiyo Campbell-Ajala, Daniel Toyama, Robert Berry, Divya Tyamagundlu, Timothy Lillicrap, and Oriana Riva. AndroidWorld: A dynamic benchmarking environment for autonomous agents. *arXiv preprint arXiv:2405.14573*, 2025.

Venus Team, Zhuohan Cai, Haoxing Chen, Jiaxuan Chen, Weizhi Chen, Changlong Gao, Zhangxuan Gu, Yuan Guo, Yusong Hu, Jianrong Jiang, Jianguo Li, Runze Li, Jinzhen Lin, Zhenyu Ma, Changhua Meng, Han Peng, Xinyu Qiu, Shuheng Shen, Zhongyi Shui, Weiqiang Wang, Ming Wen, Zhuoer Xu, Hang Yan, Kaiwen Yang, Ruilin Yao, Nanjun Yu, Zhengwen Zeng, Lianrui Zhang, Yunzhu Zhang, Zhe Zhao, and Beitong Zhou. UI-Venus-2 technical report. *arXiv preprint arXiv:2609.00028*, 2026a.

Venus Team, Changlong Gao, Zhangxuan Gu, Yulin Liu, Xinyu Qiu, Shuheng Shen, Yue Wen, Tianyu Xia, Zhenyu Xu, Zhengwen Zeng, Beitong Zhou, Xingran Zhou, Weizhi Chen, Sunhao Dai, Jingya Dou, Yichen Gong, Yuan Guo, Zhenlin Guo, Feng Li, Qian Li, Jinzhen Lin, Yuqi Zhou, Linchao Zhu, Liang Chen, Zhenyu Guo, Changhua Meng, and Weiqiang Wang. UI-Venus-1.5 technical report. *arXiv preprint arXiv:2602.09082*, 2026b.

Haoming Wang, Haoyang Zou, Huatong Song, Jiazhan Feng, Junjie Fang, et al. UI-TARS-2 technical report: Advancing GUI agent with multi-turn reinforcement learning. *arXiv preprint arXiv:2509.02544*, 2025.

Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, and Yu Qiao. OS-ATLAS: Foundation action model for generalist GUI agents. In *The Thirteenth International Conference on Learning Representations*, 2025.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *Advances in Neural Information Processing Systems*, 2024.

Haiyang Xu, Xi Zhang, Haowei Liu, Junyang Wang, Zhaozai Zhu, Shengjie Zhou, Xuhao Hu, Feiyu Gao, Junjie Cao, Zihua Wang, Zhiyuan Chen, Jitong Liao, Qi Zheng, Jiahui Zeng, Ze Xu, Shuai Bai, Junyang Lin, Jingren Zhou, and Ming Yan. Mobile-Agent-v3.5: Multi-platform fundamental GUI agents. *arXiv preprint arXiv:2602.16855*, 2026.

Xiaokai Yan, Jingtao Ding, Yong Li, and Zhiwen Yu. Act2Intention: A benchmark for developing active mobile agents through inferring user intention from GUI actions. *arXiv preprint arXiv:2608.14132*, 2026.

Qinglong Yang, Haoming Li, Haotian Zhao, Xiaokai Yan, Jingtao Ding, Fengli Xu, and Yong Li. Fingertip 20k: A benchmark for proactive and personalized mobile LLM agents. In *International Conference on Learning Representations*, 2026a.

Saelyne Yang, Jaesang Yu, Yi-Hao Peng, Kevin Qinghong Lin, Jae Won Cho, Yale Song, and Juho Kim. GUIDE: A benchmark for understanding and assisting users in open-ended GUI tasks. *arXiv preprint arXiv:2603.25864*, 2026b.

Guanhua Zhang, Susanne Hindennach, Jan Leusmann, Felix Buehler, Benedict Steuerlein, Sven Mayer, Mihai Bace, and Andreas Bulling. Predicting next actions and latent intents during text formatting. In *CHI Workshop on Computational Approaches for Understanding, Generating, and Adapting User Interfaces*, 2022.

Zhong Zhang, Yaxi Lu, Yikun Fu, Yupeng Huo, Shenzhi Yang, Yesai Wu, Han Si, Xin Cong, Haotian Chen, Yankai Lin, et al. AgentCPM-GUI: Building mobile-use agents with reinforcement fine-tuning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 155–180, 2025.

Sha Zhao, Zhiling Luo, Ziwen Jiang, et al. Appusage2vec: Modeling smartphone app usage for prediction. In *2019 IEEE International Conference on Data Engineering*, 2019. doi: 10.1109/ICDE.2019.00120.

Yuyang Zhao, Wentao Shi, Fuli Feng, and Xiangnan He. AppAgent-Pro: A proactive GUI agent system for multidomain information integration and user assistance. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 6767–6771. ACM, 2025. doi: 10.1145/3746252.3761473.

Hanzhang Zhou, Xu Zhang, Panrong Tong, Jianan Zhang, Liangyu Chen, Quyu Kong, Chenglin Cai, Chen Liu, Yue Wang, Jingren Zhou, and Steven Hoi. MAI-UI technical report: Real-world centric foundation GUI agents. *arXiv preprint arXiv:2512.22047*, 2025.

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. WebArena: A realistic web environment for building autonomous agents. In *The Twelfth International Conference on Learning Representations*, 2024.

A Annotation and Reference Versions

The manuscript uses the component evidence annotations for the FingerTip 20K corpus and the corresponding AndroidControl annotations. Both the paper and released annotations use zero-based observation indices. Action $A_{i,k}$ first becomes available in the prefix ending at $O_{i,k+1}$, where i identifies the session and k is the action index. Predictions and annotations are joined using the same session and observation identifiers.

Candidate Selection and Date Cutoff. The fixed FingerTip candidate pool was constructed from 15,200 media- and annotation-eligible sessions in the 20,000-session working corpus. Excluding 5,047 eligible sessions used in earlier experiments left 10,153 candidates, of which 10,000 were retained by a reproducible fixed selection. Current component verification retains 7,761 sessions with an observation before t^*. The calendar restriction then removes 2,591 sessions recorded before April 1, 2025, leaving 5,170 sessions across all 83 users. These counts describe the candidate pool before action-cohort filtering. This is a retrospective date restriction of the fixed evaluation pool; it does not imply that historical prompt development used only pre-April interactions.

Reference intents and temporal labels are versioned separately. The evaluation checks reference equality and complete prediction and score coverage before aggregation. A change to the reference intent requires new semantic judgments; a change to the full support time requires recomputing temporal metrics.

Annotation audit protocol. Both annotation audits are complete. The timing audit covers 150 sessions stratified by trajectory length. Annotators mark each component's first supporting observation and then t^*; the stored annotation is withheld in a separate key file and never shown in the task or interface. Table 4 compares model annotations with the final human reference and annotator A with annotator B. Frame differences are absolute differences in observation index.

The semantic audit covers 300 (prefix, hypothesis) pairs from four model rows, balanced half accepted and half rejected by the judge. Both annotators score every pair with the judge's verdict hidden. Table 5 reports pairwise agreement over this balanced audit sample. Both samples use a fixed seed, and each audit pack records its sampling design.

Table 4: Full-support timing agreement on all 150 audited sessions. Within-one-frame agreement includes exact matches.

Metric	Model vs. final human reference	Annotator A vs. annotator B
Exact agreement	141/150 (94.00%)	146/150 (97.33%)
Within one frame	146/150 (97.33%)	148/150 (98.67%)
Mean absolute frame difference	0.093	0.047
Median absolute frame difference	0	0

Table 5: Semantic-judgement agreement on all 300 audited pairs.

Comparison	Agreements	Agreement rate
Annotator A vs. annotator B	298/300	99.33%
LLM judge vs. annotator A	297/300	99.00%
LLM judge vs. annotator B	297/300	99.00%

Evidence rubric. Annotation verifies the required application, operations, entity-role relations, and constraints against the observed prefix, then checks whether an earlier prefix already supplies the same support. Passive exposure to an item is distinguished from selecting it, and a required result receives a separate component only when the operation does not already express it. Missing evidence leaves a component unresolved. Each component timestamp retains its own supporting observations and annotation version.

B TEMPORAL ELIGIBILITY AND METRIC EXTENSIONS

The evidence-eligible candidate pool contains only sessions with a verified t^* after the initial observation. The source profiles also retain 174 FingerTip 20K and five AndroidControl sessions whose full support time is the initial observation. These sessions have no interval before t^* and can be evaluated separately for recognition at full support.

Table 6: Evidence-eligible candidate pools before action-cohort filtering. N is the number of observable prefixes. Valid t^* counts sessions with verified support for every required component; $t^* > 0$ additionally requires at least one earlier observation.

Source	Sessions	Prefixes	Med. N	P90 N	Valid t^*	$t^* > 0$	Med. $\frac{t^*}{N-1}$
FingerTip 20K	5,170	70,506	11	25	5,170	5,170	0.75
AndroidControl	1,096	7,010	6	10	1,096	1,096	1.00

Table 7: Selection of the evidence-eligible candidate pools before action-cohort filtering. Evidence eligibility and date restriction are applied at the session level.

Stage	FingerTip	AndroidControl
Frozen candidate pool	10,000	1,543
No valid component-consistent t^*	2,065	442
$t^* = 0$	174	5
Evidence-eligible sessions	7,761	1,096
Excluded by date restriction	2,591	–
Candidate sessions	5,170	1,096
$N = 1$–5	915	477
$N = 6$–10	1,363	519
$N = 11$–20	2,081	96
$N = 21$–50	707	4
$N = 51$–100	104	0

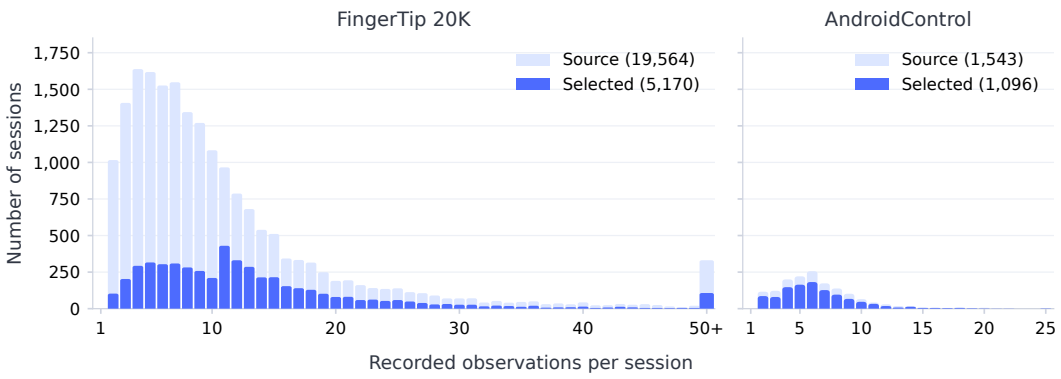

Figure 5: Session lengths for both sources, in recorded observations. Left: FingerTip 20K, where light blue shows all 19,564 source sessions with observable trajectories and dark blue overlays the 5,170 verified test sessions recorded on or after April 1, 2025. Right: AndroidControl, where light blue shows the 1,543 sessions of the official test partition and dark blue overlays the 1,096 that meet the evidence criteria. FingerTip reaches 92 observations but thins out past 50, so all sessions with 50 or more are pooled into the trailing 50+ bar; AndroidControl ends at 25 and keeps its own axis range. Both panels share the vertical scale.

Before action-cohort filtering, the selected FingerTip sessions average 13.64 observations, compared with 11.23 across observable source trajectories. AndroidControl averages 6.40 observations after eligibility filtering. The full support time is the final observation for 1,347/5,170 FingerTip sessions and 763/1,096 AndroidControl sessions. These sessions contribute to early-recognition evaluation but have no observation strictly after t^*.

Action cohort construction. The action cohort of Section 3.5 applies three thresholds to the evaluation set, in this order and with no reference to model output: at least 11 observations, at least four prefixes before t^*, and, for FingerTip, a recording date on or after April 11, 2025. The length and t^* thresholds are close to redundant on this corpus, since long sessions tend to have late full support times; retaining both excludes the case of a long session whose goal is supported at its second observation, which has no readable pre-t^* interval. From 5,170 FingerTip sessions the three thresholds retain 1,503 over 66 users, and we score every prefix of every retained session, 29,630 in total, 19.71 per session. Of the 17 users dropped, five have no session that meets the length and t^* thresholds at any date, seven have no session at all recorded after April 11, and the remaining five do have later sessions but none long enough to read against t^*. The retained sessions are longer than the pool they come from, median 16 observations against 11, and their pre-t^* interval is correspondingly longer, median nine against six, which follows from conditioning on both quantities at once; the cohort is therefore selected for readability rather than sampled at random from the evaluation set. The same thresholds on AndroidControl retain 94 of 1,096 sessions and 1,273 prefixes: its median length is 6 and its longest session has 25 observations, so a threshold calibrated to FingerTip is severe there, so AndroidControl forms a smaller candidate cohort. Its terminal prefixes carry no recorded next action, since AndroidControl marks the end of an episode by the absence of a transition rather than by a stop action, leaving 1,179 of the 1,273 prefixes scorable; FingerTip records an explicit completion, so its terminal prefixes are scored like any other. Within each retained session we score every prefix rather than a fixed offset grid, so the columns are composed of the same sessions throughout and the temporal metrics are read at full within-session resolution: the FingerTip cohort supplies 16,800 prefixes before t^* and 12,830 at or after it, 11.18 and 8.54 per session on average, of which 1,503 are terminal. Accessibility trees resolve a reference node for 93.6% of the cohort's FingerTip click references, which is the denominator of the *Node* measure; scorability is decided by the reference alone, so a condition that answers with a non-click is scored wrong at that prefix rather than dropped, and all three conditions share one denominator. The frozen cohort, the thresholds, and the excluded session identifiers with their reasons are released as a manifest so the filtering is reproducible and invertible.

Table 8: Endpoint scores on the same 29,630 prefixes as Table 2, kept out of the main table because they pool across t^* and are what Section 5.1 argues against relying on. Acc@End is rank-1 correctness at the last recorded observation and AUC@End integrates it over the whole interaction; both are session means in percent under the same Qwen3.8-27B judge. AnyEarly is repeated from Table 2 for the comparison the text makes: every row reaches a correct goal before full support more often than it holds one to the endpoint, by $1.21\times$ to $4.41\times$.

Model	Acc@End	AUC@End	AnyEarly
General-multimodal models			
Qwen3.5-0.8B	2.61	1.25	3.15
Qwen3.5-2B	6.97	4.70	16.11
Qwen3.5-4B	14.64	9.75	32.14
Qwen3.5-9B	14.64	9.87	33.60
GUI-specialized models			
GUI-Owl-1.5-2B	5.72	3.45	11.24
MAI-UI-2B	7.45	4.86	13.77
UI-Venus-1.5-2B	10.92	6.36	18.18
UI-TARS-1.5-7B	12.57	8.06	24.62
UI-Venus-1.5-8B	14.84	9.28	28.94
UI-Mate-9B	19.36	11.74	36.93
UI-Venus-2-9B	19.29	11.43	40.92
UI-Voyager	0.54	0.59	2.38

Per-type action arguments. Beyond the element-based conventions of Section 3.5, an action matches when its type equals the reference type and its arguments agree as follows: scrolls require the recorded direction, typed text a normalized edit similarity of at least 0.70 against the recorded string, application launches a matching package or display name, and the remaining discrete actions, including navigation and completion, require only the type.

Ranked-hypothesis and action-utility definitions. Over the set of scored prefixes $\mathcal{P}$, the ranked metrics of Section 3.6 are

$$\mathrm{Recall@}k = \mathrm{Avg}_{(i,t)\in\mathcal{P}}\left[\max_{1\le j\le \min(k,K_{i,t})} v_{i,t}^{(j)}\right], \qquad \mathrm{MRR@pre} = \mathrm{Avg}_{i\in\mathcal{I}}\left[\frac{1}{t_i^*}\sum_{t<t_i^*}\frac{1}{\rho_{i,t}}\right], \quad (10)$$

where $v_{i,t}^{(j)}$ is the judge's independent verdict on the rank-j candidate, $K_{i,t}$ is the realized candidate count after deduplication, and $\rho_{i,t}$ is the credited rank, the lowest-numbered candidate whose verdict is one. A prefix with no matching candidate contributes zero. Recall@k is a prefix mean over $\mathcal{P}$; MRR@pre is a session mean.

For each action condition $c \in \{\mathrm{none}, \mathrm{gold}, \mathrm{pred}\}$ of Section 3.5, let $\mathrm{ActAcc}_{i,t}^{c} \in \{0,1\}$ record whether the predicted action matches the recorded reference action, and let $\mathcal{Q}_i(\mathcal{T})$ be the prefixes of session i in interval $\mathcal{T}$ at which all three conditions have a scored prediction. Then

$$\mathrm{Act}^{c}(\mathcal{T}) = \mathrm{Avg}_{i\in\mathcal{I}}\left[\frac{1}{|\mathcal{Q}_i(\mathcal{T})|}\sum_{t\in\mathcal{Q}_i(\mathcal{T})}\mathrm{ActAcc}_{i,t}^{c}\right], \quad (11)$$

$$\mathrm{Realized}(\mathcal{T}) = \frac{\Delta_{\mathrm{real}}(\mathcal{T})}{\Delta_{\mathrm{head}}(\mathcal{T})}, \quad (12)$$

with Δ_{head} and Δ_{real} as in Equation 9. Node accuracy restricts $\mathcal{Q}_i(\mathcal{T})$ to the node-scorable prefixes, which the reference alone selects, so its denominator is likewise common to the three conditions. Counting instead only the prefixes at which the condition itself emitted a click makes the denominator a function of the behavior being measured, since supplying a goal changes how often a predictor answers with a click at all: that convention scores the goal-supplied conditions on a smaller and easier subset and manufactures a gain for them.

Observed input budget. Each query at t includes observations 0 through t and the t completed actions. The shared input image encoder preserves aspect ratio, caps the longer side at 768 pixels, and encodes JPEG at quality 85. The temporal selection retains complete sessions, including every prefix of each selected session. Reaggregation changes the session membership without changing the original predictor inputs or reference intents.

Component support is accumulated over the observed prefix. A trajectory that contradicts a required component cannot be assigned full support simply by taking the latest of its components' first-support timestamps. Such conflicts must be resolved under the annotation rubric before the session is eligible.

B.1 Paired Action Analysis Before Full Support

Table 3 compares *None*, *Gold*, and *Pred* on the 1,503-session FingerTip action cohort. Of its 16,800 pre-t^* prefixes, each model retains 16,598–16,800 with parseable actions in all three conditions (Equation 11). Conditions are paired within a model, while coverage can differ across models. Success is averaged within each session and then equally across sessions. The difference $\delta = \mathrm{Pred} - \mathrm{Gold}$ is computed before rounding; subtracting displayed percentages can differ by 0.01 percentage points.

No pre-t^* prefix has completion as its reference action, so a predicted completion remains an error in the paired denominator. This restriction removes gains from correct terminal stopping while retaining premature-stopping and action-argument errors. Parsing exclusions make the comparison conditional on parseable outputs.

The significance markers compare *Pred* with *Gold* using McNemar tests on paired prefix-level success indicators. These tests do not model within-session dependence and differ from the session-bootstrap intervals of the primary evaluation. Seven of nine negative differences are marked, as are the positive differences for Qwen3.5-2B and UI-TARS-1.5-7B.

Full-action success checks action types and arguments; node accuracy evaluates reference-eligible element selections. Under inferred goals, Qwen3.5-4B exceeds 9B in pooled full-action success (17.84% versus 17.62%) but trails it in pre-t^* node accuracy (16.19% versus 19.25%; Table 2). Table 3 instead restricts full-action success to $t < t^*$. Here, *Gold–None* measures the benefit of a reference goal, *Pred–None* the benefit realized with an inferred goal, and δ their difference.

C Scope and Validation

The benchmark evaluates predictions of a recorded session intent. It does not measure GUI execution success, user acceptance, or on-device latency. The action arm of Section 3.5 measures the utility of a supplied goal for predicting the recorded next action, and nothing beyond that: a predicted action is scored against what the user did at that prefix, so an alternative action that would also have served the goal counts as an error, and this noise is larger before t^*, where more than one next step is reasonable, which pushes the pre-t^* absolute values down. A correct action at one prefix also does not establish that a model could carry the remainder of the session, so the arm speaks to the value of a goal at a step, not to how a task would unfold. The action cohort is filtered by session length, the size of the pre-t^* interval, and recording date, which retains 66 of the 83 FingerTip users and removes sessions whose goal is fully supported almost immediately; the filters use no model output, and Appendix B states how the retained sessions differ from the pool. A user's intent may also evolve during browsing; evaluating every prefix against a fixed recorded intent does not model such changes. Before full support, prediction errors may reflect insufficient information as well as model limitations.

The full support time depends on the annotation rubric, reference decomposition, and interpretation of observed actions. Component verification asks when the record supports an intent requirement; a blind prefix study would ask when a person can infer the hidden intent. These are separate validation questions. The primary confidence intervals resample sessions and do not account for all dependence among sessions from the same user.

Mobile screenshots and intent entities may contain private information. Source media remain subject to the datasets' access and redistribution conditions; the paper repository contains aggregate evidence and manuscript assets.