Carnegie Mellon University
School of Computer Science
DUKE COMPUTER SCIENCE

Modeling Distinct Human Interaction in Web Agents

Faria Huq[C,†] **Zora Zhiruo Wang**[C,†] **Zhanqiu Guo**[C,‡] **Venu Arvind Arangarajan**[C,‡]
Tianyue Ou[C] **Frank Xu**[C] **Shuyan Zhou**[D] **Graham Neubig**[C] **Jeffrey P. Bigham**[C]
[C]Carnegie Mellon University [D]Duke University
[†] Co-first Authors [‡] These authors contributed equally
{fhuq, zhiruow}@cs.cmu.edu

Models:huggingface.co/CowCorpus Code:github.com/oaishi/PlowPilot

Abstract

Despite rapid progress in autonomous web agents, human involvement remains essential for shaping preferences and correcting agent behavior as tasks unfold. However, current agentic systems lack a principled understanding of *when* and *why* humans intervene, often proceeding autonomously past critical decision points or requesting unnecessary confirmation. In this work, we introduce the task of *modeling human intervention* to support collaborative web task execution. We collect CowCorpus, a dataset of 400 real-user web navigation trajectories containing over 4,200 interleaved human and agent actions. We identify four distinct patterns of user interaction with agents – **hands-off** supervision, **hands-on** oversight, **collaborative** task-solving, and full user **takeover**. Leveraging these insights, we train language models (LMs) to anticipate when users are likely to intervene based on their interaction styles, yielding a 61.4–63.4% improvement in intervention prediction accuracy over base LMs. Finally, we deploy these intervention-aware models in live web navigation agents and evaluate them in a user study, finding a 36.8% increase in user-rated agent usefulness. Together, our results show structured modeling of human intervention leads to more adaptive, collaborative agents.

1. Introduction

Recent advances in large language models (LLMs) have enabled AI agents to perform increasingly complex tasks in web navigation (Deng et al., 2024; Shi et al., 2017; Yao et al., 2022; Zhou et al., 2023). Despite this progress, effective use of such agents continues to rely on human involvement to correct misinterpretations or realign behavior with user preferences (Amershi et al., 2014; Misra et al., 2017; Saunders et al., 2017). However, current agentic systems lack an understanding of when and why humans intervene. As a result, agents may pursue autonomy under incorrect assumptions about user intent, overlooking critical factors as tasks unfold (Hadfield-Menell et al., 2016; Mitchell et al., 2025). Even when agents proactively stop to check with users, they often do so at inappropriate moments or interrupt too frequently with unnecessary confirmation requests (Chen et al., 2024; Huq et al., 2025), forcing users to step in mid-execution and incurring a heavy oversight burden (Bansal et al., 2024; Wang et al., 2020, 2025). Modeling when humans are likely to intervene can help agents anticipate preventable mistakes, minimize unnecessary disruptions, and reduce the oversight burden without sacrificing

reliability.

To build more effective collaborative agents, autonomy should complement human involvement rather than override it, and engage users only when their input is necessary. Although recent work has explored proactive assistance in collaborative agent (Feng et al., 2024; Huq et al., 2025; Ramrakhya et al., 2025; Shao et al., 2025), these approaches typically focus on specific interaction mechanisms, including asking follow-up questions or co-planning with users, rather than modeling the broader spectrum of human interaction patterns that arise during execution, such as mid-task intervention, alternative action-taking, and transfer of control.

This motivates a central question: *Can agents proactively anticipate when human intervention is likely and adapt their behavior accordingly?* To answer this, we introduce CowCorpus: a real-user corpus of 400 human-agent collaborative task execution trajectories on the open web. The dataset comprises over 2,748 agent action steps and 1,476 human action steps, with step-level annotations marking when users intervened by pausing, resuming, or overriding agent execution. Analyzing CowCorpus reveals that human involvement is driven by three recurring needs: error correction, preference refinement, and assistive takeover. Although these motivations appear in individual actions, they combine into consistent higher-level collaboration strategies over the course of a task. Accordingly, we group users into four distinct collaboration styles: **Takeover**, **Hands-on**, **Hands-off**, and **Collaborative**; which capture how users balance supervision, intervention, and share control with the agent (§3).

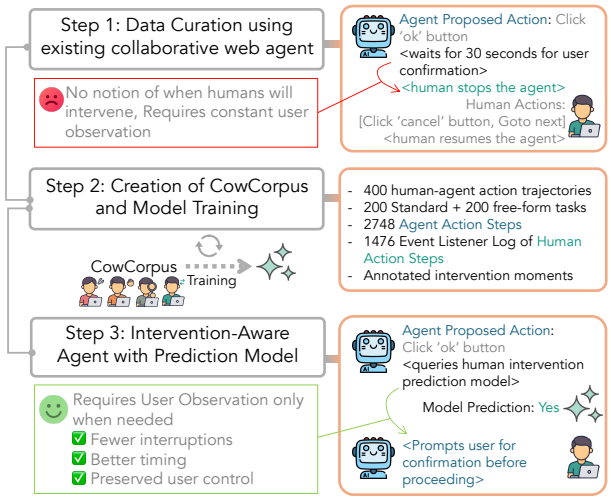


Figure 1: In this paper, we present CowCorpus, a dataset of 400 real-user collaborative web trajectories that captures when and how humans intervene during execution, enabling intervention-aware agents that engage users only when needed.

Building on this empirical characterization, we formulate the task of *modeling human intervention patterns* conditioned on user collaboration style. We cast this problem as a stepwise sequence prediction task: at each agent action, the model estimates the likelihood of user intervention given the evolving task context. We train LMs in two settings: (i) a general intervention-aware model that moves beyond purely autonomous execution, and (ii) style-conditioned models adapted to specific collaboration styles. Across multiple model backbones, our intervention-aware models improve intervention prediction accuracy by 61.4–63.4% over baselines (§4).

Beyond offline prediction accuracy, we integrate our intervention-aware models into live web agents and evaluate them through a user study on real-world web tasks. Agents equipped with intervention modeling achieve a 36.8% increase in user-perceived usefulness over baseline systems, demonstrating that anticipating human intervention leads to more adaptive and effective human–agent collaboration in practice (§5).

More broadly, this work suggests a shift from optimizing agent autonomy to designing agents that dynamically adapt to human preferences and collaboration styles over time.

2. Problem Formulation: Human Intervention Modeling

We formulate the human-agent collaboration as a Partially Observable Markov Decision Process (POMDP). Given a task instruction q, the two actors, $z \in \{agent, human\}$ following policies π_{agent} and π_{human}, attempt to complete it by taking a sequence of discrete actions $\tau = [a_1, a_2, \ldots, a_T]$, where each action a_t at time t is taken either by the agent or the human. At each time step t, we construct a multimodal observation $o_t = (V_t, A_t)$ of the current state, consisting of the current screenshot V_t and the webpage accessibility tree A_t. This observation is passed into the actor policy to generate the next action $a_t = \pi(o_t|\tau_{0:t-1})$, where $\tau_{t-1} = [(o_1, a_1), \ldots, (o_{t-1}, a_{t-1})]$ denotes the past trajectory. By default, the agent generates the proposed action $\hat{a}_t = \pi_{agent}(o_{0:t}, a_{0:t-1})$.

At any time step, the human can choose to intervene, formalized as a binary intervention variable $y_t \in \{0, 1\}$. We define this *human intervention modeling* task to be a step-wise binary classification, where the objective is to learn a predictive model f_θ that estimates:

$$p(y_t = 1 \mid o_t, \hat{a}_t, \tau_{t-1}). \tag{1}$$

We approach this using a large multimodal model (LMM) optimized via supervised fine-tuning (SFT). The model takes a serialized prompt containing (1) the history trajectory τ_{t-1}, (2) the current observation o_t, and (3) a description of the agent-proposed action $\hat{a}_t$. The model is fine-tuned to generate designated tokens: `<ask_user>` or `<agent_continue>`, indicating the intervention decision to intervene or proceed.

Evaluation Metrics To measure the effectiveness of human intervention modeling, we measure the step accuracy, F1 score, and Perfect Timing Score (PTS) across all trajectory steps and report the average performance on the test split. *Step Accuracy* measures the fraction of steps where the model correctly predicts whether a human intervenes, while *F1 Score* measures the harmonic mean of precision and recall for intervention prediction.

Perfect Timing Score (PTS): evaluates how accurately a model predicts the timing of human intervention, which is computed with respect to each ground-truth intervention event: $\text{PTS} = \frac{1}{Z} \cdot \sigma\left(\mathbb{I}_{\text{correct}} - \sum_{i \in E} \alpha \cdot d_i^2\right)$.

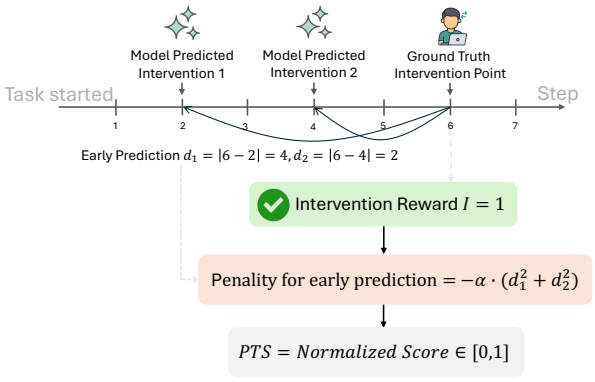


Figure 2: Visual Illustration of how PTS is calculated. We measure the L_2 squared distance between the ground truth intervention and false-positive predictions. The score then penalizes based on the following distance.

$\mathbb{I}_{\text{correct}}$ indicates whether the model predicts intervention exactly at the ground-truth intervention step $t_{\text{intervene}}$, while E denotes false-positive intervention predictions made before $t_{\text{intervene}}$. The term $d_i = |i - t_{\text{intervene}}|$ penalizes wrong predictions based on their temporal distance from the true intervention step, with α controlling the penalty strength.[1] The score is normalized to $[0, 1]$ using sigmoid $\sigma(\cdot)$ and a factor $Z = \sigma(1)$, where higher values indicate more accurate and well-timed intervention predictions (Figure 2). For trajectories with multiple

[1]We set $\alpha = 0.2$ by default. We test how sensitive PTS is to α in §B.5 and find it presents consistent measures across a wide range of alpha values (0.1 – 0.5).

Domain	Subdomain	Website 🌐	Task Prompt 🗣️
Travel	Airlines	united.com	*Find a round trip from Phoenix to Miami with maximum budget of $2000.*
	Restaurant	yelp.com	*Find parking in California city for Limos which also offers military discounts and free wi-fi.*
Info	Housing	student.com	*Find a property in London with Bike Storage and Gym facilities with lowest price.*
	Job	indeed.com	*Search for nutritionist jobs in Ohio.*
Service	Health	babycenter.com	*Show me the popularity in 2015 of the current most popular baby girl name.*
	Government	dmv.virginia.gov	*Find information on how to request a Police Crash Report.*
Shopping	Speciality	gamestop.com	*Check the trade-in value for Call of Duty: Black Ops III for Xbox One.*
	Auto	carmax.com	*Search for used BMW X5 Crossovers and compare the mileage of the first two cars.*
Entertainment	Event	ticketcenter.com	*Show MLB tickets for this weekend and select the next one.*
	Music	last.fm	*Play the top track for the top indie artist in the last 30 days.*

Table 1: Standard tasks selected from Mind2Web (Deng et al., 2024).

interventions, the final trajectory-level PTS is the average over all intervention-event-level PTS scores. While step accuracy measures the precise correctness, PTS models the temporal context when measuring the performance.

3. CowCorpus: Collecting Human-Agent Collaborative Web Activities

We introduce CowCorpus, a dataset for studying human intervention patterns in collaborative web workflows. In this section, we first describe the data collection process (§3.1), then analyze the motivation behind user intervention (§3.2) and summarize user interaction patterns (§3.3).

3.1. Data Collection

To ensure CowCorpus is consistent with established benchmarks and reflects individual user preferences, we designate a mixture of free-form tasks and benchmark tasks in our dataset — (1) 10 standard tasks from the Mind2Web dataset (Deng et al., 2024) and (2) 10 free-form tasks of the participants' own choice. We recruited 20 human users to complete these 20 web tasks in collaboration with an AI agent, specifically, the open-source framework CowPilot (Huq et al., 2025). For all collected trajectory steps, we record the action, the actor (human or agent), along with timestamps and web snapshots. For more information on the agent framework used in task annotation, please refer to §A.2.

Standard Tasks To isolate differences in human interaction styles, we analyze how different users behave when performing the same agent-mediated standardized tasks. We select 10 tasks from Mind2Web (Deng et al., 2024) by first identifying the most frequently visited websites within the top two sub-domains, and then randomly sampling one task per website. Table 1 summarizes the selected sub-domains, websites, and tasks. This selection procedure

Task Type	Example	Percentage
Information Access	*search the latest model of Honda SUV car and show me its price (P4)*	57.98%
Personalized Interests & Lifestyle	*search daredevil: born again and play the first episode (P5)*	7.98%
Product & Service Interaction	*buy 1 sweater for me from the first shop of the list (P1)*	4.79%
Content Generation	*Help me draft a post on this platform in the search query about why AI models are not good for the environment (P17)*	3.19%
Multi-Agent Collaboration	*Use grok thinking model to answer this question: Find the sum of all integer bases $b > 9$ for which 17_b is a divisor of 97_b. (P13)*	2.66%
Security Testing	*Send a mail to xxx@gmail.com about how frustrated you are with their performance (P6)*	2.66%
Planning & Scheduling	*Create a meeting event on Monday 2-3pm (March. 17th) (P15)*	2.13%
Reasoning & Meta-Analysis	*Help me compare prices to travel to Dhaka from JFK on December 10th for one way (P6)*	2.128%
Misc.	*Create an issue in this repo to say thanks for this great work (P13)*	16.49%

Table 2: Examples of free-form tasks across nine categories, with task description and distribution percentages.

preserves alignment with the original benchmark's task distribution while enabling controlled comparisons of user interaction patterns.

Free-form Tasks To complement standardized benchmark tasks, we ask participants to conduct 10 free-form tasks of their choice on arbitrary websites. This open-ended setting captures the types of tasks users naturally attempt to automate and the instruction styles and levels of specificity they use in real-world interactions. By moving beyond predefined benchmarks, this design provides insight into how humans collaborate with agents under unconstrained, user-driven objectives. Table 2 illustrates the overall distribution of user-issued tasks. §A.3.1 provides a more detailed description of how this distribution is curated.

With the human-agent collaborative trajectories we collect on both sets of tasks, we evaluate the number of steps taken by human and agent actors during the task-solving sessions, as well as the time taken to solve the tasks. We report the dataset statistics in Table 3.

Task Category	Intervention Intensity	Step Count			Time (seconds)		
		Agent	Human	Total	Agent	Human	Total
Standard	21.63%	7.1	1.6	8.7	93.1	23.9	117.0
Free-form	16.06%	6.1	0.9	7.0	71.7	13.8	85.5

Table 3: CowCorpus statistics for standard and free-form tasks: (1) intervention intensity: percentage of human actions across all trajectories, (2) step count: number of steps taken by agent or human actors, (3) time: time taken by agent or human actors.

3.2. Step-Level User Intervention

To understand when and why users intervene during collaborative task execution, we analyzed post-task annotations and open-ended responses from all participants.

3.2.1. When Do Users Intervene?

To quantitatively measure when users intervene, we extract four per-user features to capture how often users intervene, how much they intervene, when interventions occur, and whether

control is returned to the agent, providing a compact characterization of when users intervene.

For a user u, let $\mathcal{D}_u$ denote the set of trajectories involving that user, where $\tau = (a_1, ..., a_T)$ is the action trajectory a_t with length $T = |\tau|$. For each trajectory $\tau \in \mathcal{D}_u$, let τ_{agent} and τ_{human} denote the subsequences of agent and human actions, with lengths $|\tau_{\text{agent}}|$ and $|\tau_{\text{human}}|$.

We define an intervention event e as a contiguous interval of time steps $[t_s, t_e]$ where the human is in control (i.e., taking actions). Let $\mathcal{E}(\tau)$ be the set of all such events in τ, and let $I(\tau) = |\mathcal{E}(\tau)|$ denote the number of intervention events in τ.

Intervention Frequency measures how often a user intervenes over the total number of actions:

$$\text{frequency}(u) \;=\; \frac{\sum_{\tau \in \mathcal{D}_u} I(\tau)}{\sum_{\tau \in \mathcal{D}_u} |\tau|}. \tag{2}$$

Intervention Intensity measures the ratio of total human steps to total agent steps:

$$\text{intensity}(u) \;=\; \frac{\sum_{\tau \in \mathcal{D}_u} |\tau_{\text{human}}|}{\sum_{\tau \in \mathcal{D}_u} |\tau_{\text{agent}}|}. \tag{3}$$

Normalized Intervention Position To characterize when interventions occur within a trajectory, let $H(\tau) = \{t \mid a_t \in \tau_{human}\}$ be the indices for human actions. We compute the mean normalized position of all human action steps:

$$\text{pos}(u) \;=\; \frac{1}{N_u} \sum_{\tau \in \mathcal{D}_u} \sum_{t \in H(\tau)} \frac{t}{|\tau|}, \quad N_u = \sum_{\tau \in \mathcal{D}_u} |H(\tau)|. \tag{4}$$

Handback Rate. We measure whether control returns to the agent after a human intervention. For each event $e = [t_s, t_e] \in \mathcal{E}(\tau)$, define an indicator $b_e = 1$ if $t_e < |\tau|$, where the agent takes at least one action after the human intervention ends, and $b_e = 0$ otherwise. The handback rate is:

$$\text{handback}(u) \;=\; \frac{1}{M_u} \sum_{\tau \in \mathcal{D}_u} \sum_{e \in \mathcal{E}(\tau)} b_e, \quad M_u = \sum_{\tau \in \mathcal{D}_u} |\mathcal{E}(\tau)|. \tag{5}$$

3.2.2. Why Do Users Intervene?

Error Correction and Recovery Participants frequently intervene to correct agent mistakes or redirect execution when the agent becomes stuck. Two common scenarios emerge. *(1) Incorrect or premature actions:* The agent selects the wrong element or executes an action before necessary prerequisites are met. For example, before completing a product search, the agent prematurely selected a location filter from a drop-down menu. *(2) Agent stuck or looping:* When the agent repeatedly performs invalid or redundant actions, users step in to break the loop and carry out a few corrective steps to move the task forward.

Preference Misalignment Users also intervene when agent actions diverge from their intended preferences, often due to incomplete or underspecified instructions. *(1) Unmet prerequisite:* Agents sometimes ignore or overlook key requirements specified by the user, such as price (shoes "under $100") or location when searching for information (e.g., weather "in Pittsburgh"). *(2) Ambiguity in task description:* In other cases, initial task descriptions lack sufficient detail, leaving room for interpretation. Participants noted that they often did not fully specify preferences, such as the brand when asking the agent to buy toothpaste, until they observed the agent's intermediate actions against their preference, prompting mid-task clarification.

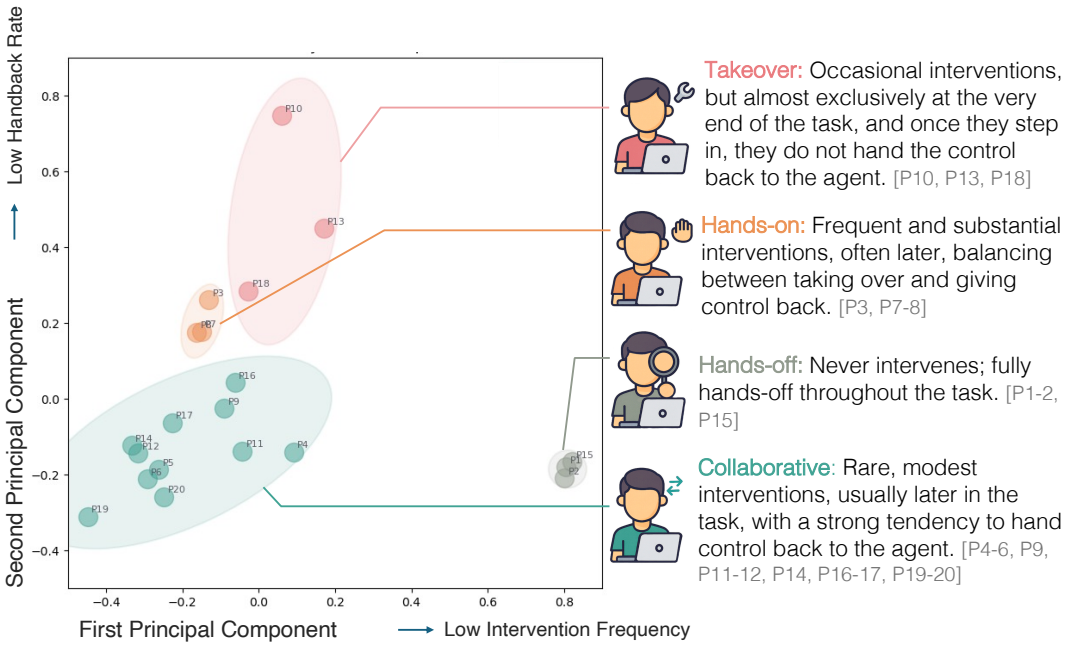


Figure 3: **Four distinct types of human-agent interaction patterns**: Takeover, Hands-on, Hands-off, and Collaborative. We visualize the user groups using PCA (left), and describe the interaction mechanism of each group (right).

Assistive Intervention for Complex Environments Users sometimes intervene not to correct explicit errors, but to compensate for limitations in the agent's ability to operate reliably within complex web environments. *(1) Complex UI elements:* Agents struggled with dropdowns, captchas, dynamic layouts, or complex DOM elements in certain websites. *(2) Missing resources:* Agents occasionally failed to load required links or components of the page. *(3) Manual takeover for control:* Users may preemptively stop agent execution to avoid mistakes (especially unrecoverable ones), particularly in tasks they have had to repeat due to prior errors.

3.3. Task-Level Interaction Patterns

We analyze *when* human interventions occur during collaborative task execution and how such temporal patterns vary across users. We summarize each participant's intervention behavior with four distinct collaborative features derived from action logs.

Using the four participant-level measures in §3.2.1, we cluster users by interaction behavior with k-means (k=4). We then project these features into two dimensions using PCA for visualization (Figure 3). The resulting structure is largely explained by two axes corresponding to decreasing intervention frequency and decreasing handback rates. This analysis reveals four distinct and stable groups of users with qualitatively different patterns of intervention timing and control sharing. Based on cluster centroids and representative trajectories, we characterize the four groups as follows:

• Takeover: Users intervene infrequently and typically late in the task. When they do step in, they tend to retain control rather than returning it to the agent, resulting in low handback rates. These interventions often coincide with completing the task themselves rather than correcting the agent mid-execution.

• **Hands-on**: Users intervene frequently and with high intensity. Their interventions tend to occur relatively late in the trajectory, but unlike Takeover users, they regularly alternate control with the agent, leading to medium handback rates and sustained joint execution.

• **Hands-off**: Users rarely intervene throughout the task. They exhibit low intervention frequency and intensity, allowing the agent to execute most trajectories end-to-end with minimal human involvement.

• **Collaborative**: Users intervene selectively and consistently return control to the agent. This group is characterized by high handback rates and earlier intervention positions, reflecting targeted, short-lived interventions that support ongoing collaboration.

Overall, users exhibit systematic differences in *when* interventions occur, how much they intervene, and whether control is relinquished afterward. Such temporal intervention patterns are consistent across tasks and motivate modeling distinct human–agent interaction patterns.

4. Experiments: Modeling Human Intervention

In this section, we train language models (LMs) to model human intervention patterns in collaborative web navigation. We study a progression from fully autonomous operation to two levels of collaboration: (1) a general intervention-aware model that captures common user behaviors (§4.2), and (2) style-conditioned models that tailor interaction to different user collaboration preferences (§4.3).

4.1. Experiment Setup

Setup We split CowCorpus data into train and test sets at the trajectory level to avoid leakage. We keep the intervention steps ratio consistent across train and test splits (approximately 1:7 for intervention and non-intervention steps). We exclude the **Hands-off** cluster from the train and test set as it contains no intervention events, making the prediction task irrelevant for this particular cluster. The processed dataset contains 1,247 training steps and 251 test steps. Each step is represented as a multimodal input consisting of the prior interaction history and current web snapshot (accessibility tree and screenshot).

Method We train (1) a general intervention-aware model using all training data and (2) style-conditioned models tailored to each interaction group using the corresponding subset of trajectories. To evaluate effectiveness, we compare these models against both prompting-based proprietary LMs and fine-tuned open-weight models on the Human Intervention Prediction task, using the metrics defined in §2.

To contextualize performance and assess the value of modeling interaction, we also include two non-learning baselines: (1) *Always No Interv*, a fully autonomous policy that never requests user intervention, and (2) *Always Interv*, a fully confirmation-dependent policy that requests intervention at every step.

4.2. Benchmarking Intervention Awareness in Autonomous Agents

Proprietary Models remain overly conservative: We evaluate three families of closed-source LMs (`Claude 4 Sonnet` (Anthropic, 2025), `GPT-4o` (Hurst et al., 2024), and `Gemini 2.5 Pro` (Comanici et al., 2025)) using zero-shot without reasoning. Although these models possess strong general knowledge, they struggle with the temporal dynamics necessary for accurate human

Model	Step Accuracy	F1 Score		PTS
		Intervention	Non-Intervention	
Baselines				
Always Interv	0.147	0.257	0.000	0.151
Always No Interv	0.853	0.000	0.920	0.000
Closed Source				
`Claude 4 Sonnet`	0.681	0.231	0.799	**0.293**
`GPT-4o`	**0.741**	0.198	0.846	0.147
`Gemini 2.5 Pro`	0.681	0.286	0.795	0.262
Open Source				
`Gemma 27B`	**0.239**	0.264	0.214	**0.187**
`Llava 8B`	0.183	0.000	0.343	0.017
Ours				
`Gemma 27B` (SFT)	**0.853**	0.302	0.918	**0.303**
`Llava 8B` (SFT)	0.817	0.296	0.897	0.201

Table 4: Model performance on predicting human intervention. We report F1 scores separately for intervention and non-intervention steps to account for class imbalance. See A.3.3 for more results with few-shot and reasoning enabled in models.

intervention prediction. Notably, `GPT-4o` achieves high performance on non-intervention steps (Non-interv F1: 0.846), but it fails on active interventions (Interv F1: 0.198). The drastic F1 disparity indicates that generalist models are overly conservative and struggle to balance the dynamic with the need for proactive assistance. This results in a low PTS (0.147). Additional results with two-shot prompting and with reasoning are reported in §A.3.3.

Fine-tuned Open-weight Models with Specialized Data Beats Scale: In contrast, fine-tuning open-weight models on COWCORPUS yields the most significant performance gains, surpassing proprietary models. Our fine-tuned `Gemma-27B` (SFT) achieves the state-of-the-art PTS (0.303), outperforming `Claude 4 Sonnet` (0.293), while the smaller `LLaVA-8B` (SFT) achieves a competitive PTS (0.201), beating `GPT-4o` (0.147). These results demonstrate that fine-tuning on high-quality interaction traces effectively bridges the alignment gap, allowing smaller models to master the nuance of intervention timing where generalized giant models fail.

Importance of Proper Interaction While the *Always No Interv* baseline achieves a high overall step accuracy (85.3%) due to class imbalance, it yields a PTS of 0, failing to identify any intervention. Conversely, the *Always Interv* baseline captures all interventions but suffers from a low PTS (0.151) due to heavy penalties for mistimed interruptions. These two extreme cases underscore that successful modeling requires temporal localization, not just binary classification.

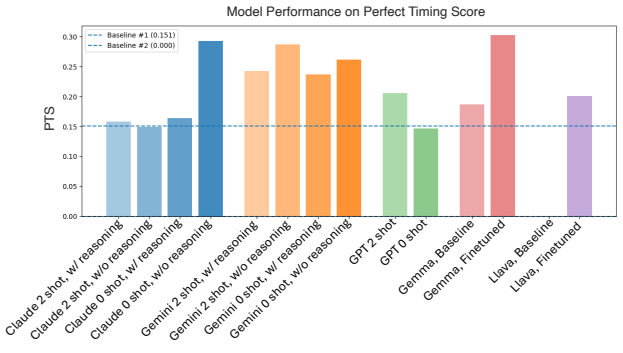


Figure 4: Perfect Timing Score on COWCORPUS. Out of the proprietary models, Claude outperforms GPT-4o and Gemini-2.5. On the finetuned model, Gemma 27B significantly boosts the performance when finetuned on COWCORPUS.

4.3. Interaction Pattern Customization

Beyond modeling generalized human interaction patterns, we also explore how to

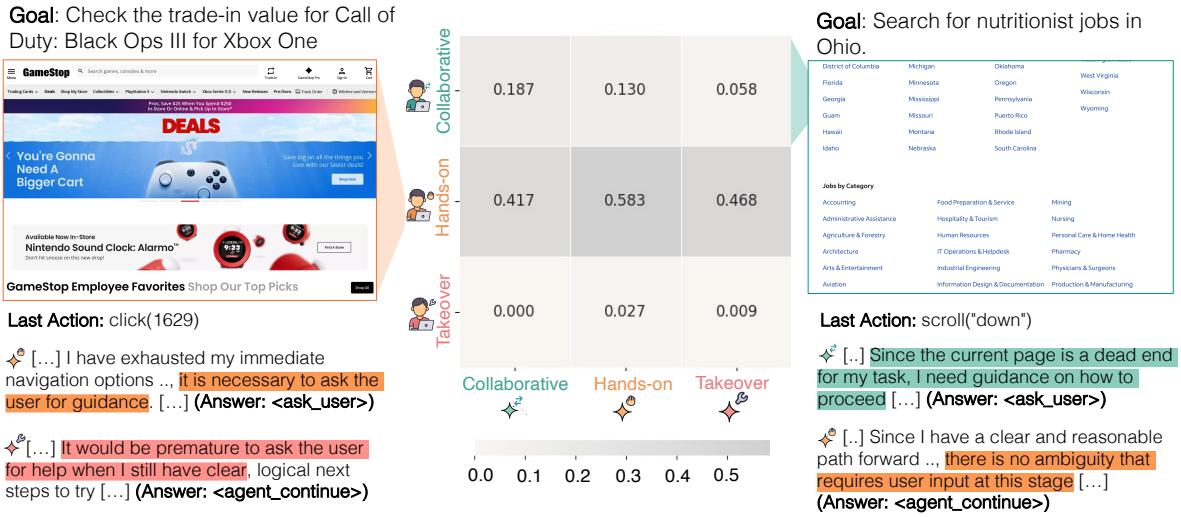


Figure 5: The heatmap shows the PTS score on the cluster-wise trained models for each of the three clusters. Models trained for corresponding clusters generally outperform the others, with the only exception of the Takeover group, which is analyzed in §4.3

adapt predictions to the four distinct user interaction patterns. Concretely, we adapt the models to user groups by further fine-tuning the `LLaVA-8B-Next` model from §4.2 with a reduced learning rate. This allows us to move from generally cooperative behavior to user-adaptive collaboration that aligns with individual interaction preferences. We finetune the base `LLaVA-8B-Next` model (§4.2) three times for each of the three clusters: **Takeover**, **Hands-on**, and **Collaborative**). We did not train further for **Hands-off** since the user never intervenes in this cluster, making the prediction irrelevant in this case.

We evaluated the performance of these specialized models against their corresponding validation sets. As shown in Figure 5, diagonal dominance indicates that models trained on specific clusters generally outperform the rest in the corresponding cluster. The only exception is for **Takeover** group, where the **Hands-on** model yields the best performance. This behavior can be explained by data sparsity: the **Takeover** cluster contains only 11 intervention steps (out of 131 total), compared to 37 intervention steps (out of 296) for **Hands-on**, limiting the strength of supervision available for the Takeover-specific model.

Specifically, these results suggest a dual strategy for personalized agents: while distinct interaction styles require specialized models to avoid misalignment, users with sparse feedback could benefit from models trained on high intervention frequency user group, which reveals error boundaries more clearly.

5. Deploying Collaborative Web Agents

To evaluate whether improved intervention modeling translates to real-world impact in human–agent collaboration, we integrate our intervention-aware model into a web navigation agent and deploy it as a Chrome extension, PLOWPILOT (Allen et al., 2007). Concretely, rather than confirming with users and allowing them to intervene at any step, PLOWPILOT now prompts for intervention only at moments where the model predicts a high likelihood of user intervention.

To evaluate this method in practice, we invited our original 20 annotators to participate in a

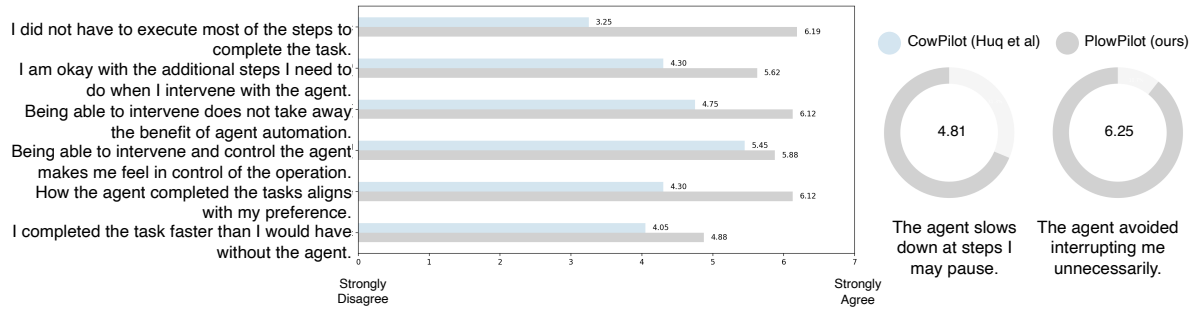


Figure 6: User response to the Likert scale questionnaire after the study. On average, user reports 36.8% higher in user rating compared to existing collaborative agents (Huq et al., 2025).

second round of sessions. Four participants responded and completed the same experimental protocol as before, consisting of 10 standard tasks and 10 free-form tasks. We assigned the customized interactive prediction model from §4.3 based on the cluster they belonged to. We compare their post-session ratings for the intervention-aware agent against their earlier ratings for the baseline agent to assess changes in user satisfaction.

Participants rated their experience on a 7-point Likert scale across six dimensions: (1) the extent to which they had to execute most task steps themselves; (2) whether the effort required during interventions was justified; (3) whether the ability to intervene diminished the benefits of automation; (4) whether intervention capabilities increased their sense of control; (5) whether the agent's behavior aligned with their preferences; and (6) whether they completed tasks faster than they would have without the agent.

Since the followup sample is small, we additionally recruited 12 participants who had not previously interacted with either system and ran them through the same protocol and questionnaire. A two-sided Mann-Whitney U test found no significant difference between the new participants and the four returning annotators across all six dimensions ($p > 0.05$ for four dimensions; $p > 0.01$ for the remaining two). We therefore report the PLOWPILOT results jointly over the combined sample of n=16 participants in Figure 6. Across all six measures, preliminary results from our user study show that PLOWPILOT outperforms the existing collaborative web agent (Huq et al., 2025), with an average improvement of 36.8% (Figure 6). Importantly, the underlying execution agent remains unchanged from COWPILOT; PLOWPILOT differs only by the addition of the intervention-aware module. The observed gains therefore arise solely from proactively modeling human intervention. These findings provide initial evidence that anticipating user intervention can substantially improve the effectiveness and usability of collaborative agent systems in practice.

6. Related Work

Autonomous Web Agents Web automation has been transformed by LLM-based agents capable of navigating complex environments. Benchmarks such as Mind2Web (Deng et al., 2024) and WebArena (Zhou et al., 2023) have pushed agents toward real-world, multi-domain tasks that use HTML and accessibility tree representations. The emergence of recent Computer Use capabilities from models like Claude (Anthropic, 2024) and Operator have further closed the gap between human browsing and machine execution as demonstrated by plugin-based web agents tools like WebCanvas (Pan et al., 2024), WebOlympus (Zheng et al., 2024), OpenWebAgent (Iong et al., 2024), and Taxy (TaxyAI, 2024) that can integrate into natural browsing contexts.

However, these extensions often prioritize autonomy over collaboration, lacking mechanisms for interactive control by users. Our work builds upon this plugin-based paradigm and emphasizes human-agent collaborations beyond solo agent autonomy.

Modeling Human-Agent Collaboration Human-AI collaboration has been widely studied in various settings, ranging from robotics (Ajoudani et al., 2018; Chandrasekaran and Conrad, 2015) and productivity tools (Khadpe et al., 2020; Zhang et al., 2021) to LLM-based collaboration. Frameworks such as Magentic-UI (Mozannar et al., 2025), Cocoa (Feng et al., 2024), Collaborative Gym (CoGym) (Shao et al., 2025), and Collaborative STORM (Shao et al., 2024), A2C (Tariq et al., 2024) further advance these interactions by introducing mechanisms for co-planning and co-execution across real-time web and persistent document environments. By moving toward non-turn-taking protocols, these systems enable more flexible control where humans and agents can operate in the same execution space. Notably, LLM-based collaboration has made significant progress in writing assistance, with examples including CoAuthor (Lee et al., 2022), PEER (Schick et al., 2022), VISAR (Zhang et al., 2023). Early interactive systems like PUMICE (Li et al., 2019) and PLOW (Allen et al., 2007) demonstrated the value of end-user programming and demonstration. Previous studies such as interaction to impact (Zhang et al., 2025), TrustAgent (Hua et al., 2024), and ToolEmu (Ruan et al., 2024) focus primarily on safety and trustworthiness. We shift our focus towards the overall communication patterns in human-agent web browsing collaboration.

7. Conclusion

In this work, we show that human intervention in web navigation constitutes a structured behavioral signal that reflects distinct collaboration styles, ranging from passive supervision to active co-piloting. We introduce CowCorpus, a dataset of 400 real-user web navigation trajectories designed to support the study of intervention modeling in collaborative settings. Our analysis reveals that while proprietary generalist models demonstrate strong reasoning capabilities, they struggle to capture the temporal dynamics of when users choose to intervene. By fine-tuning models on collaborative interaction traces, we bridge this gap, achieving a 61.4–63.4% improvement in intervention prediction accuracy over base LMs. Finally, we deploy our intervention-aware models in a live web agent, and show that anticipating human intervention leads to tangible benefits in practice, increasing user satisfaction by 36.8%. Together, these results highlight the value of deliberately modeling human-agent interaction patterns. We hope that our work encourage the development of agents that are more responsive, adaptive, and capable of functioning as truly collaborative partners.

Acknowledgement

We thank our lab members, Prof. Daniel Fried, Prof. Hirokazu Shirado, Prof. Fernando Diaz, Prof. Tianqi Chen, Sanxing Chen, and Dr. Azad Salam for their valuable feedback. We also thank all our participants for their help in curating CowCorpus. Zora Wang is supported by Google PhD Fellowship. We also thank the developers of TaxyAI for open-sourcing their codebase.

References

A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib. Progress and prospects of the human—robot collaboration. Auton. Robots, 42(5):957–975, June 2018.

ISSN 0929-5593. doi: 10.1007/s10514-017-9677-2. URL https://doi.org/10.1007/s10514-017-9677-2.

J. Allen, N. Chambers, G. Ferguson, L. Galescu, H. Jung, M. Swift, and W. Taysom. Plow: a collaborative task learning agent. In Proceedings of the 22nd National Conference on Artificial Intelligence - Volume 2, AAAI'07, page 1514–1519. AAAI Press, 2007. ISBN 9781577353232.

S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza. Power to the people: The role of humans in interactive machine learning. AI magazine, 35(4):105–120, 2014.

Anthropic. Computer use (beta), 2024. URL https://docs.anthropic.com/en/docs/build-with-claude/computer-use.

Anthropic. Claude 4 system card. Technical report, Anthropic, 2025. URL https://www.anthropic.com/claude-4-system-card. Accessed: 2026-01-29.

G. Bansal, J. W. Vaughan, S. Amershi, E. Horvitz, A. Fourney, H. Mozannar, V. Dibia, and D. S. Weld. Challenges in human-agent communication. arXiv preprint arXiv:2412.10380, 2024.

B. Chandrasekaran and J. M. Conrad. Human-robot collaboration: A survey. In SoutheastCon 2015, pages 1–8, 2015. doi: 10.1109/SECON.2015.7132964.

S. Chen, S. Wiseman, and B. Dhingra. Chatshop: Interactive information seeking with language agents. arXiv preprint arXiv:2404.09911, 2024.

G. Comanici, E. Bieber, M. Schaekermann, I. Pasupat, N. Sachdeva, I. Dhillon, M. Blistein, O. Ram, D. Zhang, E. Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. arXiv preprint arXiv:2507.06261, 2025.

X. Deng, Y. Gu, B. Zheng, S. Chen, S. Stevens, B. Wang, H. Sun, and Y. Su. Mind2web: Towards a generalist agent for the web. Advances in Neural Information Processing Systems, 36, 2024.

K. Feng, K. Pu, M. Latzke, T. August, P. Siangliulue, J. Bragg, D. S. Weld, A. X. Zhang, and J. C. Chang. Cocoa: Co-planning and co-execution with ai agents. arXiv preprint arXiv:2412.10999, 2024.

D. Hadfield-Menell, S. J. Russell, P. Abbeel, and A. Dragan. Cooperative inverse reinforcement learning. Advances in neural information processing systems, 29, 2016.

S. Han, Q. Zhang, Y. Yao, W. Jin, Z. Xu, and C. He. Llm multi-agent systems: Challenges and open problems, 2024. URL https://arxiv.org/abs/2402.03578.

W. Hua, X. Yang, M. Jin, Z. Li, W. Cheng, R. Tang, and Y. Zhang. Trustagent: Towards safe and trustworthy llm-based agents, 2024. URL https://arxiv.org/abs/2402.01586.

F. Huq, Z. Z. Wang, F. F. Xu, T. Ou, S. Zhou, J. P. Bigham, and G. Neubig. CowPilot: A framework for autonomous and human-agent collaborative web navigation. In N. Dziri, S. X. Ren, and S. Diao, editors, Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations), pages 163–172, Albuquerque, New Mexico, Apr. 2025. Association for Computational Linguistics. ISBN 979-8-89176-191-9. doi: 10.18653/v1/2025.naacl-demo.17. URL https://aclanthology.org/2025.naacl-demo.17/.

A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, et al. Gpt-4o system card. arXiv preprint arXiv:2410.21276, 2024.

I. L. Iong, X. Liu, Y. Chen, H. Lai, S. Yao, P. Shen, H. Yu, Y. Dong, and J. Tang. OpenWebAgent: An open toolkit to enable web agents on large language models. In Y. Cao, Y. Feng, and D. Xiong, editors, Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), pages 72–81, Bangkok, Thailand, Aug. 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-demos.8. URL https://aclanthology.org/2024.acl-demos.8.

P. Khadpe, R. Krishna, L. Fei-Fei, J. T. Hancock, and M. S. Bernstein. Conceptual metaphors impact perceptions of human-ai collaboration. Proc. ACM Hum.-Comput. Interact., 4(CSCW2), Oct. 2020. doi: 10.1145/3415234. URL https://doi.org/10.1145/3415234.

M. Lee, P. Liang, and Q. Yang. Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities. Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems, 2022. URL https://api.semanticscholar.org/CorpusID:246016439.

T. J.-J. Li, M. Radensky, J. Jia, K. Singarajah, T. M. Mitchell, and B. A. Myers. Pumice: A multi-modal agent that learns concepts and conditionals from natural language and demonstrations. In Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology, UIST '19, page 577–589, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450368162. doi: 10.1145/3332165.3347899. URL https://doi.org/10.1145/3332165.3347899.

Z. Liao, L. Mo, C. Xu, M. Kang, J. Zhang, C. Xiao, Y. Tian, B. Li, and H. Sun. Eia: Environmental injection attack on generalist web agents for privacy leakage, 2025. URL https://arxiv.org/abs/2409.11295.

D. Misra, J. Langford, and Y. Artzi. Mapping instructions and visual observations to actions with reinforcement learning. arXiv preprint arXiv:1704.08795, 2017.

M. Mitchell, A. Ghosh, A. S. Luccioni, and G. Pistilli. Fully autonomous ai agents should not be developed, 2025. URL https://arxiv.org/abs/2502.02649.

H. Mozannar, G. Bansal, C. Tan, A. Fourney, V. Dibia, J. Chen, J. Gerrits, T. Payne, M. K. Maldaner, M. Grunde-McLaughlin, et al. Magentic-ui: Towards human-in-the-loop agentic systems. arXiv preprint arXiv:2507.22358, 2025.

Y. Pan, D. Kong, S. Zhou, C. Cui, Y. Leng, B. Jiang, H. Liu, Y. Shang, S. Zhou, T. Wu, et al. Webcanvas: Benchmarking web agents in online environments. arXiv preprint arXiv:2406.12373, 2024.

R. Ramrakhya, M. Chang, X. Puig, R. Desai, Z. Kira, and R. Mottaghi. Grounding multimodal llms to embodied agents that ask for help with reinforcement learning, 2025. URL https://arxiv.org/abs/2504.00907.

Y. Ruan, H. Dong, A. Wang, S. Pitis, Y. Zhou, J. Ba, Y. Dubois, C. J. Maddison, and T. Hashimoto. Identifying the risks of lm agents with an lm-emulated sandbox, 2024. URL https://arxiv.org/abs/2309.15817.

W. Saunders, G. Sastry, A. Stuhlmueller, and O. Evans. Trial without error: Towards safe reinforcement learning via human intervention. arXiv preprint arXiv:1707.05173, 2017.

T. Schick, J. Dwivedi-Yu, Z. Jiang, F. Petroni, P. Lewis, G. Izacard, Q. You, C. Nalmpantis, E. Grave, and S. Riedel. Peer: A collaborative language model. ArXiv, abs/2208.11663, 2022. URL https://api.semanticscholar.org/CorpusID:251765117.

Y. Shao, Y. Jiang, T. Kanell, P. Xu, O. Khattab, and M. Lam. Assisting in writing Wikipedia-like articles from scratch with large language models. In K. Duh, H. Gomez, and S. Bethard, editors, Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 6252–6278, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.347. URL https://aclanthology.org/2024.naacl-long.347/.

Y. Shao, V. Samuel, Y. Jiang, J. Yang, and D. Yang. Collaborative gym: A framework for enabling and evaluating human-agent collaboration, 2025. URL https://arxiv.org/abs/2412.15701.

T. Shi, A. Karpathy, L. Fan, J. Hernandez, and P. Liang. World of bits: An open-domain platform for web-based agents. In International Conference on Machine Learning, pages 3135–3144. PMLR, 2017.

S. Tariq, M. B. Chhetri, S. Nepal, and C. Paris. A2c: A modular multi-stage collaborative decision framework for human-ai teams. ArXiv, abs/2401.14432, 2024. URL https://api.semanticscholar.org/CorpusID:267301279.

TaxyAI. Taxy ai, 2024. URL https://taxy.ai/.

D. Wang, E. Churchill, P. Maes, X. Fan, B. Shneiderman, Y. Shi, and Q. Wang. From human-human collaboration to human-ai collaboration: Designing ai systems that can work together with people. In Extended abstracts of the 2020 CHI conference on human factors in computing systems, pages 1–6, 2020.

Z. Z. Wang, Y. Shao, O. Shaikh, D. Fried, G. Neubig, and D. Yang. How do ai agents do human work? comparing ai and human workflows across diverse occupations. arXiv preprint arXiv:2510.22780, 2025.

S. Yao, H. Chen, J. Yang, and K. Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. Advances in Neural Information Processing Systems, 35:20744–20757, 2022.

R. Zhang, N. J. McNeese, G. Freeman, and G. Musick. "an ideal human": Expectations of ai teammates in human-ai teaming. Proc. ACM Hum.-Comput. Interact., 4(CSCW3), Jan. 2021. doi: 10.1145/3432945. URL https://doi.org/10.1145/3432945.

Z. Zhang, J. Gao, R. S. Dhaliwal, and T. J.-J. Li. Visar: A human-ai argumentative writing assistant with visual programming and rapid draft prototyping. In Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, UIST '23, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701320. doi: 10.1145/3586183.3606800. URL https://doi.org/10.1145/3586183.3606800.

Z. J. Zhang, E. Schoop, J. Nichols, A. Mahajan, and A. Swearngin. From interaction to impact: Towards safer ai agent through understanding and evaluating mobile ui operation impacts. In Proceedings of the 30th International Conference on Intelligent User Interfaces, IUI '25, page 727–744, New York, NY, USA, 2025. Association for Computing Machinery. ISBN

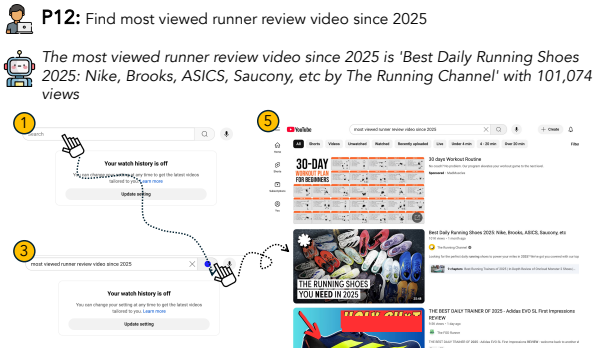
9798400713064. doi: 10.1145/3708359.3712153. URL https://doi.org/10.1145/3708359.3712153.

B. Zheng, B. Gou, S. Salisbury, Z. Du, H. Sun, and Y. Su. WebOlympus: An open platform for web agents on live websites. In D. I. Hernandez Farias, T. Hope, and M. Li, editors, Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 187–197, Miami, Florida, USA, Nov. 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-demo.20. URL https://aclanthology.org/2024.emnlp-demo.20.

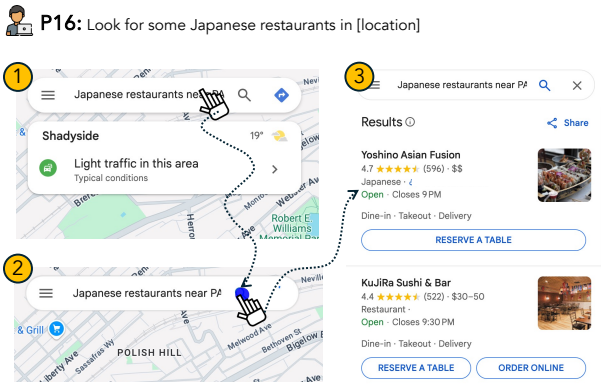
S. Zhou, F. F. Xu, H. Zhu, X. Zhou, R. Lo, A. Sridhar, X. Cheng, T. Ou, Y. Bisk, D. Fried, et al. Webarena: A realistic web environment for building autonomous agents. arXiv preprint arXiv:2307.13854, 2023.

A. COWCORPUS: Human-Agent Collaborative Web Corpus

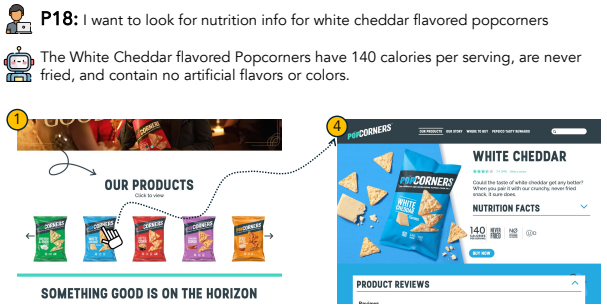
A.1. Task Annotation Setup and Participant Information



(a) Information Access



(b) Personalized Interests and Lifestyle



(c) Product and Service Interaction

Figure 7: Three example tasks from top three free-form task categories. (all identifiable information has been trimmed for anonymity.)

Each annotator was asked to execute 20 web tasks in collaboration with the LM-based agent. The annotators receive a base payment of $0.50 per task, resulting in a total of $10 (provided as an Amazon gift card, as approved by the IRB of our home institute). Our participants are aged between 20–30 and have varied levels of knowledge about AI agents and varied distribution on daily web tasks.

At the beginning of the study, each participant is onboarded using a walk-through installation video and a detailed description of the study setup. We also offered the participants an opt-

Symbols: ChatGPT Claude Perplexity Cursor Deepseek Openhands			
PID	**AI usage frequency**	**Familiarity with Agent** (1–7)	**Task selection**
P1	Few times a day	3 (,)	Personalized Interests & Lifestyle, Planning & Scheduling, Product & Service Interaction
P2	Few times a day	7 ()	Information Access
P3	[undisclosed]	[undisclosed]	Information Access, Personalized Interests & Lifestyle
P4	Few times a day	2 (,)	Information Access, Personalized Interests & Lifestyle
P5	Few times a day	5 (,)	Information Access, Personalized Interests & Lifestyle, Planning & Scheduling, Product & Service Interaction
P6	Few times a day	1 ()	Information Access, Product & Service Interaction, Reasoning & Meta-Analysis
P7	Few times a week	5 ()	Information Access, Personalized Interests & Lifestyle
P8	Few times a day	5 ()	Information Access, Personalized Interests & Lifestyle
P9	Few times a day	4 (, ,)	Information Access, Product & Service Interaction
P10	Few times a day	6 (,)	Information Access
P11	Few times a day	5 (, ,)	Information Access, Content Generation, Security Testing
P12	Few times a day	6 ()	Content Generation, Information Access
P13	Few times a day	5 ()	Information Access, Multi-Agent Collaboration, Reasoning & Meta-Analysis
P14	Few times a day	7 ()	Information Access, Multi-Agent Collaboration, Personalized Interests & Lifestyle, Product & Service Interaction
P15	Few times a day	5 ()	Information Access, Planning & Scheduling, Product & Service Interaction
P16	Few times a day	6 ()	Information Access, Multi-Agent Collaboration, Personalized Interests & Lifestyle, Planning & Scheduling, Security Testing
P17	Few times a day	2 ()	Content Generation, Information Access, Multi-Agent Collaboration
P18	Few times a day	6 ()	Information Access
P19	Few times a day	7 ()	Information Access, Personalized Interests & Lifestyle
P20	Few times a day	5 ()	Information Access

Table 5: Related backgrounds of participants, including frequency of AI usage, familiarity with AI agents with examples, and the types of tasks executed. Agent familiarity is scored on a 1–7 scale.

in option for installation support, where we helped them to install the agent extension. 11 participants requested the installation support call. On average, users take around 1-1.5 hours to complete the entire annotation. We also provided the participants with a pre-paid API key, so participating in our study did not incur any additional cost to the participants. Participants are given up to 3 chances to execute the given task with the AI agent. Even if a user retries the task multiple times, we keep only one trajectory per task so that it is balanced between each user. In such cases, the participant decides which trajectory they want to submit as their final annotation.

We restricted the free-form tasks to have multiple steps (e.g., not achievable via a single button click or type action). We also encourage users to explore tasks of varied length and complexity. All our annotators are familiar with the agentic frameworks. Table 5 provides the distribution of each annotator's expertise and choice of tasks.

We purposefully opt for a self-initiated data annotation paradigm — i.e., at the end of each

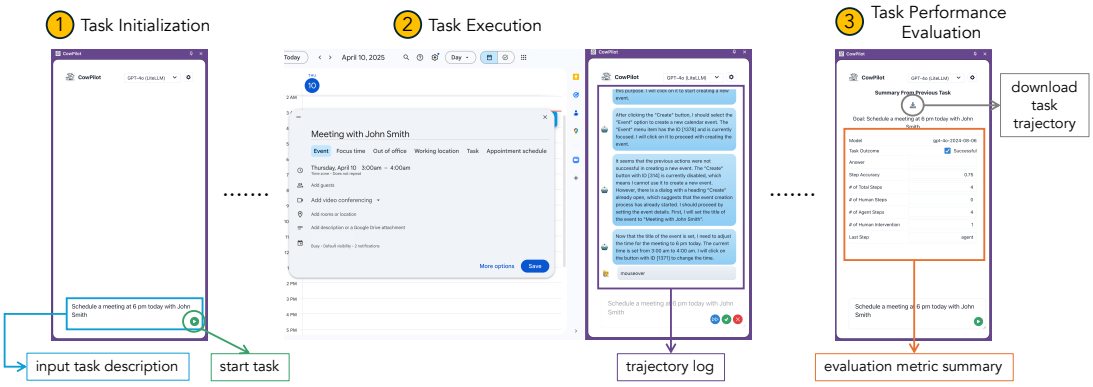


Figure 8: Overview of the collaborative AI agent, CowPilot Huq et al. (2025) used in our data collection. 1) Before the task is initiated, the user gives a textual task description as input. 2) During task execution, the system tracks the actions performed by the user and the agent. 3) After the task is executed, the user can download the task log.

task, participants are shown a summary of their annotation, where they can download the data log form if they wish. Such self-initiated data collection ensures *the user has full control over which data they want to share with us and which they do not*.

A.2. CowPilot: Task Annotation Framework used to annotate CowCorpus

We select CowPilot (Huq et al., 2025), an open-sourced Chrome extension for collaborative web navigation between a human and an LLM-powered agent (Figure 8), to annotate our data. We chose CowPilot since it can be downloaded as a Chrome extension and easily integrated into the users' current browsing workflow. The AI agent is instantiated using LLM, which is capable of extracting the web HTML and generating a step-by-step task execution plan on real-world websites.

The system offers a *suggest-then-execute* workflow where the AI agent proposes UI actions (e.g., clicking buttons, filling text boxes, going to a specific URL) that are visually highlighted for user approval. Users can allow the agent to proceed, pause its execution, or intervene in the agent and take over control. Users can intervene at arbitrary times and for an unlimited number of times if they wish. The entire interaction between the user and the agent, as well as the web environment information, can be logged in detail, capturing both human and agent actions for later analysis — making the system an ideal candidate to collect CowCorpus.

While CowPilot supports a wide range of LLM backends, we used `GPT-4o` in our study for consistency across annotators. Participants primarily used CowPilot in the *copilot mode*, as both agent automation and human intervention are important.

A.3. Analysis

A.3.1. Users Collaborate on Versatile Tasks with Agents

The participants picked a wide range of free-form tasks in CowCorpus. To understand which kinds of tasks a collaborative agent is most impactful with, we categorize the free-form tasks into the 9 categories in Table 2, and annotate common tasks performed by individual users in Table 5.

- Information Access: Users are looking for specific information, such as facts, news, academic papers, and definitions.
- Personalized Interests & Lifestyle: Engagement with entertainment or lifestyle content.
- Product & Service Interaction: Users shop for specific products, compare prices, or book services like flights or rentals.
- Content Generation: Users want to compose written content for communication such as social media posts and emails.
- Multi-Agent Collaboration: Tasks where users coordinate actions across multiple AI tools — such as delegating subtasks to ChatGPT or combining system outputs to accomplish a goal.
- Security Testing: Tasks designed to probe the agent's robustness or ethical boundaries — including prompt injections, adversarial inputs, or potentially inappropriate or harmful requests.
- Planning & Scheduling: Tasks centered on organizing time-bound activities — such as scheduling meetings or creating events.
- Reasoning & Meta-Analysis: Tasks that require analytical thinking or synthesis — such as comparing models, summarizing research contributions, or evaluating options beyond basic retrieval.
- Misc. Tasks that do not clearly align with any of the above categories. These may include underspecified commands or uncommon task types.

Three authors followed an open-coded approach to develop the final categories. Throughout the process, we followed standard practices from past works Deng et al. (2024) and incorporated security Liao et al. (2025) and multi-agent collaborative aspects Han et al. (2024) of AI agents.

A.3.2. Current Agent Bottleneck: Time Demand

While collaboration yielded benefits in terms of control and success, it also introduced additional time demands. As shown in Table 3, in CowPilot, agent execution took on average 93.1 seconds for standard tasks and 71.7 seconds for free-form tasks. In contrast, human intervention time was relatively short—23.9 seconds and 13.8 seconds, respectively

On the other hand, from post-annotation ratings, participants gave a neutral rating of 4.05 when asked if they completed the task faster than they would have without CowPilot (Figure 6, last row), suggesting uncertainty about whether relying on the agent actually saved time for users, compared to executing the tasks themselves.

This agent-heavy time distribution reflects current limitations of LLM-based agents. Each step in the task sequence involves non-trivial latency, and as shown in the time log (Figure 9), agents proceed at a constant pace unless interrupted. These delays accumulate especially in longer-horizon tasks. Moreover, users need to continuously monitor the agent and be ready to intervene, requiring sufficient observation windows to inspect each step, which further slows down the overall process.

On the other hand, with PlowPilot, we see a significant increase in time requirement (an average score of 5.25). Users were more satisfied with the updated interaction module where the agent only intervened as needed rather than the continiously needing to monitor it. They also gave a higher rating of 5.75 that PlowPilot avoided interrupting them unnecessarily.

These findings highlight a key limitation of current agents: the **inability to proactively request help**. Future collaborative agents could incorporate uncertainty estimation mechanisms to identify decision points where user input is most valuable, rather than maintaining a fixed execution pace throughout the task.

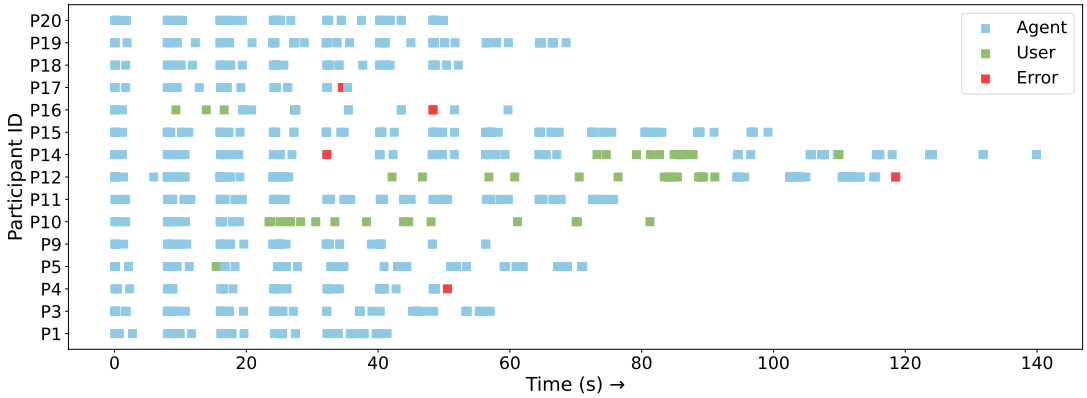


Figure 9: Time log across participants for the same task

A.3.3. Full Benchmark Table

We present the comprehensive evaluation results across all metrics in Table 6, including Precision and Recall which were omitted from the main text for brevity. Our fine-tuned model (Gemma 27B) demonstrate the most balanced performance, maintaining high non-intervention accuracy while significantly improving intervention recall compared to their base counterparts and closed source models.

	Step	**Precision**		**Recall**		**F1 Score**		**PTS**
	Acc	Interv	Non-Interv	Interv	Non-Interv	Interv	Non-Interv	
Baselines								
Always Interv	0.147	0.147	0.000	1.000	0.000	0.257	0.000	0.151
Always No Interv	0.853	0.000	0.853	0.000	1.000	0.000	0.920	0.000
Closed Source Models								
`Claude 4 Sonnet`								
0 shot (w/o reasoning)	0.681	0.179	0.864	0.324	0.743	0.231	0.799	**0.293**
0 shot (w/ reasoning)	0.697	0.169	0.859	0.270	0.771	0.208	0.813	0.164
2 shot (w/o reasoning)	**0.749**	0.158	0.854	0.162	**0.850**	0.160	**0.852**	0.149
2 shot (w/ reasoning)	0.721	0.149	0.853	0.189	0.813	0.167	0.833	0.158
`GPT-4o`								
0 shot (w/o reasoning)	0.741	0.182	0.860	0.216	0.832	0.198	0.846	0.147
2 shot (w/o reasoning)	0.661	0.147	0.852	0.270	0.729	0.190	0.786	0.206
`Gemini 2.5 Pro`								
0 shot (w/o reasoning)	0.681	0.213	0.881	0.432	0.724	0.286	0.795	0.262
0 shot (w/ reasoning)	0.689	0.211	0.878	0.405	0.738	0.278	0.802	0.237
2 shot (w/o reasoning)	0.586	0.181	0.877	0.514	0.598	0.268	0.711	0.287
2 shot (w/ reasoning)	0.641	**0.221**	**0.897**	**0.568**	0.654	**0.318**	0.757	0.243
Open Source Models								
`Gemma 27B (base)`								
0 shot (w/o reasoning)	0.239	0.154	**0.897**	**0.919**	0.121	0.264	0.214	0.187
`Llava 8B (base)`								
0 shot (w/o reasoning)	0.183	0.000	0.852	0.000	0.215	0.000	0.343	0.017
Our Models								
`Gemma 27B (finetuned)`								
0 shot (w/o reasoning)	0.853	**0.500**	0.877	0.216	**0.963**	**0.302**	**0.918**	**0.303**
`Llava 8B (finetuned)`								
0 shot (w/o reasoning)	**0.817**	0.471	0.876	0.216	0.921	0.296	0.897	0.201

Table 6: Step Accuracy, Precision, Recall, F1-score and PTS score for Human Intervention Prediction Task on All Data.

B. Ablation on Modeling Human Intervention

B.1. Ablation: Few Shot Example Count

We investigate whether providing in-context examples (2-shot) improves intervention timing. As shown in Table 7, the impact of few-shot prompting is inconsistent across models. While 2-shot prompts slightly improve PTS for GPT-4o and Gemini, they significantly degrade Claude's intervention timing. This suggests that for certain models, few-shot examples might introduce bias or over-constrain the model's decision boundary, making zero-shot the more robust setting for this specific task.

Model	Shot	Step Acc	F1 Score		PTS
			Interv	Non-Interv	
`Claude 4 Sonnet`	0-shot	0.681	**0.231**	0.799	**0.293**
	2-shot	**0.749**	0.160	**0.852**	0.149
GPT-4o	0-shot	**0.741**	**0.198**	**0.846**	0.147
	2-shot	0.661	0.190	0.786	**0.206**
`Gemini 2.5 Pro`	0-shot	**0.681**	**0.286**	**0.795**	0.262
	2-shot	0.586	0.268	0.711	**0.287**

Table 7: Ablation on Few-Shot Setting (without reasoning)

B.2. Ablation: Models with Reasoning vs No Reasoning

The result from Table 8 reveals a counter-intuitive finding: explicit reasoning tends to lower the Perfect Timing Score (PTS) across models. While reasoning slightly improves Step Accuracy by reducing false positives, it also strengthens the models' bias toward staying silent. This suggests that human intervention is often a reactive, intuitive decision, and forcing a model to articulate a logical justification result in hesitant agents that intervene too late.

Model	Reasoning	Step Acc	F1 Score		PTS
			Interv	Non-Interv	
`Claude 4 Sonnet`	No	0.681	**0.231**	0.799	**0.293**
	Yes	**0.697**	0.208	**0.813**	0.164
`Gemini 2.5 Pro`	No	0.681	**0.286**	0.795	**0.262**
	Yes	**0.689**	0.278	**0.802**	0.237

Table 8: Ablation on Reasoning Setting (0 shot)

B.3. Ablation: Impact of Human Action History

To understand the importance of temporal context in collaborative tasks, we conduct an ablation study by removing the history of human actions from the agent's input. As shown in Table 9, explicitly including the human action history improves model performance, increasing Step Accuracy from 76.27% to 81.36%. This gain suggests that

Includes Human Action History	Step Accuracy	Macro F1
✓	**0.8136**	0.4486
✗	0.7627	0.4327

Table 9: Ablation on the inclusion of human actions (`Claude` 2 shot w/o reasoning).

the agent's decision-making process is not solely dependent on the current state observation but is sensitive to the trajectory of past interactions.

B.4. Ablation: Impact of Input Format on Performance

We further investigate the contribution of different observation modalities by evaluating the agent's performance when restricted to either visual inputs (Screenshots) or structural text inputs (AXTree).

Includes Screenshot?	Includes AXTree?	Step Accuracy	Macro F1
✗	✓	0.697	0.208
✓	✗	0.729	0.171
✓	✓	0.749	0.160

Table 10: Ablation on input format

As shown in Table 10, the Screenshot-only and AXTree-only approaches yield Step Accuracies of 72.9% and 69.7%, respectively. Both are lower than the multimodal baseline (74.9%), demonstrating the benefit of combining visual and structural information.

B.5. Intervention Prediction Remains Robust Under Time Offsets

Across a wide range of α values, PTS provides a consistent and stable metric for comparing model performance. In the zero-shot setting, we observe that increasing α leads to a monotonic decrease in PTS scores across all baseline models, including those equipped with explicit reasoning capabilities, while preserving their relative ranking, as confirmed by a high Kendall's W. This demonstrates that PTS maintains consistent model ranking across variations of α and reliably reflects the underlying quality of a model's intervention prediction.

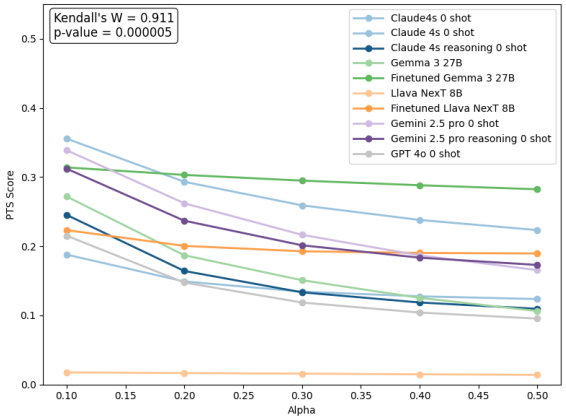


Figure 10: Zero-shot models maintain consistent PTS rankings across α. We sweep α over a fixed grid while holding all inputs constant and recompute PTS for each model under zero-shot setting. Kendall's W significant test reveals that PTS preserves relative ordering under different temporal penalties.

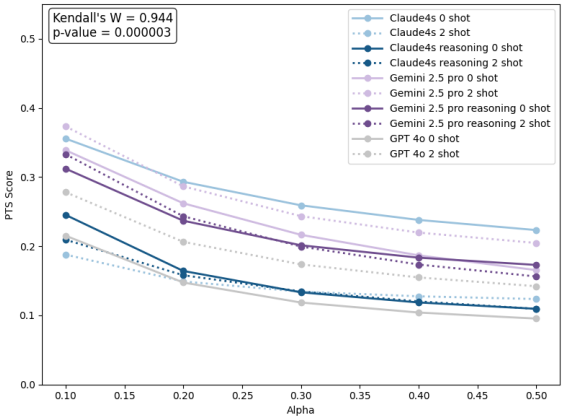


Figure 11: Closed-source models maintain consistent PTS rankings across α. We sweep α over a fixed grid while holding all inputs constant and recompute PTS for each closed-source model. Kendall's W significant test reveals that PTS preserves relative ordering under different temporal penalties.

Notably, the PTS curves of fine-tuned models show much stable to changes in α. Since α controls how strongly early or mistimed predictions are penalized, a model whose intervention timing is already close to the ground truth will accumulate only small penalties regardless of the exact α value. The relative flatness of the fine-tuned curves therefore indicates that these models consistently make temporally accurate predictions with fewer premature or missed calls.