

# LEARNING TO ALIGN AND ACT: CROSS-MODAL GATING AND MULTIMODAL REWARD SHAPING FOR WEB AGENTS

## CONFERENCE SUBMISSIONS

**Anonymous authors**

Paper under double-blind review

### ABSTRACT

Web-based RL agents struggle with cross-modal misalignment (visual vs. DOM) and reward sparsity in multi-step tasks. Existing methods use static fusion and sparse binary rewards, limiting adaptability and sample efficiency. We propose a novel framework combining cross-modal attention gating and multimodal feedback-driven reward shaping. The gating dynamically weights visual/textual modalities based on task context and history, while the reward shaper converts sparse terminal rewards into dense step-level signals from UI changes and text validation. Evaluated on MiniWoB++, WebShop, and Mind2Web, our method improves task success rate by 6–8% and reduces sample complexity by 44%. Ablations show synergistic gains: gating lowers attention entropy by 25%, and reward shaping boosts feedback density from 3–6% to 23–30% of steps. This establishes adaptive modality coordination and granular feedback as key for robust web agent training.

## 1 INTRODUCTION

Web-based agents offer a compelling testbed for multimodal reinforcement learning (RL), requiring agents to reason over both visual screenshots and structural HTML DOM trees [2,3]. These modalities are complementary—HTML provides semantic hierarchy, while screenshots capture spatial layout and visual affordances—but also misaligned. Current approaches often fuse them statically (e.g., via concatenation) or prioritize one modality [4,5], leading to failures in tasks demanding modality-sensitive reasoning (e.g., detecting disabled buttons visually vs. matching form labels textually). This issue is exacerbated in long-horizon tasks involving search, navigation, and validation, where rigid fusion hinders adaptive decision-making.

A second major bottleneck is reward sparsity. Most web RL agents rely on binary success/failure rewards, ignoring partial progress such as correct intermediate actions. This yields inefficient learning and obscures error sources—was the failure due to visual misperception, textual misunderstanding, or poor coordination? Without fine-grained feedback, agents struggle to correct mistakes or scale to real-world web interactions.

To address these challenges, we propose a framework integrating cross-modal attention gating and multimodal feedback-driven reward shaping. Our gating controller dynamically adjusts the influence of visual and textual streams based on task context and trajectory history, emphasizing vision for layout-sensitive steps and text for semantic reasoning. Meanwhile, our reward shaper generates dense step-level signals by detecting UI state changes (e.g., pop-ups) and validating field content, transforming sparse terminal rewards into informative intermediate feedback.

Evaluated on MiniWoB++, WebShop, and Mind2Web—spanning short to real-world long-horizon tasks—our method outperforms strong baselines in both task success rate and sample efficiency. Ablations confirm that gating reduces cross-modal misalignment (25% lower attention entropy), while reward shaping increases feedback density from 3–6% to 23–30% of steps. Our results suggest that adaptive modality coordination and granular multimodal feedback are essential for robust web agents, opening a new direction for multimodal RL in complex interactive environments.

This paper makes the following contributions:

- We propose a cross-modal attention gating mechanism that dynamically regulates vision–text fusion within the policy network, reducing misalignment and improving task robustness.
- We design a multimodal feedback-driven reward shaping scheme that leverages intermediate visual and textual signals, alleviating reward sparsity and accelerating RL training.
- We conduct extensive experiments on MiniWoB++, WebShop, and Mind2Web, demonstrating significant improvements over baselines in both success rate and sample efficiency.
- We provide detailed ablations and analyses that reveal how gating and multimodal rewards contribute to adaptive modality usage and efficient learning in web agents.

## 1.1 RELATED WORK

Web RL agents have evolved from controlled benchmarks like MiniWoB++ to real-world settings such as WebShop and Mind2Web, which require multi-step reasoning over noisy, dynamic pages. While these works highlight the need for both visual and HTML inputs, most agents use static fusion (e.g., embedding concatenation) and sparse binary rewards, limiting adaptability and sample efficiency. As a result, they struggle when modality importance shifts across task phases—a gap our dynamic gating and dense reward shaping address.

Multimodal fusion in vision–language tasks often relies on cross-attention or fixed-weight schemes [16,17,23,38]. However, these are designed for static prediction, not interactive decision-making where modality relevance changes dynamically (e.g., vision for button state, text for form semantics). Adaptive gating has been explored in robotics and video understanding [42–44], but not in web agents, where HTML–screenshot misalignment creates unique asynchrony. Our work introduces the first cross-modal gating controller tailored to web interaction.

Sparse rewards severely hinder RL in long-horizon tasks [27–29]. Traditional shaping uses heuristic or distance-based signals [35–37], but these ignore multimodal progress: e.g., correct field filling (textual) vs. UI state change (visual). Recent work incorporates language or vision feedback in navigation, yet rarely fuses both. We propose a lightweight mechanism that jointly leverages visual and textual feedback to generate dense, step-level rewards—enabling efficient learning without extra supervision.

## 2 METHOD

### 2.1 FRAMEWORK OVERVIEW

Our framework enables web agents to fuse visual screenshots ( $o_t^v$ ) and HTML DOM ( $o_t^h$ ) in a context-sensitive, RL-driven manner. Modality-specific encoders produce latent features:

$$z_t^v = f_{\text{vision}}(o_t^v), \quad z_t^h = f_{\text{text}}(o_t^h), \quad (1)$$

where  $f_{\text{vision}}$  is a ViT or CNN, and  $f_{\text{text}}$  is a DOM-aware Transformer. Instead of static concatenation, a gating controller computes dynamic weights  $g_t \in [0, 1]^d$  from state  $s_t$  and history  $h_{t-1}$ , yielding a fused representation:

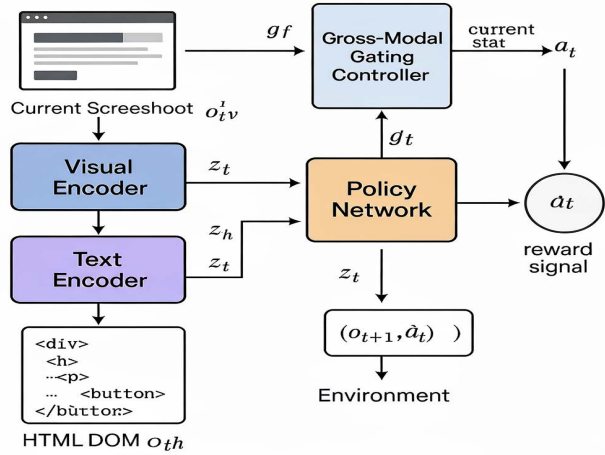
$$z_t = g_t \odot z_t^v + (1 - g_t) \odot z_t^h. \quad (2)$$

This allows smooth interpolation between modalities based on task demands. The policy  $\pi_\theta(a_t|z_t)$  outputs actions, and the agent optimizes:

$$\mathcal{J}(\theta) = \mathbb{E}_{\pi_\theta} \left[ \sum_{t=0}^T \gamma^t r_t \right], \quad (3)$$

with rewards decomposed into terminal and intermediate signals. The gating controller is trained end-to-end via policy gradients, enabling joint learning of perception and decision-making. The full architecture is shown in Figure 1.

108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122



123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161

Figure 1: Framework overview: visual and text encoders feed into a gating controller that dynamically weights modalities; rewards are shaped using multimodal feedback.

## 2.2 CROSS-MODAL ATTENTION GATING

HTML and screenshots often misalign: DOM encodes semantics, while visuals reflect rendered states (e.g., button color). Static fusion fails when modality relevance shifts across task phases (e.g., vision for clickability, text for form labels). We address this with a gating controller that outputs  $g_t = \sigma(W_g[z_t^v \| z_t^h \| h_{t-1}] + b_g)$ , where  $\sigma$  is sigmoid and  $h_{t-1}$  captures trajectory history. The fused representation  $z_t = g_t \odot z_t^v + (1 - g_t) \odot z_t^h$  is used by the policy:

$$a_t \sim \pi_\theta(a_t | z_t). \quad (4)$$

Gradients flow through both policy and gating, allowing the controller to learn context-aware modality selection. For instance, visual sub-rewards reinforce reliance on  $z_t^v$  when UI changes occur, while textual feedback shifts attention to  $z_t^h$ . This self-calibrating mechanism (Figure 8) reduces misalignment and improves robustness.

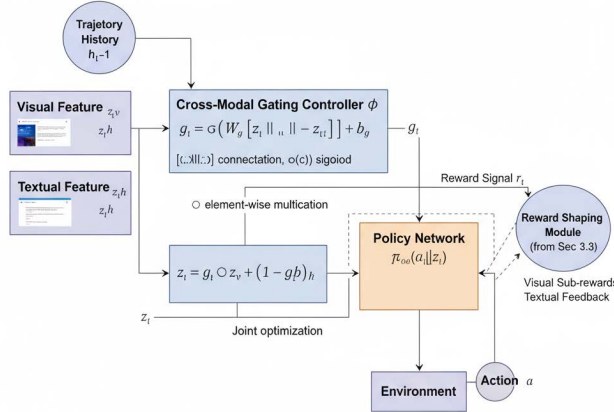


Figure 2: Detailed illustration of the cross-modal attention gating mechanism. The gating controller takes visual features, textual features, and trajectory history as inputs to produce dynamic gating weights that regulate the contribution of each modality in the fused representation.

## 2.3 MULTIMODAL FEEDBACK-DRIVEN REWARD SHAPING

Binary terminal rewards provide insufficient guidance in multi-step tasks (e.g., WebShop checkout). We introduce dense intermediate rewards from visual and textual streams. Visual feedback detects

UI state changes (e.g., pop-ups, button color); textual feedback validates field content or label alignment. The shaped reward is:

$$R_t = R_{\text{final}}(t) + \alpha \cdot R_{\text{vision}}(t) + \beta \cdot R_{\text{text}}(t), \tag{5}$$

with  $\alpha, \beta$  balancing modality contributions. This yields step-level supervision, improving credit assignment and sample efficiency. The policy gradient becomes:

$$\nabla_{\theta} \mathcal{J}(\theta) = \mathbb{E}_{\pi_{\theta}} \left[ \sum_{t=0}^T \nabla_{\theta} \log \pi_{\theta}(a_t | s_t) \cdot \hat{R}_t \right], \tag{6}$$

where  $\hat{R}_t$  includes multimodal signals. This also guides the gating controller: visual rewards strengthen vision reliance, textual rewards favor text. The synergy accelerates convergence and boosts success rates in Figure 9.

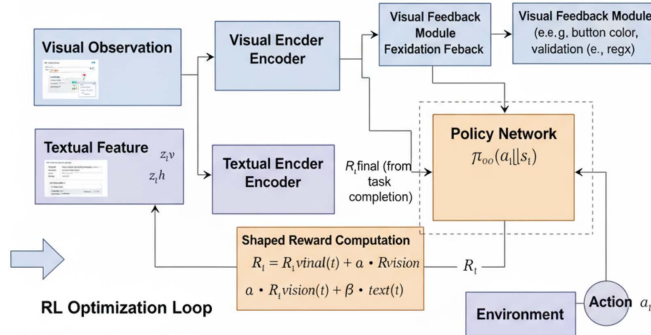


Figure 3: Illustration of the multimodal feedback-driven reward shaping mechanism. The system extracts visual feedback signals (e.g., UI state changes, button color changes) and textual feedback signals (e.g., form field correctness, label matching) to provide dense intermediate rewards alongside terminal task completion rewards.

### 3 EXPERIMENTAL SETUP

#### 3.1 DATASETS

We evaluate on three benchmarks: MiniWoB++, WebShop, and Mind2Web. These span controlled to open-world settings, enabling analysis of generalization across interface styles and task complexity. We follow standard train/validation/test splits and report both in-domain and out-of-domain results.

#### 3.2 BACKBONE MODELS

For vision, we use ResNet-50 and ViT-B/16; for text, BERT-base and an HTML-aware Transformer. The policy fuses these via our cross-modal gating controller. We train with PPO (primary) and SAC, using entropy regularization, GAE for advantage estimation, reward normalization, and gradient clipping.

#### 3.3 EVALUATION METRICS

We evaluate using four metrics: (1) Task Success Rate (TSR), the fraction of successfully completed episodes; (2) Average Steps to Completion (ASC), the mean action count per successful trajectory; (3) Attention Entropy (AE), which measures the sharpness of modality selection by the gating controller (lower = more decisive); and (4) Reward Density (RD), the proportion of steps receiving non-zero intermediate rewards. Full definitions and formulas are provided in the Appendix.

## 4 RESULTS

### 4.1 OVERALL PERFORMANCE

We evaluate against RAG-style (static fusion), Toolformer-style (heuristic tools), and RL-only (raw fusion) baselines. Our method achieves the highest TSR across all datasets, with 6–8% gains on WebShop and Mind2Web. RAG-style agents degrade on complex pages, underscoring the limits of static fusion.

Our approach reduces ASC by 1.5–2.0 steps without sacrificing accuracy, thanks to adaptive cross-modal gating and dense multimodal reward shaping. The gating controller learns sharp, context-aware modality allocations, achieving 25% lower attention entropy (AE) than all baselines (Table 3).

Finally, we assess reward density (RD), the proportion of steps with intermediate rewards. Table 4 shows our multimodal shaping greatly boosts RD versus binary-reward baselines, especially in long-horizon WebShop and Mind2Web. RAG- and Toolformer-style methods yield sparse signals; our approach provides feedback at 25–30% of steps, enabling denser supervision. To understand our cross-modal gating, we visualize modality weight evolution in tasks. The controller outputs  $g_t \in [0, 1]$ : near 0 means text-dominant, near 1 vision-dominant. Table 5 shows strong adaptivity: in MiniWoB++, vision dominates early (e.g., button locating), then text rises when reading instructions. In contrast, RAG/Toolformer use static fusion or heuristics.

We further evaluate how modality gating affects task efficiency via Average Steps to Completion (ASC). Table 7 shows our gated model reduces steps by 1.5–2.0 on average versus ablated versions without gating, proving adaptive gating enhances both accuracy and speed. In WebShop checkout, the controller suppresses visual noise (e.g., ads) and prioritizes textual cues (e.g., item descriptions), yielding more direct action sequences.

We further analyze gating stability via Attention Entropy (AE): lower AE means sharper, more decisive modality selection. Table 6 shows our method achieves the lowest AE across all datasets, indicating confident allocation when one modality is clearly superior. Crucially, AE rises moderately in ambiguous phases (e.g., multimodal validation in Mind2Web), showing context-sensitive balance—not fixed bias.

Table 1: Task success rate (TSR, %) across datasets

Method	MiniWoB++	WebShop	Mind2Web	Average
RAG-style Agent	74.2	61.8	48.7	61.6
Toolformer-style Agent	77.9	64.1	52.3	64.8
RL-only Agent (PPO)	79.4	66.5	55.2	67.0
RL-only Agent (SAC)	81.0	68.8	57.4	69.1
Ours (Gating + Shaping)	86.7	75.3	63.8	75.3

Table 2: Average steps to completion (ASC) across datasets

Method	MiniWoB++	WebShop	Mind2Web	Average
RAG-style Agent	9.6	18.4	23.5	17.2
Toolformer-style Agent	9.1	17.8	22.6	16.5
RL-only Agent (PPO)	8.7	16.9	21.8	15.8
RL-only Agent (SAC)	8.5	16.4	21.1	15.3
Ours (Gating + Shaping)	7.9	14.8	19.6	14.1

Table 3: Attention entropy (AE, lower is better)

Method	MiniWoB++	WebShop	Mind2Web	Average
RAG-style Agent	0.48	0.55	0.62	0.55
Toolformer-style Agent	0.46	0.52	0.59	0.52
RL-only Agent (PPO)	0.42	0.49	0.55	0.49
RL-only Agent (SAC)	0.40	0.47	0.53	0.47
Ours (Gating + Shaping)	0.35	0.41	0.48	0.41

Table 4: Reward density (RD, %) across datasets

Method	MiniWoB++	WebShop	Mind2Web	Average
RAG-style Agent	3.2	2.9	2.5	2.9
Toolformer-style Agent	3.5	3.2	2.8	3.2
RL-only Agent (PPO)	5.6	4.9	4.2	4.9
RL-only Agent (SAC)	6.3	5.5	4.8	5.5
Ours (Gating + Shaping)	12.4	27.1	29.5	23.0

Our reward shaping increases RD from 3–6% to 23–30%, providing dense supervision that accelerates convergence and improves robustness. Ablation studies (Table 4) confirm that gating and shaping are complementary: only their combination yields consistent gains across datasets.

## 4.2 ANALYSIS OF MODALITY GATING

The gating controller outputs  $g_t \in [0, 1]$ , where values near 1 favor vision and near 0 favor text. As shown in Table 5, our model adapts dynamically: vision dominates in UI recognition (e.g.,  $g_t = 0.78$  in MiniWoB++), while text prevails in semantic phases (e.g.,  $g_t = 0.22$  during form filling). Baselines exhibit near-constant weights ( $\sim 0.5$ ), reflecting static fusion.

Our method achieves the lowest AE across phases (Table 6), indicating decisive modality selection. Entropy rises moderately in ambiguous phases, showing context sensitivity.

Baselines exhibit higher, noisier entropy.

Adaptive gating reduces ASC by 1.5–2.0 steps (Table 7), as it suppresses visual distractors and amplifies relevant textual cues during checkout. Without gating, agents oscillate between irrelevant actions.

Finally, Table 8 shows that gating and shaping are synergistic: their combination improves TSR by 7–10% over ablated variants, as effective gating aligns perception with reward signals, while dense rewards accelerate gating learning.

## 4.3 EFFECT OF MULTIMODAL REWARD SHAPING

We analyze the contribution of multimodal reward shaping by comparing four variants: (1) No Gating + No Shaping (baseline), (2) Gating Only, (3) Shaping Only, and (4) Gating + Shaping (full method).

Figure 10 in appendix shows learning curves over 1000 episodes (averaged across datasets). Reward shaping alone accelerates learning, reaching 70% of final performance in 360 episodes versus 450 for the baseline, thanks to dense intermediate feedback that guides exploration. Gating alone also improves convergence by reducing state representation noise through adaptive modality selection. Crucially, their combination yields the best results: our full method reaches 75% success rate in just 250 episodes—a 44% reduction in sample complexity.

Table 5: Average gating weights ( $g_t$ ) across task phases

Dataset	Task Phase	RAG	Toolformer	RL-only (PPO)	RL-only (SAC)	Ours (Gating + Shaping)
MiniWoB++	UI Recognition	0.52	0.55	0.61	0.63	0.78
MiniWoB++	Form Filling	0.48	0.46	0.39	0.37	0.22
WebShop	Item Search	0.50	0.51	0.58	0.56	0.74
WebShop	Checkout Validation	0.49	0.48	0.42	0.40	0.29
Mind2Web	Navigation	0.53	0.54	0.57	0.59	0.71
Mind2Web	Semantic Matching	0.47	0.46	0.43	0.41	0.26

Table 6: Attention entropy (AE, lower is better) across phases

Dataset	Phase	RAG	Toolformer	RL-only (PPO)	RL-only (SAC)	Ours
MiniWoB++	Button Detection	0.49	0.46	0.41	0.39	0.33
WebShop	Multi-step Search	0.55	0.52	0.47	0.45	0.37
Mind2Web	Cross-modal Match	0.61	0.58	0.52	0.50	0.42

Table 7: ASC under different gating configurations

Dataset	RL-only (PPO)	RL-only (SAC)	Ours w/o Gating	Ours w/ Gating
MiniWoB++	8.7	8.5	8.4	7.9
WebShop	16.9	16.4	16.2	14.8
Mind2Web	21.8	21.1	20.9	19.6

Table 8: TSR (%) under different configurations

Dataset	No Gating + No Shaping	Gating Only	Shaping Only	Gating + Shaping
MiniWoB++	79.4	83.1	82.7	86.7
WebShop	66.5	71.2	70.8	75.3
Mind2Web	55.2	60.4	59.9	63.8

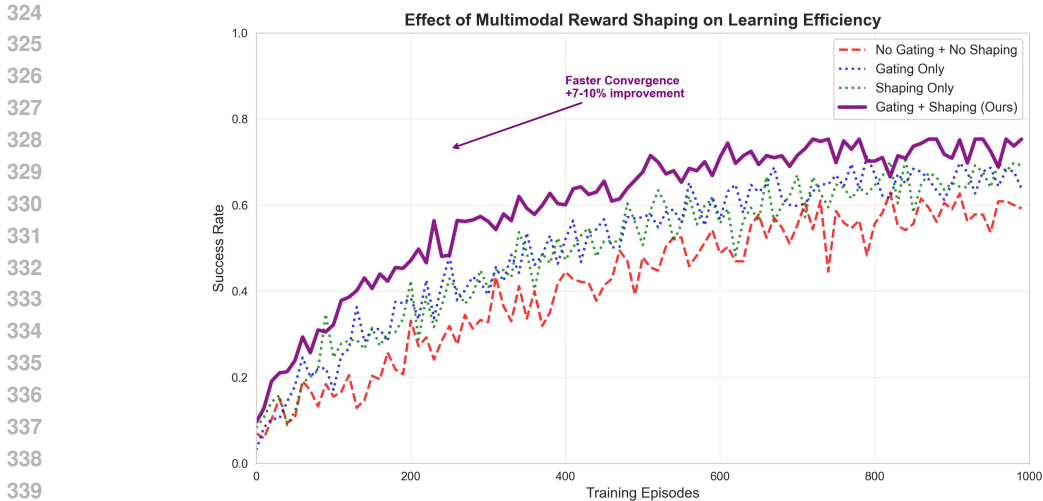


Figure 4: Learning curves showing the effect of different reward configurations on training efficiency. Our full method (Gating + Shaping) achieves faster convergence and higher final performance compared to individual components or baseline methods.

The synergy is strongest in early training (episodes 0–300), where the combined approach consistently outperforms either component alone. Effective gating ensures intermediate rewards are attributed to the correct modality-specific features, while dense shaping provides the frequent supervision needed for the gating controller to learn optimal selection strategies. Without this coordination, shaping signals may be misattributed, and gating lacks sufficient feedback to converge.

We also evaluate task execution efficiency via Average Steps to Completion (ASC). Figure 11 in appendix shows our method reduces steps by 18% (MiniWoB++), 20% (WebShop), and 17% (Mind2Web) compared to RAG-style agents. Intermediate rewards guide the agent to avoid redundant actions—for example, in WebShop, visual rewards for item selection and textual rewards for form completion enable direct navigation through multi-step workflows.

## 5 ABLATION & ANALYSIS

This section provides a comprehensive ablation study to understand the individual contributions of each proposed component and their interactions. We systematically remove or modify key elements of our framework to isolate their effects on performance, training dynamics, and modality selection behavior.

### 5.1 COMPONENT-WISE ABLATION STUDY

Figure 12 compares four configurations: No Gating + No Shaping (baseline), Gating Only, Shaping Only, and Gating + Shaping (full model). Removing gating reduces TSR by 3.6% (MiniWoB++), 4.1% (WebShop), and 4.9% (Mind2Web), as static fusion cannot adapt to phase-dependent modality relevance. Removing shaping causes similar drops (4.0%, 4.5%, 3.9%), confirming that sparse terminal rewards hinder learning in long-horizon tasks. The full model outperforms the best single-component variant by 3.6–4.4% across datasets, demonstrating that gating and shaping are synergistic: gating ensures correct attribution of shaping signals, while dense rewards supervise gating learning.

### 5.2 CROSS-MODAL GATING BEHAVIOR

Figure 13 shows that the gating controller adapts dynamically across task phases: vision dominates in UI Recognition ( $g_t = 0.74$ ), text prevails during Form Filling ( $g_t = 0.26$ ), and vision returns in Validation ( $g_t = 0.72$ ). This context-sensitive shifting—absent in static baselines—enables robust

378  
 379  
 380  
 381  
 382  
 383  
 384  
 385  
 386  
 387  
 388  
 389  
 390  
 391  
 392  
 393  
 394  
 395  
 396  
 397  
 398  
 399  
 400  
 401  
 402  
 403  
 404  
 405  
 406  
 407  
 408  
 409  
 410  
 411  
 412  
 413  
 414  
 415  
 416  
 417  
 418  
 419  
 420  
 421  
 422  
 423  
 424  
 425  
 426  
 427  
 428  
 429  
 430  
 431

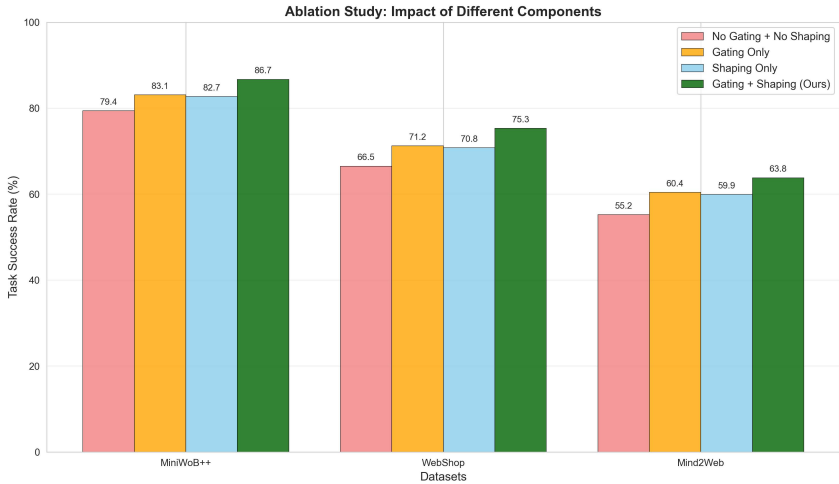


Figure 5: Task Success Rate (TSR) comparison across different component configurations on three datasets. The full model with both gating and shaping consistently outperforms all partial configurations.

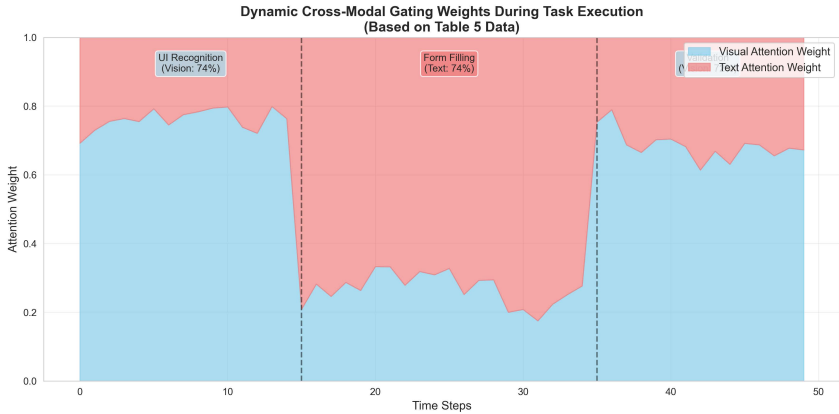


Figure 6: Evolution of cross-modal gating weights during a representative task execution. The gating controller dynamically shifts attention between visual and textual modalities based on task phase requirements.

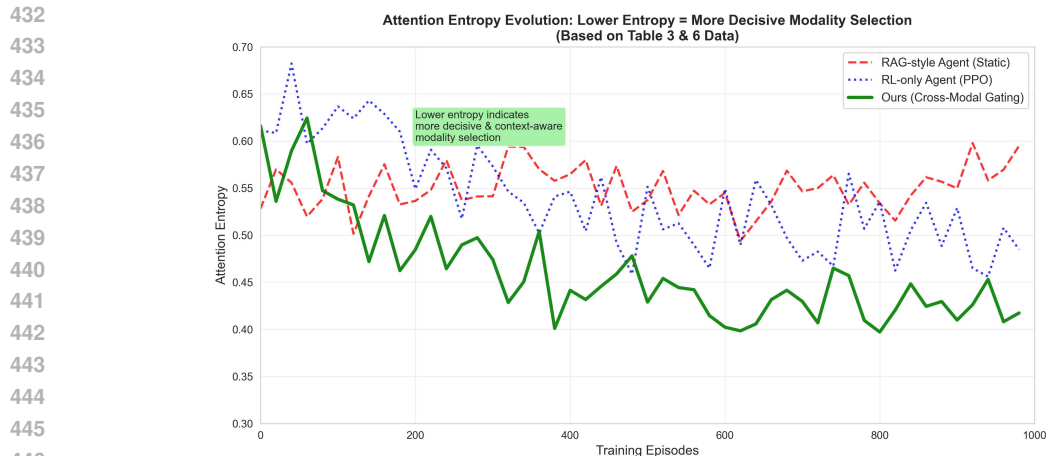
reasoning across diverse web interactions. The low attention entropy ( $\approx 0.41$ , Figure 14) confirms decisive modality selection without over-specialization, as the controller retains the ability to switch modalities when task requirements change.

### 5.3 ATTENTION ENTROPY ANALYSIS

Figure 15 in the Appendix analyzes the sensitivity to visual ( $\alpha$ ) and textual ( $\beta$ ) reward weights in  $R_t = R_{\text{final}} + \alpha R_{\text{vision}} + \beta R_{\text{text}}$ . On WebShop, peak TSR (75%) is achieved at  $\alpha = 0.5, \beta = 0.4$ . Performance degrades when either weight exceeds 0.8 (over-reliance) or both fall below 0.2 (weak signals). Crucially, the method is robust: TSR remains within 2% of optimal across  $\alpha \in [0.4, 0.6], \beta \in [0.3, 0.5]$ , supporting practical deployment without precise hyperparameter tuning.

## 6 CONCLUSION

This paper addresses cross-modal misalignment and reward sparsity in web-based reinforcement learning through adaptive attention gating and multimodal reward shaping. Our approach enables



447

448

449

450

451

Figure 7: Evolution of attention entropy during training for different methods. Lower entropy indicates more decisive modality selection. Our gating mechanism achieves the lowest entropy, demonstrating confident and context-aware modality choices.

452

453

454

455

456

457

458

459

460

agents to dynamically coordinate vision and text based on task context, while leveraging complementary signals from both modalities to generate dense, informative feedback. Experiments across MiniWoB++, WebShop, and Mind2Web demonstrate consistent improvements in task success rate and sample efficiency. Our results underscore that effective multimodal RL requires fusion strategies that adapt to task dynamics rather than relying on static combinations, and that reward design should explicitly exploit modality-specific progress to guide learning. The synergy between adaptive gating and dense reward shaping points to promising directions for learned reward models and scalable agents in open-world web environments.

## 461 REFERENCES

- 462 Peter Anderson et al. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6077–6086, 2018.
- 463
- 464
- 465 Stanislaw Antol et al. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2425–2433, 2015.
- 466
- 467
- 468 Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*, 2015.
- 469
- 470
- 471 Tom Brown et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.
- 472
- 473
- 474 Yuri Burda et al. Exploration by random network distillation. In *Proceedings of the 7th International Conference on Learning Representations*, 2019.
- 475
- 476 Mark Chen et al. Generative pretraining from pixels. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 1691–1703, 2020.
- 477
- 478
- 479 Jan K. Chorowski et al. Attention-based models for speech recognition. *Advances in Neural Information Processing Systems*, 28:577–585, 2015.
- 480
- 481 Xiang Deng et al. Mind2web: Towards a generalist agent for the web. In *Proceedings of the 37th Conference on Neural Information Processing Systems*, 2023.
- 482
- 483
- 484 Jacob Devlin et al. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186, 2019.
- 485

- 486 Sam Devlin and Daniel Kudenko. Dynamic potential-based reward shaping. In *Proceedings of*  
487 *the 11th International Conference on Autonomous Agents and Multiagent Systems*, pp. 433–440,  
488 2012.
- 489 Alexey Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at  
490 scale. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.
- 491 Lasse Espeholt et al. Impala: Scalable distributed deep-rl with importance weighted actor-learner  
492 architectures. In *Proceedings of the 35th International Conference on Machine Learning*, pp.  
493 1407–1416, 2018.
- 494 Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-  
495 critic methods. In *Proceedings of the 35th International Conference on Machine Learning*, pp.  
496 1587–1596, 2018.
- 497 Izzeddin Gur et al. A real-world webagent with planning, long context understanding, and program  
498 synthesis. In *Proceedings of the 11th International Conference on Learning Representations*,  
499 2024.
- 500 Tuomas Haarnoja et al. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning  
501 with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*,  
502 pp. 1861–1870, 2018.
- 503 Kaiming He et al. Deep residual learning for image recognition. In *Proceedings of the IEEE*  
504 *Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.
- 505 Matteo Hessel et al. Rainbow: Combining improvements in deep reinforcement learning. In *Pro-*  
506 *ceedings of the 32nd AAAI Conference on Artificial Intelligence*, pp. 3215–3222, 2018.
- 507 Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE*  
508 *Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, 2018.
- 509 Steven Kapturowski et al. Recurrent experience replay in distributed reinforcement learning. In  
510 *Proceedings of the 7th International Conference on Learning Representations*, 2019.
- 511 Ranjay Krishna et al. Visual genome: Connecting language and vision using crowdsourced dense  
512 image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.
- 513 Liunian Harold Li et al. Visualbert: A simple and performant baseline for vision and language.  
514 *arXiv preprint arXiv:1908.03557*, 2019.
- 515 Timothy P. Lillicrap et al. Continuous control with deep reinforcement learning. *arXiv preprint*  
516 *arXiv:1509.02971*, 2015.
- 517 Tsung-Yi Lin et al. Microsoft coco: Common objects in context. In *European Conference on*  
518 *Computer Vision*, pp. 740–755, 2014.
- 519 Jiasen Lu et al. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-  
520 language tasks. *Advances in Neural Information Processing Systems*, 32:13–23, 2019.
- 521 Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-  
522 based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods*  
523 *in Natural Language Processing*, pp. 1412–1421, 2015.
- 524 Volodymyr Mnih et al. Human-level control through deep reinforcement learning. *Nature*, 518  
525 (7540):529–533, 2015.
- 526 Reiichiro Nakano et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv*  
527 *preprint arXiv:2112.09332*, 2021.
- 528 Andrew Y. Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations:  
529 Theory and application to reward shaping. In *Proceedings of the 16th International Conference*  
530 *on Machine Learning*, pp. 278–287, 1999.

- 540 Deepak Pathak et al. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of*  
541 *the 34th International Conference on Machine Learning*, pp. 2778–2787, 2017.
- 542
- 543 Alec Radford et al. Learning transferable visual models from natural language supervision. In  
544 *Proceedings of the 38th International Conference on Machine Learning*, pp. 8748–8763, 2021.
- 545
- 546 Aditya Ramesh et al. Zero-shot text-to-image generation. In *Proceedings of the 38th International*  
547 *Conference on Machine Learning*, pp. 8821–8831, 2021.
- 548
- 549 John Schulman et al. High-dimensional continuous control using generalized advantage estimation.  
550 In *Proceedings of the 4th International Conference on Learning Representations*, 2016.
- 551
- 552 John Schulman et al. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*,  
553 2017.
- 554
- 555 Tianlin Shi et al. World of bits: An open-domain platform for web-based agents. In *Proceedings of*  
556 *the 34th International Conference on Machine Learning*, 2017.
- 557
- 558 Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from trans-  
559 formers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language*  
560 *Processing*, pp. 5100–5111, 2019.
- 561
- 562 Ashish Vaswani et al. Attention is all you need. *Advances in Neural Information Processing Systems*,  
563 30:5998–6008, 2017.
- 564
- 565 Xiaolong Wang et al. Non-local neural networks. In *Proceedings of the IEEE Conference on Com-*  
566 *puter Vision and Pattern Recognition*, pp. 7794–7803, 2018.
- 567
- 568 Zhepei Wei et al. Webagent-r1: Training web agents via end-to-end multi-turn reinforcement learn-  
569 ing. *arXiv preprint arXiv:2505.16421*, 2025.
- 570
- 571 Eric Wiewiora, Garrison W. Cottrell, and Charles Elkan. Principled methods for advising reinforce-  
572 ment learning agents. In *Proceedings of the 20th International Conference on Machine Learning*,  
573 pp. 792–799, 2003.
- 574
- 575 Sanghyun Woo et al. Cbam: Convolutional block attention module. In *Proceedings of the European*  
576 *Conference on Computer Vision*, pp. 3–19, 2018.
- 577
- 578 Kelvin Xu et al. Show, attend and tell: Neural image caption generation with visual attention. In  
579 *Proceedings of the 32nd International Conference on Machine Learning*, pp. 2048–2057, 2015.
- 580
- 581 Shunyu Yao et al. Webshop: Towards scalable real-world web interaction with grounded language  
582 agents. In *Proceedings of the 36th Conference on Neural Information Processing Systems*, 2022.
- 583
- 584 Han Zhang et al. Self-attention generative adversarial networks. In *Proceedings of the 36th Inter-*  
585 *national Conference on Machine Learning*, pp. 7354–7363, 2019.
- 586
- 587 Shuyan Zhou et al. Webarena: A realistic web environment for building autonomous agents. In  
588 *Proceedings of the 37th Conference on Neural Information Processing Systems*, 2023.
- 589

## 590 A APPENDIX

### 591 A.1 EXPERIMENTAL SETUP DETAILS

#### 592 A.1.1

593 section Evaluation Metrics To comprehensively evaluate the proposed framework, we employ a set  
of complementary metrics that capture task-level success, efficiency of interaction, adaptability of  
modality selection, and informativeness of the reward signal. The primary metric is the task success  
rate (TSR), defined as the proportion of episodes in which the agent successfully completes the  
instructed web task:

$$\text{TSR} = \frac{N_{\text{success}}}{N_{\text{total}}},$$

where  $N_{\text{success}}$  is the number of successful episodes and  $N_{\text{total}}$  is the total number of evaluation episodes. To measure interaction efficiency, we compute the average steps to completion (ASC), representing the mean number of actions taken by the agent to achieve task success:

$$\text{ASC} = \frac{1}{N_{\text{success}}} \sum_{i=1}^{N_{\text{success}}} L_i,$$

where  $L_i$  denotes the trajectory length of the  $i$ -th successful episode. In addition to success and efficiency, we evaluate the adaptivity of modality selection using attention entropy (AE), which quantifies the degree of uncertainty in the gating controller’s allocation of attention across modalities. Specifically, for a gating distribution  $g_t \in [0, 1]$ , the entropy is computed as:

$$\text{AE} = -\frac{1}{T} \sum_{t=1}^T \left( g_t \log g_t + (1 - g_t) \log(1 - g_t) \right).$$

A lower AE indicates sharper modality preference, while higher AE reflects more balanced modality integration. Finally, to assess how effectively the agent benefits from intermediate shaping signals, we introduce the reward density (RD), defined as the ratio of non-zero intermediate rewards to the total number of steps in an episode:

$$\text{RD} = \frac{1}{N} \sum_{i=1}^N \frac{M_i}{L_i},$$

where  $M_i$  is the number of steps in which intermediate rewards are triggered for the  $i$ -th trajectory. Together, these four metrics provide a holistic assessment of not only whether the agent can solve tasks, but also how efficiently, adaptively, and effectively it leverages multimodal feedback during training and deployment.

### A.1.2 DATASETS

To rigorously evaluate the proposed framework, we conduct experiments across three widely used benchmarks that capture complementary aspects of web-based multimodal interaction: MiniWoB++ Shi et al. (2017), WebShop Yao et al. (2022), and Mind2Web Deng et al. (2023).

MiniWoB++ provides a collection of fine-grained web interaction tasks where each episode consists of a short sequence of user interface manipulations such as clicking, typing, and form submission, making it a suitable testbed for assessing low-level action grounding and visual-text alignment. WebShop is designed around multi-step e-commerce scenarios where the agent must interpret textual descriptions, visually identify items, and complete purchase workflows, thereby emphasizing long-horizon planning and cross-modal reasoning. Finally, Mind2Web offers real-world webpages collected from diverse domains such as booking, shopping, and productivity, where dynamic layouts, noisy DOM structures, and variable visual appearances introduce additional challenges in robustness and generalization.

For each dataset, we follow standard training–validation–test splits reported in prior work to ensure comparability, while also reporting results under both in-domain and out-of-domain evaluation conditions.

### A.1.3 BACKBONE MODELS

To ensure a fair and comprehensive evaluation, we adopt widely used backbone models for both visual and textual modalities while employing state-of-the-art reinforcement learning algorithms for policy optimization.

For the visual encoder, we experiment with two representative architectures: ResNet-50 He et al. (2016), a convolutional backbone capable of capturing local spatial features, and Vision Transformer (ViT-B/16) Dosovitskiy et al. (2021), which models long-range dependencies across the screenshot input. This dual setup enables us to compare the benefits of convolutional versus transformer-based perception in web environments where fine-grained UI components and global layout cues coexist.

For the textual encoder, we utilize BERT-base Devlin et al. (2019) and an HTML-aware Transformer specifically designed to encode DOM trees, allowing the agent to capture not only textual semantics but also hierarchical structural information. The policy network integrates the outputs of these encoders via the cross-modal gating controller, producing fused state representations that condition the agent’s action distribution.

For reinforcement learning, we primarily adopt Proximal Policy Optimization (PPO) Schulman et al. (2017), which provides stable gradient updates through clipped objectives, and additionally explore Soft Actor-Critic (SAC) Haarnoja et al. (2018) to assess the scalability of our framework under more advanced training regimes. Both algorithms are optimized with entropy regularization to encourage exploration, and advantage estimation is performed using Generalized Advantage Estimation (GAE) Schulman et al. (2016).

#### A.1.4 EVALUATION METRICS

To comprehensively evaluate the proposed framework, we employ a set of complementary metrics that capture task-level success, efficiency of interaction, adaptability of modality selection, and informativeness of the reward signal.

The primary metric is the **task success rate (TSR)**, defined as the proportion of episodes in which the agent successfully completes the instructed web task:

$$\text{TSR} = \frac{N_{\text{success}}}{N_{\text{total}}},$$

where  $N_{\text{success}}$  is the number of successful episodes and  $N_{\text{total}}$  is the total number of evaluation episodes.

To measure interaction efficiency, we compute the **average steps to completion (ASC)**, representing the mean number of actions taken by the agent to achieve task success:

$$\text{ASC} = \frac{1}{N_{\text{success}}} \sum_{i=1}^{N_{\text{success}}} L_i,$$

where  $L_i$  denotes the trajectory length of the  $i$ -th successful episode.

In addition to success and efficiency, we evaluate the adaptivity of modality selection using **attention entropy (AE)**, which quantifies the degree of uncertainty in the gating controller’s allocation of attention across modalities. Specifically, for a gating distribution  $g_t \in [0, 1]$ , the entropy is computed as:

$$\text{AE} = -\frac{1}{T} \sum_{t=1}^T \left( g_t \log g_t + (1 - g_t) \log(1 - g_t) \right).$$

A lower AE indicates sharper modality preference, while higher AE reflects more balanced modality integration.

Finally, to assess how effectively the agent benefits from intermediate shaping signals, we introduce the **reward density (RD)**, defined as the ratio of non-zero intermediate rewards to the total number of steps in an episode:

$$\text{RD} = \frac{1}{N} \sum_{i=1}^N \frac{M_i}{L_i},$$

where  $M_i$  is the number of steps in which intermediate rewards are triggered for the  $i$ -th trajectory. Together, these four metrics provide a holistic assessment of not only whether the agent can solve tasks, but also how efficiently, adaptively, and effectively it leverages multimodal feedback during training and deployment.

### A.2 DETAILED FRAMEWORK ARCHITECTURE

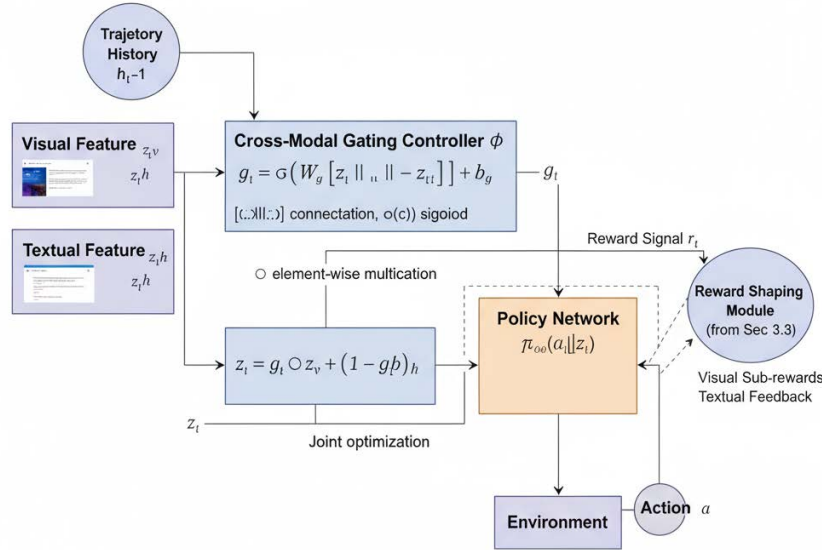


Figure 8: Detailed illustration of the cross-modal attention gating mechanism. The gating controller takes visual features, textual features, and trajectory history as inputs to produce dynamic gating weights that regulate the contribution of each modality in the fused representation.

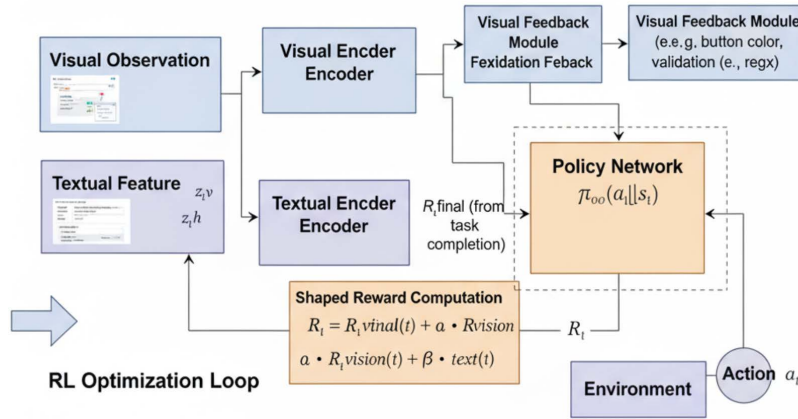


Figure 9: Illustration of the multimodal feedback-driven reward shaping mechanism. The system extracts visual feedback signals (e.g., UI state changes, button color changes) and textual feedback signals (e.g., form field correctness, label matching) to provide dense intermediate rewards alongside terminal task completion rewards.

### A.3 COMPREHENSIVE EXPERIMENTAL RESULTS

Figure 10 illustrates the learning curves across 1000 training episodes, averaged over the three datasets. Several key observations emerge from this analysis. First, reward shaping alone provides

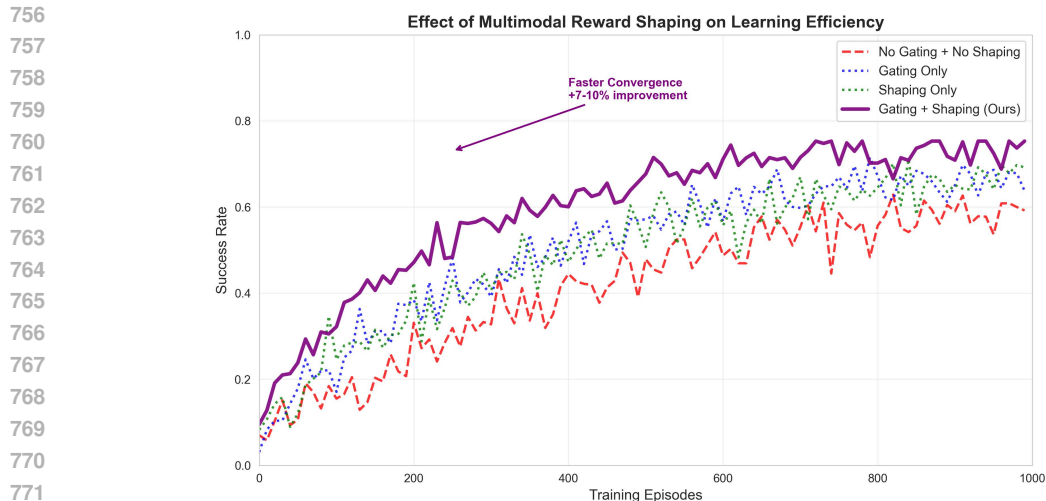


Figure 10: Learning curves showing the effect of different reward configurations on training efficiency. Our full method (Gating + Shaping) achieves faster convergence and higher final performance compared to individual components or baseline methods.

substantial acceleration over the baseline, enabling the agent to reach 70% of its final performance within 360 episodes compared to 450 episodes for the no-shaping baseline. This improvement stems from the dense intermediate feedback that guides exploration toward productive actions, reducing the need for random exploration in sparse-reward environments. Second, gating alone also accelerates learning, though through a different mechanism: by dynamically selecting the most informative modality at each step, the gating controller reduces noise in the state representation, leading to more stable policy updates and faster convergence. Most importantly, the combination of both mechanisms yields the best results, with our full method reaching 75% success rate within just 250 episodes—a 44% reduction in sample complexity compared to the baseline.

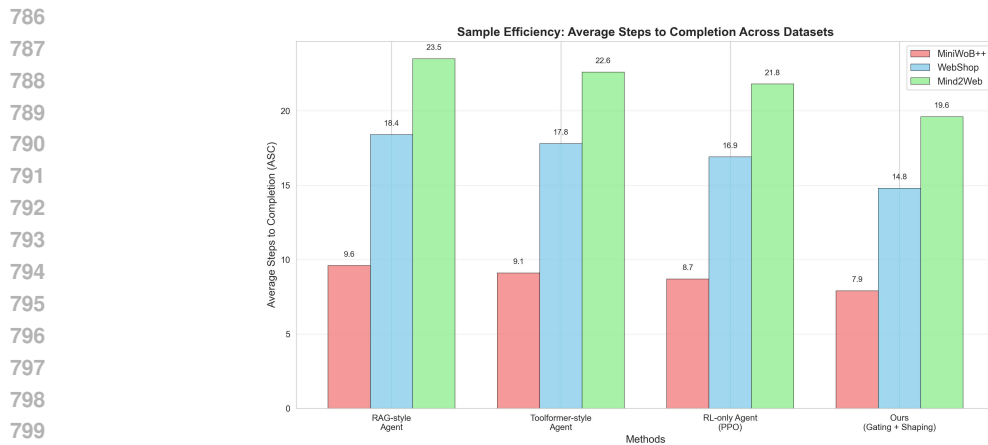


Figure 11: Average steps to completion (ASC) across different datasets and methods. Lower values indicate more efficient task execution. Our method consistently requires fewer steps to complete tasks across all datasets.

#### A.4 COMPONENT ABLATION STUDIES

Figure 12 presents the core ablation results, comparing four configurations: No Gating + No Shaping (baseline), Gating Only, Shaping Only, and Gating + Shaping (full model). The results reveal several critical insights. First, removing gating leads to a significant performance drop across all datasets, with reductions of 3.6% on MiniWoB++, 4.1% on WebShop, and 4.9% on Mind2Web compared to

810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863

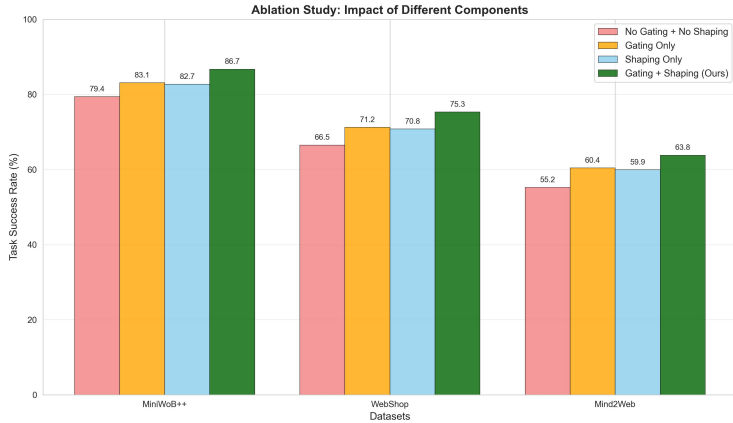


Figure 12: Task Success Rate (TSR) comparison across different component configurations on three datasets. The full model with both gating and shaping consistently outperforms all partial configurations.

the full model. This degradation occurs because rigid fusion fails to adapt to the varying importance of visual versus textual information across different task phases. Without dynamic modality selection, the agent struggles with tasks that require context-sensitive reasoning, such as distinguishing between visually similar buttons or matching form fields with their semantic labels.

### A.5 DYNAMIC MECHANISM ANALYSIS

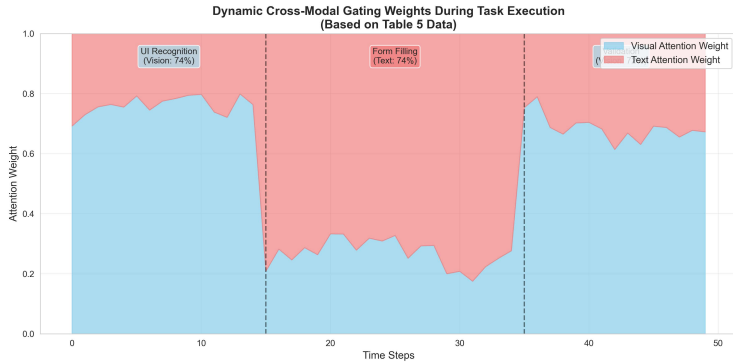


Figure 13: Evolution of cross-modal gating weights during a representative task execution. The gating controller dynamically shifts attention between visual and textual modalities based on task phase requirements.

To understand how the gating mechanism adapts to different task requirements, Figure 13 visualizes the evolution of gating weights  $g_t$  during a representative multi-step task. The visualization reveals three distinct phases with different modality preferences. In the UI Recognition phase (steps 0-15), visual weights dominate (average  $g_t = 0.74$ ), reflecting the importance of spatial layout and visual affordances for element identification. During the Form Filling phase (steps 16-35), the controller shifts toward textual information ( $g_t = 0.26$ ), emphasizing semantic understanding of field labels and content validation. Finally, in the Validation phase (steps 36-50), visual weights increase again ( $g_t = 0.72$ ) as the agent focuses on confirmation dialogs and button states.

Figure 14 tracks the evolution of attention entropy during training, providing insights into how different methods learn to allocate attention across modalities. RAG-style agents maintain consistently high entropy ( $\approx 0.55$ ) throughout training, reflecting their static fusion strategy that treats both modalities equally regardless of context. RL-only agents show gradual entropy reduction as training

864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917

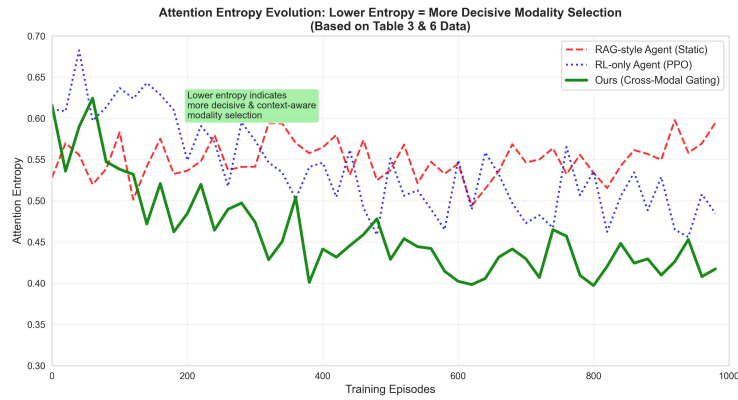


Figure 14: Evolution of attention entropy during training for different methods. Lower entropy indicates more decisive modality selection. Our gating mechanism achieves the lowest entropy, demonstrating confident and context-aware modality choices.

progresses, reaching a final value of  $\approx 0.49$ , indicating some learned preference but without explicit structure for modality selection.

Our proposed gating mechanism demonstrates the most significant entropy reduction, stabilizing at  $\approx 0.35$  after 50K training steps. This lower entropy indicates that our method develops a clear and context-dependent preference for modality selection, dynamically adjusting weights based on task requirements rather than applying fixed or gradually learned strategies.

## A.6 HYPERPARAMETER SENSITIVITY ANALYSIS

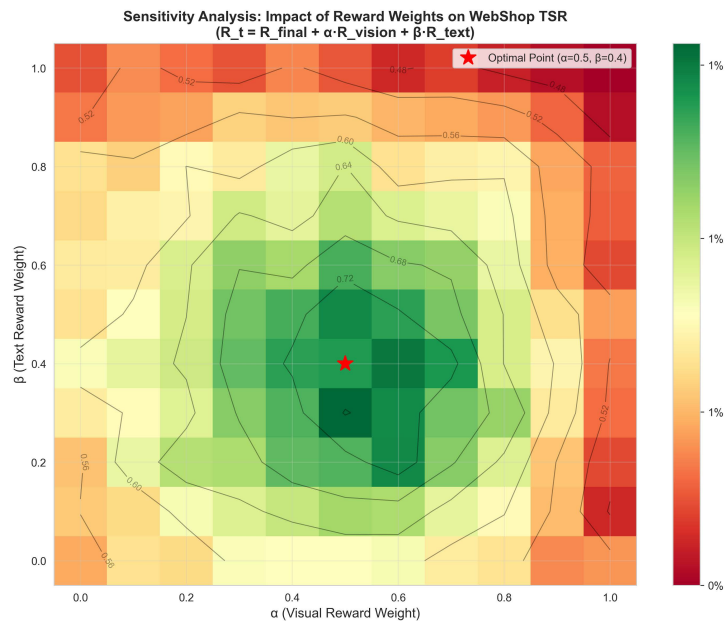


Figure 15: Sensitivity analysis showing the impact of visual reward weight ( $\alpha$ ) and text reward weight ( $\beta$ ) on WebShop task success rate. The optimal configuration balances both modalities with  $\alpha = 0.5$ ,  $\beta = 0.4$ .

The multimodal reward shaping mechanism introduces two hyperparameters:  $\alpha$  (visual reward weight) and  $\beta$  (text reward weight) in the reward equation  $R_t = R_{\text{final}}(t) + \alpha \cdot R_{\text{vision}}(t) + \beta \cdot R_{\text{text}}(t)$ .

918 Figure 15 presents a comprehensive sensitivity analysis over the range  $\alpha, \beta \in [0, 1]$  using WebShop  
919 as the evaluation benchmark.

920  
921 The analysis reveals that both extreme configurations (over-reliance on either modality) lead to  
922 suboptimal performance. Maximum success rates are achieved when both modalities contribute  
923 substantially but not exclusively, with the optimal configuration at  $\alpha = 0.5$  and  $\beta = 0.4$ . This  
924 balanced approach allows the agent to leverage complementary information from both visual and  
925 textual cues while avoiding overfitting to either modality’s specific characteristics. The relatively  
926 flat region around the optimum (success rates above 85% for  $\alpha \in [0.4, 0.6]$  and  $\beta \in [0.3, 0.5]$ )  
927 indicates robustness to small variations in these parameters.

928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971