CATS: Acceptance-Oriented Critical Token Adaptive Selection for Multimodal Speculative Decoding

Kaiwen Liu[1,2] **Yangkai Xie**[1,2] **Shuxia Lin**[1,2] **Chonghan Liu**[3] **Xu Yang**[1,2] **Yiguo Qiao**[1,2]*
[1]School of Computer Science and Engineering, Southeast University, Nanjing, China
[2]Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, China
[3]University of California, Los Angeles
220245043@seu.edu.cn 220242414@seu.edu.cn shuxialin@seu.edu.cn
khazzz1c@gmail.com xuyang_palm@seu.edu.cn yqiao@seu.edu.cn

Abstract

While Speculative Decoding (SD) has become an essential lossless acceleration technique for Large Language Models, its direct application to Multimodal Large Language Models (MLLMs) is hindered by the intricate visual dependencies present in cross-modal generation. Existing SD methods enforce a uniform global alignment between the draft and target models, which proves ineffective in multimodal settings due to a structured distribution bias: deviations of the draft model are systematically concentrated on a sparse subset of tokens that demand fine-grained visual comprehension. To overcome this limitation, we reformulate the training of multimodal draft models as an **acceptance-oriented critical-token alignment problem** and introduce **CATS**, a novel two-stage training framework. CATS employs two complementary selection mechanisms to pinpoint critical tokens: **hard tokens** that exhibit high rejection probabilities, and **vision-critical tokens** whose prediction fundamentally depends on visual semantics. Following an initial global alignment phase, CATS performs sparse, targeted refinement exclusively on the union of these identified token sets. Extensive experiments across three representative benchmarks using multiple LLaVA and Qwen2.5-VL target models demonstrate that our approach consistently improves the token acceptance rate over strong baselines such as EAGLE-2 and supervised fine-tuning. Ultimately, CATS substantially accelerates MLLM inference, achieving speedups of up to $3.17\times$. Our code is available at `https://github.com/ChocoLab-Q/CATS`.

1 Introduction

Multimodal Large Language Models (MLLMs) have achieved remarkable success in image understanding, visual question answering, and OCR [1–6]. However, their inference cost is substantially higher than that of pure-text LLMs due to visual encoding, cross-modal fusion, and long-sequence generation [1, 2, 6]. In the pure-text domain, **Speculative Decoding (SD)** has proven effective in reducing inference latency: a lightweight draft model generates candidate tokens, which are then verified in parallel by the target model [7, 8].

Existing approaches for improving SD, such as SFT [4, 9], knowledge distillation [10–14, 9], and architecture enhancement(e.g., EAGLE [15, 16]), aim to increase token acceptance rates but typically do so indirectly. From the perspective of SD's core metric, acceptance rate is determined by the overlap between the target and draft conditional distributions. Thus, the most direct way to **maximize acceptance** is to explicitly minimize this distributional gap.

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Motivated by SD's success in pure-text settings, researchers directly migrated these methods to multimodal scenarios [17–19]. However, the migration effectiveness falls far short of their pure-text counterparts. The root cause is that discrepancies in pure-text SD stem primarily from language modeling limitations, whereas multimodal draft models must maintain consistency with the target under visual conditioning [1–4, 6, 20]. When generation involves fine-grained visual information, draft models suffer from insufficient visual perception and cross-modal alignment deviations [21, 2–6]. More critically, these deviations systematically concentrate on tokens requiring visual details, OCR, or chart understanding [22–24, 6]. We term this **structured distribution bias** in multimodal SD.

To address the above limitations, we first establish minimizing the conditional distribution gap between the draft and target models as the direct training objective. Through theoretical derivation, this can be realized via forward Kullback–Leibler (KL) minimization as a tractable surrogate. However, structured distribution bias makes uniformly optimizing this objective over all tokens inefficient. Easy-to-learn language tokens dominate training, while critical rejection-causing deviations are overlooked. To this end, we reframe draft model training as an **acceptance-oriented critical-token alignment problem**. Specifically, we characterize critical tokens along two complementary dimensions: tokens with large distributional discrepancies that directly cause rejection are **hard tokens**; tokens heavily rely on visual evidence, where insufficient visual conditioning may cause hallucinations even when the gap is small, are **vision-critical tokens**. Their union constitutes the core of the structured bias.

Based on the above insights, we propose **CATS**, an acceptance-oriented two-stage training framework for multimodal SD. Stage I performs global distribution alignment (e.g., via forward KL minimization). Stage II conducts sparse refinement on the union of hard tokens and vision-critical tokens, enabling targeted correction of structured distribution bias. CATS thus improves token acceptance rates without increasing inference overhead. Our contributions are threefold:

- We explicitly establish that maximizing the acceptance rate in SD is equivalent to minimizing the conditional distribution gap between the draft and target models, with forward KL divergence as a practical surrogate objective.
- Based on the identified structured distribution bias, we design two complementary token selection mechanisms: Hard-Token Dynamic Selection and Vision-Critical Token Selection.
- We propose **CATS**, a two-stage training framework that performs global alignment followed by sparse refinement on the union of critical tokens, achieving efficient and targeted alignment for multimodal SD.

2 Related Work

2.1 Speculative Decoding for MLLMs

SD is a lossless acceleration paradigm for autoregressive generation [7, 8], where a lightweight draft model proposes candidate tokens and a target model verifies them in parallel. Since speedup is largely governed by draft-token acceptance, prior work has focused on improving draft quality and verification efficiency. Medusa [25] introduces multiple prediction heads on the target model, while EAGLE [15] and EAGLE-2 [16] improve draft prediction through target hidden-state modeling and context-aware draft-tree construction. AdaSPEC [26] selectively filters difficult-to-fit positions during knowledge distillation, and Flatter Tokens [27] identifies higher-value supervision through the flatness of target token distributions. These methods are developed for text-only or sample-level draft training; they do not locate the positions within a multimodal generation that are particularly difficult for the draft model according to visual dependence and draft–target mismatch. Recent studies extend SD to MLLMs: DREAM [17] injects refined target features, MASSV [18] adapts small language models into multimodal drafters, SpecFLASH [28] uses latent-guided visual-token compression, and ViSpec [19] enhances vision-aware drafting. However, these methods primarily improve draft architectures or visual feature injection. In contrast, our work adopts an **acceptance-oriented perspective**, targeting the observed concentration of residual draft-target mismatch on sparse hard and vision-critical tokens.

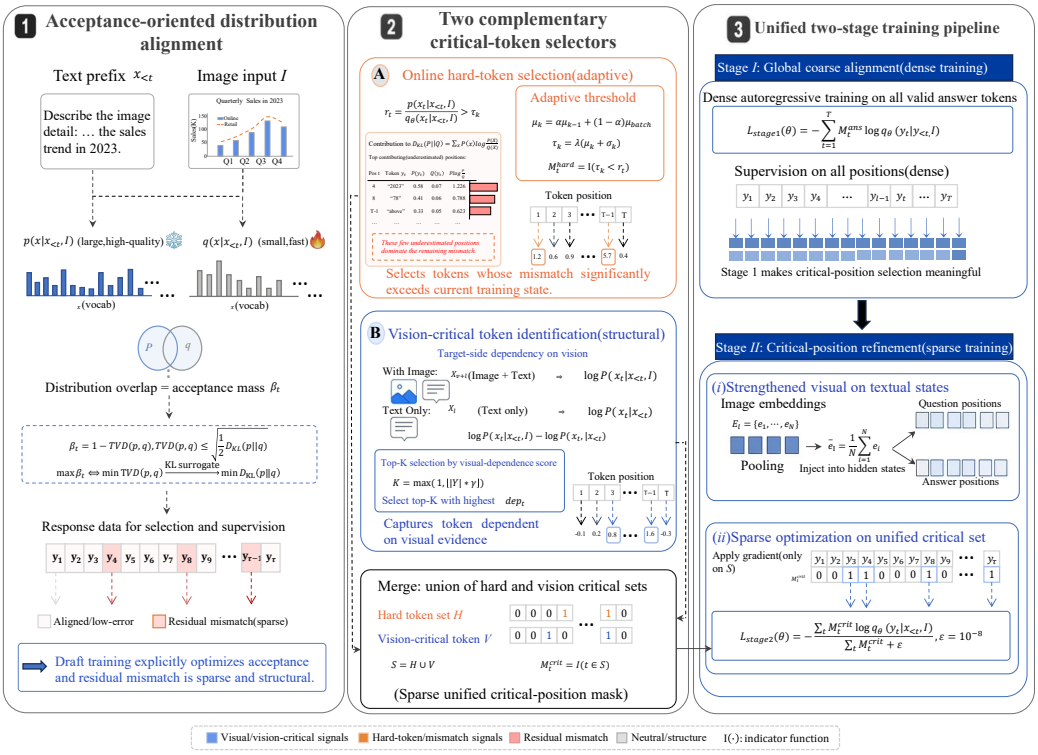

Figure 1: Overview of CATS. Given image I and prefix $x_{<t}$, CATS formulates draft training as acceptance-oriented critical-token alignment. Stage I performs dense global alignment; Stage II applies sparse refinement on the union of hard tokens and vision-critical tokens.

2.2 Training for Draft-Target Alignment in MLLMs

Improving draft-model acceptance has been extensively studied in pure-text speculative decoding. SFT, knowledge distillation, and architectural methods (e.g., EAGLE) globally align the draft distribution with the target, achieving significant progress in text-only scenarios. When directly migrated to multimodal speculative decoding, these methods often underperform, as they treat all tokens uniformly and fail to account for visual dependencies. Separately, token-level visual grounding has been explored to mitigate hallucination in MLLM generation. Visual Contrastive Decoding (VCD) [29] measures visual dependence by comparing target confidence with and without image input, revealing which tokens require visual evidence. OPERA [30] and HA-DPO [31] further improve grounding via logit penalties or preference optimization. AIR [32] selectively compresses visual tokens and reinforces image patches whose representations are aligned with the current multimodal hidden state. However, these techniques are designed for inference-time grounding in general generation, not integrated into draft model training. To our knowledge, no existing approach addresses the structured bias we observe when migrating SD to multimodal scenarios: residual draft-target mismatch systematically concentrates on sparse **hard tokens** and **vision-critical tokens**. This motivates our CATS framework, which jointly handles both types of critical positions through acceptance-oriented sparse refinement.

3 Methodology

We propose CATS, as illustrated in Figure 1, a two-stage training framework for multimodal SD with acceptance-rate maximization as the training objective. Stage I performs global distribution alignment. Stage II refines the draft model only on critical tokens, including hard tokens and vision-critical tokens.

3.1 Acceptance-Oriented Training Objective

SD employs a lightweight draft model to generate candidate tokens, which are then verified in parallel by the target model. The practical speedup is governed not only by the draft model's latency but more fundamentally by how many drafted tokens are accepted. Acceptance therefore becomes the central optimization target for training the draft model.

Let $p(\cdot \mid x_{<t}, I)$ and $q(\cdot \mid x_{<t}, I)$ denote the target and draft conditional distributions at decoding step t, where $x_{<t}$ is the generated prefix and I is the visual input. Under SD, the acceptance rate at step t is

$$\beta_t = \sum_x \min\left\{p(x \mid x_{<t}, I),\ q(x \mid x_{<t}, I)\right\} = 1 - \mathrm{TVD}\big(p(\cdot \mid x_{<t}, I), q(\cdot \mid x_{<t}, I)\big). \tag{1}$$

where TVD denotes total variation distance. Therefore, maximizing acceptance is equivalent to minimizing the conditional distribution gap between the draft and target models.

Directly optimizing TVD is inconvenient in standard autoregressive training. Using Pinsker's inequality,

$$\mathrm{TVD}(p, q) \le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(p \| q)}, \tag{2}$$

reducing the forward KL divergence provides a sufficient optimization direction for improving acceptance.

In practice, the draft model is trained on tokens sampled from the target model p. Under teacher forcing, the standard negative log-likelihood (NLL) loss serves as a Monte Carlo surrogate for the forward KL:

$$\begin{aligned} \nabla_\theta \mathcal{L}_{\mathrm{NLL}} = \nabla_\theta \mathbb{E}_{x \sim p}\left[-\log q_\theta(x)\right] & \\ &= \nabla_\theta \left(H(p) + D_{\mathrm{KL}}(p \| q_\theta)\right) \\ &= \nabla_\theta D_{\mathrm{KL}}(p \| q_\theta). \end{aligned} \tag{3}$$

Thus, training the draft model to predict target-generated tokens is equivalent to minimizing a bound on TVD, thereby indirectly optimizing for acceptance. This equivalence ensures that the structure of the forward KL divergence directly informs the training dynamics induced by NLL-based optimization.

However, conventional NLL training treats all token positions uniformly. After coarse alignment, the remaining draft-target mismatch is not uniformly distributed but concentrated on a sparse subset of critical positions, making uniform optimization inefficient.

3.2 Structured Distribution Bias in Multimodal SD

The above analysis clarifies what to optimize, but not where the optimization budget should be allocated. This becomes important once coarse alignment has been established. Expanding the divergence gives

$$D_{\mathrm{KL}}(P \| Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)}. \tag{4}$$

Hence, tokens with large $P(x)$ but severely underestimated $Q(x)$ contribute disproportionately to the residual alignment error, whereas already aligned tokens contribute little. Under a fixed training budget, continuing to optimize all positions uniformly is therefore inefficient: after coarse alignment, the remaining acceptance bottleneck is dominated by a sparse subset of positions. In multimodal SD, this sparse residual error manifests as a structured distribution bias.

In multimodal generation, this concentration is amplified by the uneven distribution of visual evidence across the response. Many outputs depend on a few visually decisive positions, such as object attributes, OCR content, spatial relations, or chart values. When the draft model fails to ground on the image, it falls back to language priors and generates locally plausible but visually unsupported tokens, which are likely to be rejected during target-side verification. Therefore, the residual mismatch in multimodal SD should not be modeled as uniform token-level noise. Instead, it forms a structured distribution bias concentrated on sparse, visually and semantically important positions.

Table 1: Experiments for structured distribution bias.

Model / dataset	Top-10% KL mass	Top-20% KL mass	Critical-token share	Failure coverage
Qwen2.5-VL-7B / ScienceQA	52.8%	74.0%	20.6%	85.4%
Qwen2.5-VL-7B / A-OKVQA	53.2%	73.1%	19.9%	77.0%
Qwen2.5-VL-7B / TextVQA	46.5%	66.7%	26.9%	81.6%
Qwen2.5-VL-3B / ScienceQA	60.4%	80.7%	14.9%	83.5%
LLaVA-1.5-7B / ScienceQA	70.3%	86.3%	7.1%	79.3%

To verify whether structured distribution bias is present and acceptance-relevant, we perform the following analysis.

Table 1 shows only 7.1%–26.9% of response positions are critical, yet they cover 77.0%–85.4% of top-1 disagreement positions, while the top 10% of positions account for 46.5%–70.3% of the total KL mass. For Qwen2.5-VL-7B, retaining 19.9%–26.9% of positions covers 49.0%–53.5% of aggregate TVD and 67.9%–71.3% of KL mass.

These observations motivate acceptance-rate-driven critical-token alignment: after coarse alignment, training should prioritize positions that contribute most to the structured residual bias. We next introduce two complementary selection mechanisms, one based on current draft–target discrepancy (hard tokens) and the other based on visual dependence (vision-critical tokens).

3.3 Hard-Token Dynamic Selection (HTDS)

After coarse alignment, the remaining optimization budget should be concentrated on positions that dominate the residual structured bias. We therefore introduce an online hard-token selection mechanism to identify such positions during training.

Hardness-relevant positions are those where the draft model still substantially underestimates tokens favored by the target model under the same multimodal context. These positions contribute most directly to the remaining draft-target gap, making them the most informative targets for refinement.

Concretely, let x_t denote a token sampled from current multimodal context $(x_{<t}, I)$. We define the mismatch ratio as

$$r_t = \frac{p(x_t \mid x_{<t}, I)}{q_\theta(x_t \mid x_{<t}, I)}, \tag{5}$$

where p and q_θ are the target and draft conditional distributions. A large ratio indicates that the target assigns high probability to a token that the draft underestimates, serving as a direct proxy for unresolved mismatch. Positions with ratios near one are already well aligned.

Since the scale of mismatch evolves during training, a fixed threshold is unstable. We therefore maintain running statistics of the ratio distribution. At optimization step k, we update the mean via exponential moving average,

$$\mu_k = \alpha\mu_{k-1} + (1-\alpha)\mu_{\text{batch}}, \tag{6}$$

and similarly for the variance, with $\alpha = 0.99$. The selection threshold is

$$\tau_k = \lambda(\mu_k + \sigma_k), \tag{7}$$

where λ controls retention rate which is the fraction of tokens selected for sparse refinement. The hard-token mask is then

$$M_t^{\text{hard}} = \mathbb{I}(r_t > \tau_k). \tag{8}$$

This mechanism dynamically focuses refinement on positions with the largest current draft-target discrepancy. However, mismatch magnitude alone is insufficient in multimodal settings, as some positions depend on visual evidence even when their current mismatch is not yet maximal. We address this next.

3.4 Vision-Critical Token Selection(VTS)

Hard-token selection identifies where the draft is currently most misaligned, but it does not fully capture multimodal difficulty. Some positions are important not because they are currently hard,

but because their prediction intrinsically requires visual grounding. We therefore design a second criterion that measures visual dependence directly from the target model.

Given a multimodal input and target response sequence $Y = \{y_1, \ldots, y_T\}$, we construct two controlled inputs. The first retains both text and image, yielding full multimodal context X_{v+l}. The second removes visual placeholders, yielding text-only context X_l. Both are evaluated under teacher forcing, so each token y_t conditions on the same history $Y_{<t}$ and differs only in the availability of visual evidence. The visual dependence score is defined as

$$\mathrm{dep}_t = \log P(y_t \mid X_{v+l}, Y_{<t}) - \log P(y_t \mid X_l, Y_{<t}). \tag{9}$$

This score measures how much the presence of the image increases the target model's confidence on the same token. A large positive value indicates strong visual dependence, while a value near zero suggests the token can be recovered from text alone.

To obtain a stable token set, we avoid a fixed global threshold, as the scale of visual dependence varies across samples. Instead, we use a relative ranking criterion: given a ratio γ, we retain

$$K = \max(1, \lfloor |Y| \times \gamma \rfloor), \tag{10}$$

tokens with the highest dep_t values and mark them as vision-critical. This yields a sample-adaptive visual mask determined by target-side contrastive analysis, independent of the draft model's state.

Hard tokens capture the error-magnitude dimension of structured bias; vision-critical tokens capture the visual-dependence dimension. Together, they define the critical positions for sparse refinement.

3.5 Unified Two-Stage Training Pipeline

We integrate the above into a two-stage pipeline: Stage I for global coarse alignment, and Stage II for sparse refinement on the union of critical positions.

Coarse Training: Global Coarse Alignment. The goal of this stage is to establish a globally usable draft distribution where critical positions become identifiable. At this point, the draft-target discrepancy is still too diffuse to localize reliably, so dense optimization is necessary. We train the draft model with the standard autoregressive objective over valid response tokens. Let $X = \{x_1, \ldots, x_T\}$ denote the training data and $M_t^{\mathrm{ans}} \in \{0, 1\}$ the loss mask for answer tokens. The objective is

$$\mathcal{L}_{\mathrm{Stage1}}(\theta) = -\sum_{t=1}^{T} M_t^{\mathrm{ans}} \log q_\theta(x_t \mid x_{<t}, I), \tag{11}$$

where q_θ is the draft model. This stage establishes coarse multimodal alignment rather than resolving fine-grained bottlenecks.

Refinement Training: Critical-Position Repair. After coarse training, continuing dense optimization over all positions becomes inefficient, as the remaining error is no longer uniformly distributed but concentrated on the union of hard and vision-critical positions. This stage refines the draft model specifically on this union.

Visual hidden-state enhancement (VHE). As a lightweight optional enhancement before sparse refinement, we inject a pooled visual representation into textual hidden states to provide a visual-conditioning prior. Let $E_I = \{e_1, \ldots, e_N\}$ denote the image embeddings extracted by the visual encoder, and let

$$\bar{e}_I = \frac{1}{N} \sum_{i=1}^{N} e_i \tag{12}$$

be the pooled visual representation. For question positions $t \in \mathcal{Q}$ and answer positions $t \in \mathcal{A}$, we update the hidden states as

$$\tilde{h}_t^{(Q)} = h_t + \eta_Q \bar{e}_I, \qquad t \in \mathcal{Q}, \tag{13}$$

$$\tilde{h}_t^{(A)} = h_t + \eta_A \bar{e}_I, \qquad t \in \mathcal{A}, \tag{14}$$

where h_t is the original hidden state and η_Q, η_A are enhancement coefficients. This operation leaves the critical-position selection criteria unchanged.

We define the active refinement set as the union of hard and vision-critical positions:

$$\mathcal{S} = \mathcal{H} \cup \mathcal{V}, \tag{15}$$

where $\mathcal{H}$ is the hard-token set and $\mathcal{V}$ is the vision-critical set. The mask of this stage is

$$M_t^{\text{crit}} = \mathbb{I}(t \in \mathcal{S}), \tag{16}$$

and positions outside $\mathcal{S}$ receive no gradients. This union captures the two aspects of structured bias, hard tokens reflect its error-magnitude dimension, while vision-critical tokens reflect its visual-dependence dimension.

We use target-sampled tokens as supervision to ensure consistency between selection and refinement. Let y_t denote the token sampled from the target model at position t. The same trajectory is used to estimate the mismatch ratio for hard-token selection and to define the sparse refinement target. The objective is

$$\mathcal{L}_{\text{Stage2}}(\theta) = -\frac{\sum_t M_t^{\text{crit}} \log q_\theta(y_t \mid y_{<t}, I)}{\sum_t M_t^{\text{crit}} + \epsilon}, \qquad \epsilon = 10^{-8}. \tag{17}$$

Thus, CATS first performs dense coarse alignment to make critical positions identifiable, then sparsely refines them to improve acceptance.

4 Experiments

4.1 Experimental Setup

Datasets and Metrics. We evaluate on ScienceQA [33], A-OKVQA [34], and TextVQA [22], covering scientific reasoning, knowledge-based VQA, and text-rich understanding. We report two metrics: speedup ratio (wall-clock time of autoregressive vs. speculative decoding [7]) and average acceptance length τ (accepted draft tokens per verification cycle). All results are reported under temperature 0 and 1.

Baselines. We compare against EAGLE-2 [16] (25 epochs) and SFT (dense supervision, epochs 20–25). Besides EAGLE-2 and SFT, we evaluate several matched-budget selective-training baselines such as Static-HTDS-K, High-NLL-K, AdaSPEC-style-K applies an AdaSPEC [26] filtering rule and ViSpec [19]. CATS-K uses dynamic hard-token selection together with vision-critical token selection. All selective methods start from the same Stage-I checkpoint and Stage-II data, optimizer, training epoch, inference configuration, and fixed per-sample token budget K. The value of K is matched to the retention used by CATS-K.

Implementation Details. Experiments use 4 RTX 4090 and 4 A6000 GPUs. Target models LLaVA-1.5 [35] (7B/13B) and Qwen2.5-VL [20] (3B/7B), frozen during training. All methods share the same EAGLE-2 draft architecture and inference configuration. Stage I trains on 68K LLaVA-Instruct samples [4] for 20 epochs (lr 3×10^{-5}). Stage II trains on each benchmark's training split for 5 epochs (lr 8×10^{-6}).

4.2 Main Results

Table 2 reports speedup α and average acceptance length τ on ScienceQA, A-OKVQA, and TextVQA under two temperatures. CATS consistently achieves the best performance across all target models and tasks. At $T = 0$, CATS achieves speedups of $1.85\times$–$3.17\times$ and the highest τ in all settings. Compared to EAGLE-2 and SFT, CATS brings consistent gains on LLaVA and Qwen2.5-VL families, demonstrating that explicitly aligning critical positions outperforms dense fine-tuning. For instance, on LLaVA-1.5-7B, CATS achieves $2.97\times$, $3.17\times$, and $2.91\times$ speedups on ScienceQA, A-OKVQA, and TextVQA, respectively, significantly outperforming EAGLE-2 ($1.70\times$, $2.12\times$, $2.28\times$) and SFT ($2.00\times$, $2.21\times$, $2.30\times$). A similar trend is observed on LLaVA-1.5-13B, where CATS reaches up to $3.05\times$ on ScienceQA and $3.07\times$ on A-OKVQA.

Task-level comparison shows CATS is particularly effective on visually grounded reasoning and VQA. On ScienceQA and A-OKVQA, CATS achieves strong gains across all model families. For instance, with Qwen2.5-VL-3B at $T = 0$, CATS achieves $2.73\times$ ($\tau = 3.65$) on ScienceQA and $3.15\times$ ($\tau = 4.14$) on A-OKVQA, substantially outperforming baselines. These results support

Table 2: Evaluation of SD methods through speedup ratio α and average accepted token length τ.

Model	Method	Temperature=0						Temperature=1					
		ScienceQA		A-OKVQA		TextVQA		ScienceQA		A-OKVQA		TextVQA	
		τ	α	τ	α	τ	α	τ	α	τ	α	τ	α
LLaVA-1.5 7B	EAGLE-2	2.35	1.70×	2.68	2.12×	2.83	2.28×	2.12	1.57×	2.01	1.48×	2.32	1.80×
	SFT	2.72	2.00×	2.88	2.21×	2.91	2.30×	2.53	1.91×	2.21	1.60×	2.41	1.81×
	CATS(ours)	**3.85**	**2.97**×	**4.12**	**3.17**×	**3.76**	**2.91**×	**3.18**	**2.34**×	**2.66**	**1.91**×	**2.49**	**1.84**×
LLaVA-1.5 13B	EAGLE-2	2.58	1.86×	3.12	2.53×	2.79	1.98×	2.41	1.73×	2.54	1.91×	2.28	1.68×
	SFT	2.83	2.11×	3.18	2.58×	3.05	2.08×	2.47	1.74×	2.61	1.94×	2.34	1.72×
	CATS(ours)	**3.96**	**3.05**×	**3.98**	**3.07**×	**3.32**	**2.38**×	**3.25**	**2.48**×	**2.79**	**2.08**×	**2.47**	**1.83**×
Qwen2.5-VL 3B	EAGLE-2	1.55	1.61×	2.19	1.66×	1.24	1.43×	1.41	1.50×	1.52	1.36×	1.12	1.21×
	SFT	1.99	1.85×	2.31	2.11×	2.04	1.84×	1.46	1.53×	1.34	1.48×	1.53	1.54×
	CATS(ours)	**3.65**	**2.73**×	**4.14**	**3.15**×	**2.98**	**2.25**×	**2.60**	**2.15**×	**2.23**	**2.03**×	**2.11**	**1.83**×
Qwen2.5-VL 7B	EAGLE-2	1.92	1.32×	2.26	1.66×	1.89	1.32×	1.85	1.31×	1.83	1.31×	1.67	1.19×
	SFT	2.24	1.59×	2.45	1.80×	1.97	1.37×	2.04	1.45×	1.92	1.38×	1.72	1.22×
	CATS(ours)	**3.21**	**2.50**×	**3.42**	**2.65**×	**2.51**	**1.85**×	**2.74**	**2.05**×	**2.27**	**1.68**×	**2.09**	**1.51**×

Table 3: Matched-budget selective training.

Model	Method	Temperature=0
		ScienceQA
		α
Qwen2.5-VL 7B	Static-HTDS-K	2.51×
	High-NLL-K	2.49×
	AdaSPEC-style-K	2.44×
	CATS-K (ours)	**2.62**×

Table 4: Cross-task comparison with ViSpec.

Model	Method	Temperature=0	
		ScienceQA	COCO Caption
		α	α
Qwen2.5-VL 7B	ViSpec	1.912×	1.944×
	CATS (ours)	**2.565**×	**2.070**×

that multimodal SD is limited by structured draft-target mismatch, not uniform token errors. By selecting hard tokens (large residual mismatch) and vision-critical tokens (high visual dependence), CATS refines positions most critical for target-side verification. In contrast, TextVQA shows lower speedups, e.g., $1.85\times$ on Qwen2.5-VL-7B at $T = 0$, suggesting OCR-oriented generation remains more challenging due to stricter visual grounding requirements.

Temperature has a clear impact on SD performance. Under deterministic decoding ($T = 0$), CATS achieves longer acceptance spans and higher speedups, as draft and target distributions are easier to align along a stable trajectory. When sampling is enabled ($T = 1$), increased token variance reduces τ and speedup for all methods, yet CATS remains consistently better than baselines. For example, on LLaVA-1.5-13B at $T = 1$, CATS still achieves $2.48\times$, $2.08\times$, and $1.83\times$ speedups on ScienceQA, A-OKVQA, and TextVQA, respectively, outperforming both EAGLE-2 and SFT. These results indicate that the proposed critical-position refinement remains robust under stochastic decoding, though its advantage is more pronounced in the deterministic setting.

4.3 Broader Baseline Comparisons

Matched-budget selective training. Under a matched per-sample token budget, CATS-K achieves the highest speedup among the selective-training baselines, exceeding AdaSPEC-style-K by $0.18\times$ (Table 3). Shared backbones, training budgets, and inference configurations isolate the contribution of dynamic mismatch selection and vision-critical token selection.

Cross-task comparison with ViSpec. We compare CATS with the visual-feature-based ViSpec [19] under cross-task transfer. On ScienceQA and COCO Caption, CATS improves speedup by $0.653\times$ and $0.126\times$, respectively (Table 4); average output lengths are 56.4 and 158.6 tokens. This complements the matched-budget comparison by evaluating an architectural and visual-feature alternative.

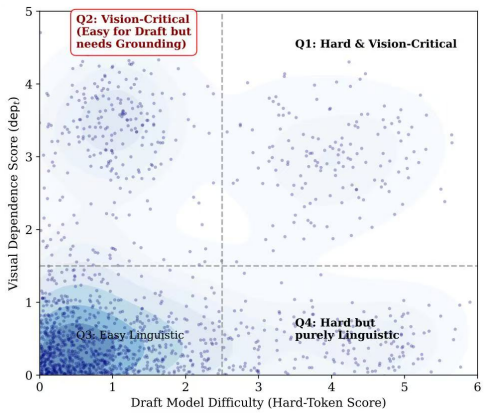

Figure 2: Token criticality along two dimensions. Four regimes: hard + vision-critical (Q1), vision-critical only (Q2), easy linguistic (Q3), and hard but linguistic (Q4).

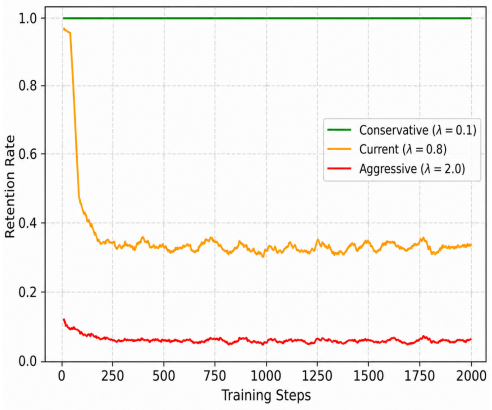

Figure 3: Retention dynamics of online hard-token selection. Too low a threshold degenerates to dense training; too high a threshold discards useful errors.

4.4 Complementarity of Hard-Token and Vision-Critical Selection

Table 5: Distribution of answer positions, residual KL mass, and failure coverage across hard-token and vision-critical regions.

Region	Token share	KL mass	Failure coverage
$\mathcal{H} \cap \mathcal{V}$	2.25%	6.91%	9.30%
H-only	18.39%	59.07%	76.08%
V-only	7.88%	3.48%	1.62%
Neither	71.49%	30.53%	13.00%

We partition answer positions according to the hard-token mask $\mathcal{H}$ and the vision-critical mask $\mathcal{V}$. The intersection $\mathcal{H} \cap \mathcal{V}$ covers only 2.25% of answer positions and 6.91% of the residual KL mass, showing that an intersection-only policy is too restrictive. In contrast, H-only positions account for 59.07% of the KL mass and 76.08% of failure coverage, while V-only positions provide a distinct visual-dependence signal that is not captured by current mismatch ranking. Under the same ScienceQA evaluation, $\mathcal{H} \cap \mathcal{V}$, H-only, and $\mathcal{H} \cup \mathcal{V}$ achieve speedups of $1.759\times$, $2.326\times$, and $2.565\times$, respectively. These results support using the union to combine realized draft–target mismatch with structural visual dependence. Further selector-wise results are provided in Appendix B.5.

4.5 Ablation Study

Impact of hard-token dynamic selection. We first evaluate HTDS, which focuses on positions with substantial draft–target discrepancy. Table 6 shows HTDS brings the largest single-step improvement over Stage I on both LLaVA-1.5-7B and Qwen2.5-VL-7B. For instance, on ScienceQA at $T = 0$, HTDS improves speedup from $1.69\times$ to $2.56\times$ (LLaVA-1.5-7B) and from $1.39\times$ to $2.23\times$ (Qwen2.5-VL-7B). This indicates that after coarse alignment, acceptance is limited by a sparse set of high-mismatch positions rather than a global distribution gap. Figure 3 shows that a moderate threshold ($\lambda = 0.8$) yields appropriate sparsity, filtering out well-aligned tokens while preserving sufficient critical positions.

Impact of vision-critical token selection. We then evaluate VTS, which targets positions by target-side visual dependence rather than current draft error. Adding VTS to HTDS consistently improves performance. On TextVQA at $T = 0$, HTDS+VTS increases speedup from $2.79\times$ to $2.89\times$ on LLaVA-1.5-7B and from $1.71\times$ to $1.84\times$ on Qwen2.5-VL-7B. This improvement matters because OCR-style and visually grounded tokens may not have the largest immediate mismatch, yet

Table 6: Ablation study of CATS. Comparison over different model architectures through speedup ratio α and average accepted token length τ.

Model	Method	Temperature=0						Temperature=1					
		ScienceQA		A-OKVQA		TextVQA		ScienceQA		A-OKVQA		TextVQA	
		τ	α	τ	α	τ	α	τ	α	τ	α	τ	α
LLaVA-1.5 7B	Stage1	2.19	1.69×	3.39	2.61×	2.93	2.27×	2.12	1.56×	2.49	1.79×	2.31	1.71×
	HTDS	3.32	2.56×	3.68	2.83×	3.60	2.79×	2.84	2.09×	2.41	1.73×	2.35	1.74×
	HTDS+VTS	3.59	2.77×	3.87	2.98×	3.73	2.89×	2.99	2.20×	2.51	1.80×	2.41	1.80×
	HTDS+VHE	3.34	2.58×	4.04	3.11×	3.70	2.86×	2.87	2.11×	2.56	1.84×	2.37	1.75×
	HTDS+VHE+VTS	**3.85**	**2.97**×	**4.12**	**3.17**×	**3.76**	**2.91**×	**3.18**	**2.34**×	**2.66**	**1.91**×	**2.49**	**1.84**×
Qwen2.5-VL 7B	Stage1	1.78	1.39×	2.14	1.66×	1.79	1.32×	1.79	1.34×	1.85	1.37×	1.58	1.14×
	HTDS	2.86	2.23×	3.23	2.50×	2.32	1.71×	2.11	1.58×	2.22	1.64×	2.06	1.49×
	HTDS+VTS	3.04	2.37×	3.41	2.64×	2.50	1.84×	2.67	2.00×	2.23	1.65×	1.97	1.42×
	HTDS+VHE	2.98	2.32×	3.38	2.62×	2.43	1.79×	2.55	1.91×	2.20	1.63×	2.01	1.45×
	HTDS+VHE+VTS	**3.21**	**2.50**×	**3.42**	**2.65**×	**2.51**	**1.85**×	**2.74**	**2.05**×	**2.27**	**1.68**×	**2.09**	**1.51**×

become fragile if visual evidence is missing. VTS thus complements HTDS by capturing structural multimodal difficulty that mismatch-only selection misses.

Figure 2 illustrates this complementarity. Hard tokens and vision-critical tokens are correlated but not identical. Some tokens fall into the high-hardness and high-visual-dependence region where both selectors agree. Others are visually dependent but only moderately hard, or hard but weakly visual-dependent. This distinction motivates using their union: HTDS targets the error-magnitude dimension of structured bias, while VTS targets the visual-grounding dimension. Together, they cover more rejection-prone positions than either criterion alone.

Impact of visual hidden-state enhancement. We further evaluate VHE, which injects pooled visual representations into textual hidden states before sparse refinement. Unlike HTDS, VHE does not alter token selection but enriches the visual context for selected tokens. Gains are moderate but consistent on visually grounded tasks. For example, on A-OKVQA at $T = 0$, adding VHE to HTDS improves speedup from $2.83\times$ to $3.11\times$ on LLaVA-1.5-7B and from $2.50\times$ to $2.62\times$ on Qwen2.5-VL-7B. This suggests sparse refinement benefits from stronger visual conditioning: a compact visual prior helps align the draft model with image-conditioned target behavior. The smaller gains at $T = 1$ also indicate VHE primarily aids deterministic visual grounding rather than sampling-induced variance.

Finally a complete dense-versus-sparse refinement comparison is provided in Appendix B.4. Under the same Stage-II refinement budget, sparse refinement consistently achieves higher speedup than dense refinement across both model families and all evaluated tasks.

5 Conclusion

We propose CATS, an acceptance-oriented two-stage training framework that performs global coarse alignment followed by sparse refinement on the union of hard tokens and vision-critical tokens. Experiments on multiple MLLM backbones and multimodal benchmarks show that CATS consistently improves average acceptance length and speedup over strong baselines, demonstrating the effectiveness of acceptance-oriented sparse refinement for fast multimodal generation. Despite its efficiency gains, CATS has been primarily validated on limited tasks and model families. Future work will extend it to broader scenarios, diverse architectures, and real-world deployments, while addressing training efficiency and safety considerations.

Acknowledgments and Disclosure of Funding

This research was supported by the National Natural Science Foundation of China (No. 62306072). We thank the EAGLE team for making their code publicly available. We also thank the creators and maintainers of the datasets and pretrained models for making these resources publicly available.

References

[1] Jean-Baptiste Alayrac et al. Flamingo: A visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, 2022.

[2] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, pages 19730–19742. PMLR, 2023.

[3] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pages 49250–49267, 2023.

[4] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916, 2023.

[5] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[6] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *CoRR*, abs/2308.12966, 2023.

[7] Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*, pages 19274–19286. PMLR, 2023.

[8] Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. Accelerating large language model decoding with speculative sampling. *CoRR*, abs/2302.01318, 2023.

[9] Yongchao Zhou, Kaifeng Lyu, Ankit Singh Rawat, Aditya Krishna Menon, Afshin Rostamizadeh, Sanjiv Kumar, Jean-François Kagy, and Rishabh Agarwal. Distillspec: Improving speculative decoding via knowledge distillation. In *International Conference on Learning Representations*, 2024.

[10] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[11] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[12] Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling bert for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, 2020.

[13] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems*, volume 33, pages 5776–5788, 2020.

[14] Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, 2024.

[15] Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. EAGLE: Speculative sampling requires rethinking feature uncertainty. In *International Conference on Machine Learning*. PMLR, 2024.

[16] Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. EAGLE-2: Faster inference of language models with dynamic draft trees. In *Empirical Methods in Natural Language Processing*, 2024.

[17] Yunhai Hu, Tianhua Xia, Zining Liu, Rahul Raman, Xingyu Liu, Bo Bao, Eric Sather, Vithursan Thangarasa, and Sai Qian Zhang. DREAM: Drafting with refined target features and entropy-adaptive cross-attention fusion for multimodal speculative decoding. *CoRR*, abs/2505.19201, 2025.

[18] Mugilan Ganesan, Shane Segal, Ankur Aggarwal, Nish Sinnadurai, Sean Lie, and Vithursan Thangarasa. MASSV: Multimodal adaptation and self-data distillation for speculative decoding of vision-language models. In *EMNLP (Findings)*, 2025.

[19] Jialiang Kang, Han Shu, Wenshuo Li, Yingjie Zhai, and Xinghao Chen. ViSpec: Accelerating vision-language models with vision-aware speculative decoding. In *Advances in Neural Information Processing Systems*, 2025.

[20] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *CoRR*, abs/2502.13923, 2025.

[21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[22] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8317–8326, 2019.

[23] Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021.

[24] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *ACL (Findings)*, 2022.

[25] Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D. Lee, Deming Chen, and Tri Dao. Medusa: Simple LLM inference acceleration framework with multiple decoding heads. In *International Conference on Machine Learning*. PMLR, 2024.

[26] Yuezhou Hu, Jiaxin Guo, Xinyu Feng, and Tuo Zhao. Adaspec: Selective knowledge distillation for efficient speculative decoders. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-2967. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/801a6708014185e951a95108b8cc8349-Abstract-Conference.html.

[27] Jiaming Fan, Daming Cao, Xiangzhong Luo, Jiale Fu, Chonghan Liu, and Xu Yang. Flatter tokens are more valuable for speculative draft model training. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/97d596ca21d0751ba2c633bad696cf7f-Abstract-Conference.html.

[28] Zihua Wang, Ruibo Li, Haozhe Du, Joey Tianyi Zhou, Yu Zhang, and Xu Yang. SpecFLASH: A latent-guided semi-autoregressive speculative decoding framework for efficient multimodal generation, 2025. URL https://arxiv.org/abs/2505.12728.

[29] Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13872–13882, 2024.

[30] Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. OPERA: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13418–13427, 2024.

[31] Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiaoyi Dong, Jiaqi Wang, and Conghui He. Beyond hallucinations: Enhancing LVLMs through hallucination-aware direct preference optimization. *CoRR*, abs/2311.16839, 2023.

[32] Xingyu Zhu, Kesen Zhao, Liang Yi, Shuo Wang, Zhicai Wang, Beier Zhu, Hanwang Zhang, and Xiangnan He. Look carefully: Adaptive visual reinforcements in multimodal large language models for hallucination mitigation. In *International Conference on Learning Representations*, 2026. URL `https://proceedings.iclr.cc/paper_files/paper/2026/hash/83749a89d03df30c66af88020e4c7892-Abstract-Conference.html`.

[33] Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems*, volume 35, pages 2507–2521, 2022.

[34] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *European Conference on Computer Vision*, pages 146–162. Springer, 2022.

[35] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024.

[36] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4971–4980, 2018.

Disclosure of LLM Usage

During manuscript preparation, we used a general-purpose large language model tool only for minor copyediting (instruction-based rewriting for grammar and clarity). The tool was not used to generate scientific content, results, code, or citations. All technical statements, method designs, and empirical conclusions were produced by the authors and verified by us; we take full responsibility for the content. No confidential, unpublished, or review-related materials were shared with the tool.

A Theoretical Derivation

This section provides the theoretical derivations supporting the acceptance-oriented training principle and the critical-position refinement strategy used in CATS. We first show that target-supervised NLL training minimizes the forward KL divergence between the target and draft conditional distributions, which provides a tractable surrogate for improving the acceptance rate in speculative decoding. We then decompose the residual alignment error over token positions and show why sparse refinement should prioritize positions with large draft–target mismatch. Finally, we justify the two token selectors used in CATS: hard-token selection as an online proxy for per-position KL contribution, and vision-critical token selection as a complementary proxy for structural visual dependence.

A.1 Notation and Setup

Let p denote the target model and q_θ denote the draft model parameterized by θ. At decoding step t, given the prefix $x_{<t}$ and visual input I, we define the conditional distributions

$$p_t(\cdot) := p(\cdot \mid x_{<t}, I), \qquad q_t(\cdot) := q_\theta(\cdot \mid x_{<t}, I). \tag{18}$$

Unless otherwise specified, expectations over prefixes are taken with respect to target-generated trajectories. This matches the Stage II training setting in CATS, where the draft model is supervised on tokens sampled from or generated by the target model.

A.2 Target-Supervised NLL as Forward KL Minimization

We first formalize why training the draft model on target-generated tokens provides an acceptance-oriented alignment objective.

Theorem 1 (NLL equals forward KL up to a constant)**.** For any fixed prefix $x_{<t}$, the cross-entropy between the target conditional distribution p_t and the draft conditional distribution q_t satisfies

$$H(p_t, q_t) = H(p_t) + D_{\mathrm{KL}}(p_t \| q_t). \tag{19}$$

Therefore, the conditional NLL objective

$$\mathcal{L}_{\mathrm{NLL}}(\theta) = \mathbb{E}_{x_{<t}\sim p,\, x_t \sim p_t}\left[-\log q_\theta(x_t \mid x_{<t}, I)\right] \tag{20}$$

can be written as

$$\mathcal{L}_{\mathrm{NLL}}(\theta) = \mathbb{E}_{x_{<t}\sim p}\left[H(p_t) + D_{\mathrm{KL}}(p_t \| q_t)\right]. \tag{21}$$

Since $H(p_t)$ is independent of θ, minimizing $\mathcal{L}_{\mathrm{NLL}}$ is equivalent to minimizing

$$\mathbb{E}_{x_{<t}\sim p}\left[D_{\mathrm{KL}}(p_t \| q_t)\right]. \tag{22}$$

Proof. For a fixed prefix $x_{<t}$, we have

$$H(p_t, q_t) = -\sum_x p_t(x) \log q_t(x). \tag{23}$$

Adding and subtracting $\log p_t(x)$ gives

$$H(p_t, q_t) = -\sum_x p_t(x) \log p_t(x) + \sum_x p_t(x) \log \frac{p_t(x)}{q_t(x)}. \tag{24}$$

The first term is $H(p_t)$, and the second term is $D_{\mathrm{KL}}(p_t \| q_t)$. Taking expectation over target prefixes yields the result.

This theorem shows that target-supervised NLL does not merely improve generic next-token prediction. Under target-generated supervision, it directly minimizes the forward KL divergence from the target conditional distribution to the draft conditional distribution, which is the divergence most relevant to covering the target model's high-probability mass.

A.3 Acceptance Rate and Total Variation Distance

We next relate distributional alignment to the token-level acceptance rate in speculative decoding.

Definition 1 (Per-step acceptance rate)**.** At step t, the speculative decoding acceptance rate between the target distribution p_t and the draft distribution q_t is

$$\beta_t = \sum_x \min\{p_t(x), q_t(x)\}. \tag{25}$$

Lemma 1 (Acceptance rate equals one minus TVD)**.** Let

$$\mathrm{TVD}(p_t, q_t) = \frac{1}{2} \sum_x |p_t(x) - q_t(x)| \tag{26}$$

be the total variation distance. Then

$$\beta_t = 1 - \mathrm{TVD}(p_t, q_t). \tag{27}$$

Proof. Using

$$\min\{a, b\} = \frac{a + b - |a - b|}{2}, \tag{28}$$

we obtain

$$\begin{aligned} \beta_t &= \sum_x \min\{p_t(x), q_t(x)\} \\ &= \frac{1}{2} \sum_x \left(p_t(x) + q_t(x) - |p_t(x) - q_t(x)|\right) \\ &= 1 - \frac{1}{2} \sum_x |p_t(x) - q_t(x)| \\ &= 1 - \mathrm{TVD}(p_t, q_t). \end{aligned} \tag{29}$$

Lemma 1 shows that maximizing token-level acceptance is exactly equivalent to minimizing the total variation distance between the target and draft conditional distributions at the same decoding state.

A.4 From Forward KL to Acceptance Lower Bound

Directly optimizing TVD is inconvenient in standard autoregressive training. We therefore use the forward KL divergence as a tractable surrogate.

Theorem 2 (Forward KL controls acceptance through Pinsker's inequality)**.** For each decoding step t,

$$\mathrm{TVD}(p_t, q_t) \le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(p_t \| q_t)}. \tag{30}$$

Consequently,

$$\beta_t \ge 1 - \sqrt{\frac{1}{2} D_{\mathrm{KL}}(p_t \| q_t)}. \tag{31}$$

Taking expectation over decoding steps gives

$$\mathbb{E}[\beta_t] \ge 1 - \sqrt{\frac{1}{2} \mathbb{E}\left[D_{\mathrm{KL}}(p_t \| q_t)\right]}. \tag{32}$$

Proof. Pinsker's inequality gives

$$\mathrm{TVD}(p_t, q_t) \le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(p_t \| q_t)}. \tag{33}$$

Combining this with Lemma 1, we have

$$\beta_t = 1 - \mathrm{TVD}(p_t, q_t) \ge 1 - \sqrt{\frac{1}{2} D_{\mathrm{KL}}(p_t \| q_t)}. \tag{34}$$

Taking expectation over t, we obtain

$$\mathbb{E}[\mathrm{TVD}(p_t, q_t)] \le \mathbb{E}\left[\sqrt{\frac{1}{2} D_{\mathrm{KL}}(p_t \| q_t)}\right]. \tag{35}$$

Since $\sqrt{\cdot}$ is concave, Jensen's inequality gives

$$\mathbb{E}\left[\sqrt{\frac{1}{2} D_{\mathrm{KL}}(p_t \| q_t)}\right] \le \sqrt{\frac{1}{2}\mathbb{E}\left[D_{\mathrm{KL}}(p_t \| q_t)\right]}. \tag{36}$$

Therefore,

$$\mathbb{E}[\beta_t] = 1 - \mathbb{E}[\mathrm{TVD}(p_t, q_t)] \ge 1 - \sqrt{\frac{1}{2}\mathbb{E}\left[D_{\mathrm{KL}}(p_t \| q_t)\right]}. \tag{37}$$

Theorem 2 provides the theoretical basis for acceptance-oriented draft-model training: minimizing target-supervised NLL reduces the expected forward KL divergence, which tightens an upper bound on TVD and improves a lower bound on the acceptance rate.

A.5 Token-Level Residual Alignment Error

The previous results explain what divergence should be reduced. We now analyze where the remaining optimization budget should be allocated after coarse alignment.

For a sequence of length T, define the expected residual alignment error as

$$\mathcal{R}(\theta) = \mathbb{E}_t\left[D_{\mathrm{KL}}(p_t \| q_t)\right]. \tag{38}$$

Equivalently, for a finite sequence,

$$\mathcal{R}(\theta) = \frac{1}{T}\sum_{t=1}^{T} \Delta_t, \qquad \Delta_t := D_{\mathrm{KL}}(p_t \| q_t). \tag{39}$$

Here, Δ_t measures the residual draft–target mismatch at position t.

Expanding the per-position KL term gives

$$\Delta_t = \sum_x p_t(x) \log \frac{p_t(x)}{q_t(x)}. \tag{40}$$

Thus, positions where the target assigns high probability to tokens that are underestimated by the draft contribute disproportionately to the residual alignment error.

After Stage I coarse alignment, many positions become reasonably aligned and have small Δ_t. Continuing to optimize all positions uniformly therefore becomes inefficient under a fixed training budget. If only a subset $\mathcal{S}$ of positions can be emphasized during refinement, an ideal budget allocation would prioritize positions with large residual contributions:

$$\mathcal{S}^\star = \arg\max_{\mathcal{S}:\,|\mathcal{S}|\le K} \sum_{t\in\mathcal{S}} \Delta_t. \tag{41}$$

However, directly computing the full Δ_t requires evaluating the complete target and draft distributions over the vocabulary at every position, which is expensive and unstable during online training. CATS therefore uses two complementary proxies to approximate acceptance-critical positions: hard-token selection estimates the current draft–target mismatch, while vision-critical token selection estimates structural dependence on visual evidence.

A.6 Hard Tokens as Proxies for Per-Position KL

Hard-token selection is designed to identify positions where the current draft model significantly underestimates target-preferred tokens.

Let $x_t \sim p_t$ be a token sampled from the target model. Define the log mismatch ratio

$$\ell_t(x_t) = \log \frac{p_t(x_t)}{q_t(x_t)}. \tag{42}$$

Then $\ell_t(x_t)$ provides an unbiased sample estimate of the per-position forward KL.

Lemma 2 (Target-sampled log-ratio estimates per-position KL)**.** For $x_t \sim p_t$,

$$\mathbb{E}_{x_t \sim p_t}\left[\log \frac{p_t(x_t)}{q_t(x_t)}\right] = D_{\mathrm{KL}}(p_t \| q_t). \tag{43}$$

Proof. By definition,

$$\mathbb{E}_{x_t \sim p_t}\left[\log \frac{p_t(x_t)}{q_t(x_t)}\right] = \sum_x p_t(x) \log \frac{p_t(x)}{q_t(x)} = D_{\mathrm{KL}}(p_t \| q_t). \tag{44}$$

Lemma 2 justifies using target-sampled mismatch as an online proxy for the residual alignment error. In practice, CATS uses the ratio

$$r_t = \frac{p_t(x_t)}{q_t(x_t)} \tag{45}$$

for hard-token selection. Since r_t is a monotonic transformation of $\ell_t = \log r_t$, thresholding or ranking by r_t is equivalent to thresholding or ranking by ℓ_t. Large values of r_t indicate that the target assigns substantially higher probability to the sampled token than the draft, suggesting that the position contributes strongly to the remaining forward KL and is likely to reduce acceptance if left uncorrected.

Because the scale of r_t changes during training, CATS maintains running statistics over the mismatch ratio. At optimization step k, the mean and variance are updated using exponential moving averages:

$$\mu_k = \alpha \mu_{k-1} + (1-\alpha)\mu_{\mathrm{batch}}, \tag{46}$$

with the variance updated analogously. This adaptive rule selects positions whose current mismatch is significantly larger than the training-time baseline, thereby allocating sparse refinement to positions with large realized draft–target discrepancy.

A.7 Vision-Critical Tokens as Structural Visual-Dependence Proxies

Hard-token selection captures the current error-magnitude dimension of residual mismatch. However, in multimodal generation, some positions are important not only because they currently have large draft–target mismatch, but because their prediction intrinsically depends on visual evidence. These positions may become acceptance-critical even when their instantaneous mismatch is not among the largest, especially if the draft model tends to rely on language priors rather than visual grounding.

To identify such positions, CATS measures the target model's dependence on the image. Given a multimodal input and a target response sequence

$$Y = \{y_1, \ldots, y_T\}, \tag{47}$$

we construct two controlled contexts. The full multimodal context X_{v+l} retains both image and text, while the degraded language-only context X_l removes the visual input or visual placeholders. Both contexts are evaluated under teacher forcing, so each token y_t is conditioned on the same autoregressive history $Y_{<t}$. The visual-dependence score is

$$\mathrm{dep}_t = \log P(y_t \mid X_{v+l}, Y_{<t}) - \log P(y_t \mid X_l, Y_{<t}). \tag{48}$$

A large positive dep_t means that the presence of visual evidence substantially increases the target model's confidence in token y_t. Such tokens often correspond to visually grounded information, including object attributes, OCR content, spatial relations, chart values, or fine-grained image details.

Unlike hard-token selection, which depends on the current draft model q_θ, the visual-dependence score is computed from the target model and reflects structural multimodal sensitivity. Therefore, it complements hard-token selection: **HTDS:** current draft–target mismatch; **VTS:** target-side visual dependence. For stability across samples and tasks, CATS uses a relative ranking rule. Given a selection ratio γ, it keeps

$$K = \max(1, \lfloor |Y|\gamma \rfloor) \tag{49}$$

tokens with the largest dep_t values and marks them as vision-critical.

A.8 Union-Based Sparse Refinement

Let $\mathcal{H}$ denote the set of hard tokens selected by HTDS and $\mathcal{V}$ denote the set of vision-critical tokens selected by VTS. CATS defines the sparse refinement set as

$$\mathcal{S} = \mathcal{H} \cup \mathcal{V}. \tag{50}$$

The corresponding critical-position mask is

$$M_t^{\text{crit}} = \mathbb{I}(t \in \mathcal{S}). \tag{51}$$

This union is conservative and acceptance-oriented. The hard-token set $\mathcal{H}$ targets positions with large realized draft–target discrepancy, which directly approximates large per-position KL contributions. The vision-critical set $\mathcal{V}$ targets positions whose prediction strongly depends on visual evidence, which captures multimodal sensitivity that may not be fully exposed by the current draft mismatch alone. Refining their union therefore covers both dimensions of the structured residual bias:

$$\underbrace{\mathcal{H}}_{\text{error-magnitude dimension}} \quad \cup \quad \underbrace{\mathcal{V}}_{\text{visual-dependence dimension}} . \tag{52}$$

The Stage II objective is

$$\mathcal{L}_{\text{Stage2}}(\theta) = -\frac{\sum_t M_t^{\text{crit}} \log q_\theta(y_t \mid y_{<t}, I)}{\sum_t M_t^{\text{crit}} + \epsilon}, \qquad \epsilon = 10^{-8}. \tag{53}$$

This objective concentrates the optimization budget on positions most relevant to acceptance improvement and multimodal grounding, rather than continuing to update all positions uniformly after coarse alignment.

B Training Details

B.1 Distillation Data Construction

Teacher models and decoding. For each downstream task, we distill training data using the corresponding frozen target vision-language model as the teacher. Given an image–question pair, the teacher generates the response as the hard label for draft-model training. For prompt formatting, we follow the chat template of the corresponding teacher model and adapt the user query according to the task format. For ScienceQA, we construct a multiple-choice reasoning prompt with the question and candidate choices, and ask the teacher to output both the answer and a concise reasoning process. For A-OKVQA and TextVQA, we append an OCR-oriented instruction to the original question, encouraging the teacher to extract visual text accurately and produce a structured response. After generation, the prompt, teacher response, and end-of-sequence token are concatenated into the full training sequence, while the loss mask is applied only to the teacher-generated response tokens.

Dataset splits for efficiency. To balance data coverage and evaluation efficiency, we construct distillation data on three representative multimodal question-answering benchmarks: ScienceQA, A-OKVQA, and TextVQA. Specifically, we use 12,726 training examples from ScienceQA, 17,056 training examples from A-OKVQA, and 34,602 training examples from TextVQA for distillation. For evaluation, we uniformly sample 500 examples from the test split of each dataset, resulting in comparable evaluation cost across tasks while preserving task diversity. These three datasets cover complementary multimodal reasoning scenarios, including science-oriented visual reasoning, open-domain visual question answering, and text-centric visual understanding.

B.2 Hyperparameter Sensitivity Analysis

We analyze the sensitivity of EMA momentum α, the vision-critical token selection ratio γ, and the coefficients and aggregation strategies used for hidden-state enhancement. Extended training is evaluated separately in Section B.3.3. The purpose of this analysis is not to tune hyperparameters separately for each benchmark, but to examine whether a unified configuration remains stable across multimodal tasks.

EMA momentum. For EMA momentum, we use $\alpha = 0.99$ as a standard smoothing default, analogous to moving averages in BatchNorm or Adam, rather than a heavily tuned hyperparameter. This value is sufficient to prevent abrupt threshold spikes.

Vision-critical token selection ratio. For vision-critical token selection, we set the selection ratio to 0.1. This choice is motivated by the observation behind VQA-CP [36]: predictions in vision-language tasks can be strongly influenced by textual priors, and only a small subset of answer tokens is genuinely determined by visual evidence. Accordingly, in multimodal speculative decoding, most response tokens are expected to be governed by linguistic context, whereas strongly image-dependent tokens should remain sparse. We therefore retain only the top 10% tokens with the highest target-side visual dependence scores as vision-critical positions, and this ratio is fixed across all datasets and target models.

Hidden-state enhancement. As generated responses become longer, accumulated linguistic context can increasingly dominate later textual states, weakening visual conditioning even when image tokens remain in the prefix. VHE adds a low-magnitude pooled image residual to question and answer hidden states before sparse refinement, providing a lightweight global visual anchor. It does not change which tokens are selected. Below, $q = \eta_Q$ and $a = \eta_A$ denote the question-side and answer-side enhancement coefficients used in the main text.

We ablate hidden-state enhancement on LLaVA-7B with TextVQA under the 20-epoch Stage-I plus 5-epoch refinement setting. We compare different enhancement strengths for the question and answer parts, together with three answer-side aggregation strategies: mean pooling, max pooling, and mean-max pooling. The tested variants achieve $2.83\times$–$2.86\times$ speedup. The best result is obtained by mean pooling with $q = 0.02$ and $a = 0.1$, reaching $2.86\times$. Overall, the gains remain stable across hyperparameter choices, suggesting that hidden-state enhancement is a robust visual-conditioning mechanism rather than a sensitive design component. We therefore adopt the mean-pooling configuration by default.

In a complementary coefficient-zero ablation, disabling VHE gives a speedup of $2.76\times$ (Table 7). For the tested nonzero settings with $q \in \{0.01, 0.02\}$ and $a \in [0.08, 0.15]$, speedup ranges from $2.83\times$ to $2.86\times$; the default $q = 0.02$, $a = 0.10$ reaches $2.86\times$. The narrow $0.03\times$ spread across nonzero settings supports that VHE is not sensitive to hyperparameters.

Table 7: VHE coefficient-zero ablation and sensitivity to question-side (q) and answer-side (a) enhancement strengths.

VHE setting	Speedup α
Without VHE (zero coefficients)	$2.76\times$
$q \in \{0.01, 0.02\}$, $a \in [0.08, 0.15]$	$2.83\times$–$2.86\times$
Default $q = 0.02$, $a = 0.10$	$2.86\times$

B.3 Training Settings

Initialization and backbone. For downstream fine-tuning, we start from the 20-epoch EAGLE-2 which is trained by the official default configuration.

Data format. Both baselines and our method operate on EAGLE-style distilled supervision tuples stored as tensors. Each training instance contains `hidden_state`, `input_ids`, and `loss_mask`.

B.3.1 SFT Baseline (Fine-tuning after 20 epochs)

For the SFT baseline, we continue training the draft model from the 20-epoch Stage-I checkpoint using dense token-level refinement. The model is optimized on all response tokens indicated by the original loss mask, without applying any critical-token selection. Training is implemented with Accelerate under bf16 mixed precision. We use AdamW with $(\beta_1, \beta_2) = (0.9, 0.95)$, a learning rate of 8×10^{-6}, batch size 1, gradient accumulation 1, maximum sequence length 2048, 100 warmup steps, and gradient clipping with threshold 1.0.

B.3.2 CATS (Sparse-update after 20 epochs)

For CATS, we use the same draft architecture, cached distillation features, and loss formulation as SFT, but replace dense refinement with sparse critical-token updates. Stage II is initialized from the 20-epoch Stage-I checkpoint and trained with AdamW under bf16 mixed precision. The default hyperparameters are learning rate 8×10^{-6}, batch size 1, gradient accumulation 1, maximum sequence length 2048. CATS constructs the update mask by combining a dynamic hard-token mask, derived from the draft–target mismatch with EMA momentum 0.99 and $\lambda = 0.8$, and a static visual-token mask that marks visually dependent positions. The loss is computed only on the selected response tokens, while non-critical tokens are excluded from optimization.

B.3.3 Effect of Training Duration

To assess whether the advantage of CATS can be explained by insufficient SFT training, we extend both SFT and CATS to 40 epochs. Table 8 compares the best SFT speedup over epochs 20–40 with the CATS speedup range over epochs 30–40. Both methods plateau by approximately epoch 30, while CATS retains a substantial advantage across all four target models. These results suggest that the observed gap is not explained solely by a shorter SFT training budget.

Table 8: Speedup after extending training to 40 epochs. SFT reports its best result over epochs 20–40; CATS reports its range over epochs 30–40.

Target model	Best SFT epochs 20–40	CATS epochs 30–40
LLaVA-1.5-7B	$2.22\times$	$3.07\times$–$3.08\times$
LLaVA-1.5-13B	$2.29\times$	$2.84\times$–$2.85\times$
Qwen2.5-VL-3B	$1.98\times$	$2.75\times$–$2.77\times$
Qwen2.5-VL-7B	$1.63\times$	$2.30\times$–$2.31\times$

B.4 Dense refinement vs Sparse refinement

Finally, Table 9 compares dense refinement with our sparse critical-position refinement. Sparse refinement consistently outperforms dense refinement across both model families, especially on reasoning-oriented datasets. On LLaVA-1.5-7B at $T = 0$, it improves speedup from $2.00\times$ to $2.97\times$ on ScienceQA and from $2.21\times$ to $3.17\times$ on A-OKVQA. This indicates that dense target-supervised refinement after coarse alignment is suboptimal, as many tokens are already well-aligned and contribute little to acceptance. Optimizing them further dilutes the training signal. Sparse refinement instead concentrates supervision on tokens most associated with rejection and visual grounding, making the same budget more effective for improving acceptance and speedup.

B.5 Complementarity of Hard-Token and Vision-Critical Selection

Across the original evaluation protocol, $\mathcal{H} \cup \mathcal{V}$ achieves the highest speedup in all six model–dataset settings. Its improvement over the better single selector ranges from $0.10\times$ to $0.21\times$. This consistent advantage indicates that hard-token and vision-critical selection capture complementary sources of rejection rather than redundant token subsets.

B.6 Dynamics of Hard-Token Selection

Hard-token identities change as the draft model is updated. We track these changes across checkpoints on the same 500 held-out ScienceQA examples. Table 11 reports transitions in both directions: some previously hard positions become easy, while some previously easy positions become hard. A selection fixed at an early checkpoint can therefore miss later-emerging discrepancies.

To compare coverage at an equal selection budget, we fix both selectors to 21,116 positions and measure how many of the 23,570 target–draft top-1 disagreement positions at epoch 20 they cover (Table 12). Dynamic HTDS covers 21,116 such positions, compared with 19,519 for the static epoch-1 ranking, increasing coverage from 82.8% to 89.6%. Thus, dynamic selection covers 1,597

Table 9: Comparison over dense token-level refinement versus CATS sparse updates after global coarse alignment through speedup ratio α and average accepted token length τ.

Model	Method	Temperature=0						Temperature=1					
		ScienceQA		A-OKVQA		TextVQA		ScienceQA		A-OKVQA		TextVQA	
		τ	α	τ	α	τ	α	τ	α	τ	α	τ	α
LLaVA-1.5 7B	Dense	2.72	2.00×	2.88	2.21×	2.91	2.30×	2.53	1.91×	2.21	1.60×	2.41	1.81×
	Sparse	**3.85**	**2.97**×	**4.12**	**3.17**×	**3.76**	**2.91**×	**3.18**	**2.34**×	**2.66**	**1.91**×	**2.49**	**1.84**×
Qwen2.5-VL 7B	Dense	2.24	1.59×	2.45	1.80×	1.97	1.37×	2.04	1.45×	1.92	1.38×	1.72	1.22×
	Sparse	**3.21**	**2.50**×	**3.42**	**2.65**×	**2.51**	**1.85**×	**2.74**	**2.05**×	**2.27**	**1.68**×	**2.09**	**1.51**×

Table 10: Speedup ratios for hard-token-only, vision-critical-only, and union-based selection under the original evaluation protocol.

Model	Dataset	H-only	V-only	$\mathcal{H} \cup \mathcal{V}$
LLaVA-1.5-7B	ScienceQA	2.56×	2.42×	2.77×
LLaVA-1.5-7B	A-OKVQA	2.83×	2.68×	2.98×
LLaVA-1.5-7B	TextVQA	2.79×	2.64×	2.89×
Qwen2.5-VL-7B	ScienceQA	2.23×	2.10×	2.37×
Qwen2.5-VL-7B	A-OKVQA	2.50×	2.35×	2.64×
Qwen2.5-VL-7B	TextVQA	1.71×	1.58×	1.84×

Table 11: Changes in hard-token identities across training checkpoints on 500 held-out ScienceQA examples.

Checkpoint interval	Hard → easy	Easy → hard
Epoch 1 → 3	8.2%	5.8%
Epoch 3 → 10	11.7%	7.6%
Epoch 10 → 20	9.2%	8.2%

additional final disagreement positions without selecting more tokens. Refreshing the selected set removes positions that have become easy and admits newly emerging hard positions.

Table 12: Epoch-20 disagreement coverage at the same selection budget. Coverage is measured against 23,570 target–draft top-1 disagreement positions.

Selector	Selected positions	Disagreements covered	Coverage
Static epoch-1 ranking	21,116	19,519	82.8%
Dynamic HTDS	21,116	**21,116**	**89.6%**

B.7 Token-Level Case Study

Figure 4 illustrates how CATS corrects visually grounded brand and model subwords in a TextVQA example. At four positions within the target response excerpt "Rolex Submariner," the Stage I draft predicts different tokens, whereas CATS matches all four target subwords. The reported one-step acceptance mass also increases at each position.

C Limitations

Although CATS consistently improves the speedup ratio and average acceptance length across the evaluated settings, several limitations remain. First, our experiments focus on three representative

Case Study: Visually Grounded Token Recovery

Dataset: TextVQA.

Question: "What company makes the watch?"

Target response excerpt: "Rolex Submariner."

Aligned predictions and acceptance mass.

Token at the aligned position			One-step acceptance mass	
Target	Stage I	CATS	Stage I	CATS
`Rolex`	`close`	`Rolex`	0.392	**0.843**
`Sub`	`watch`	`Sub`	0.136	**0.879**
`mar`	`.`	`mar`	0.004	**0.993**
`iner`	*space*	`iner`	0.002	**0.999**

Here, "space" denotes the whitespace token predicted by Stage I.

Interpretation. CATS recovers all four target subwords at these positions, including the visually grounded brand and model name. The accompanying increases in one-step acceptance mass illustrate the correction of local draft–target mismatches associated with rejection.

Figure 4: A TextVQA case study of visually grounded token recovery. CATS matches the four target subwords and increases the reported one-step acceptance mass at each position.

multimodal question-answering benchmarks, namely ScienceQA, A-OKVQA, and TextVQA, and four target models from the LLaVA and Qwen2.5-VL families. These datasets cover scientific reasoning, knowledge-based visual question answering, and text-rich visual understanding, but they do not exhaustively represent all multimodal scenarios, such as video understanding, multi-image reasoning, embodied interaction, or highly specialized document/chart domains. Therefore, the generality of CATS beyond the evaluated task and model families should be further validated.

Second, CATS relies on target-model supervision to estimate draft-target mismatch and visual dependence. This design is aligned with the acceptance-oriented objective of speculative decoding, but it requires access to target-model probabilities or teacher-forced likelihoods during training. As a result, the training cost may increase compared with purely data-driven supervised fine-tuning, especially when scaling to larger target models or much larger datasets.

Third, the current framework is evaluated under fixed draft architectures and inference configurations inherited from the EAGLE-style speculative decoding setup. While this choice ensures fair comparison with the baselines, it leaves open the question of how CATS interacts with other draft architectures, tree construction strategies, hardware-aware scheduling methods, or system-level optimizations. The observed speedup also depends on the balance between draft-model latency, target-model verification cost, and the average accepted token length; therefore, the gains may vary across hardware platforms and deployment environments.

Finally, CATS is designed to improve inference efficiency without changing the target model's output distribution under standard speculative decoding. It does not directly address broader issues such as factuality, hallucination, privacy, or fairness in multimodal generation. Future work should evaluate CATS on broader multimodal tasks, more diverse model families, larger-scale training data, and real-world deployment settings, while also studying its interaction with safety and robustness constraints.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the paper's main contributions: an acceptance-oriented critical-position alignment of multimodal speculative decoding, two complementary critical-token selection mechanisms, and a two-stage training framework for sparse refinement. These claims are supported by the theoretical analysis in Section 3 and Appendix A, and by experiments on ScienceQA, A-OKVQA, and TextVQA with multiple LLaVA and Qwen2.5-VL target models, where CATS consistently improves speedup and average acceptance length over EAGLE-2 and SFT baselines.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of CATS in Appendix C. In particular, we clarify that the current evaluation is limited to three multimodal QA benchmarks and four LLaVA/Qwen2.5-VL target models, that the method relies on target-model likelihoods for critical-token selection during training, and that its practical speedup may depend on draft architecture, hardware, and deployment configuration.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: We provide the assumptions, formal statements, and complete proofs for all main theoretical results in Appendix A.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides the main experimental protocol in Section 4, including the evaluated datasets, metrics, target models, baselines, decoding temperatures, and the shared EAGLE-2 draft architecture used for fair comparison. Additional reproducibility details are provided in Appendix B.1–B.3, including distillation data construction, evaluation sampling, teacher decoding settings, hyperparameter choices, and the Stage II training configurations for both SFT and CATS.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example

(a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes].

   Justification: We provide a public code repository at `https://github.com/ChocoLab-Q/CATS`, including the implementation of CATS, configuration files, data preparation instructions, and scripts for reproducing the main experimental results. The repository documents the required environment and commands for training and evaluation. Baseline results are reproduced using the same draft architecture and inference configuration described in the paper.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes].

   Justification: Section 4.1 specifies the core experimental setting, including the evaluated datasets, metrics, target models, baselines, hardware, training stages, learning rates, decoding temperatures, and the fair-comparison protocol where all methods share the same EAGLE-2 draft architecture and inference configuration. Full training and test details are further provided in Appendix B.1–B.3, including distillation data construction, evaluation sampling, teacher decoding settings, optimizer choices, loss masks, hyperparameter sensitivity analysis, and the Stage II configurations for both dense SFT refinement and CATS sparse refinement.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Following prior speculative decoding work [15, 16], we evaluate efficiency using wall-clock speedup and average acceptance length τ. For each experimental setting, both metrics are averaged over three repeated inference-evaluation runs using the same checkpoint and evaluation protocol, consistent with the practice of repeated latency measurements and mean reporting in DistillSpec [9]. We report these mean values without error bars, confidence intervals, or formal statistical significance tests.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Section 4.1 specifies the experimental hardware, consisting of 4 RTX 4090 and 4 A6000 GPUs, along with the evaluated models, datasets, baselines, and training schedule. Appendix B.1–B.3 further documents data construction, model configurations, optimizer settings, training losses, batch sizes, and numerical precision, providing implementation details to support reproduction of the experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes].

   Justification: The research conforms to the NeurIPS Code of Ethics. The work uses publicly available datasets and open-source models, does not involve human-subject experiments or crowdsourcing, does not release high-risk generative models or scraped datasets, and is presented as an efficiency-oriented method for multimodal speculative decoding with limitations and broader-impact considerations discussed in the paper.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes].

    Justification: This work focuses on improving the inference efficiency of existing multimodal models through draft-model training. Under standard speculative decoding, CATS preserves the target model's output distribution and does not introduce new generative capabilities. However, accelerating multimodal models may allow more content to be generated within the same time budget, potentially amplifying both their benefits and existing risks. We therefore advocate the responsible use of such acceleration techniques, with appropriate content moderation in downstream applications.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A].

    Justification: This work does not release a new high-risk pretrained model, image generator, scraped dataset, or user-facing generative system. All experiments are conducted using publicly available datasets and open-source target models, and CATS is a training framework for multimodal speculative decoding rather than a new data or model release with additional misuse risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes].

    Justification: The paper uses publicly available datasets and open-source model checkpoints, including ScienceQA, A-OKVQA, TextVQA, LLaVA, Qwen2.5-VL, and the EAGLE-2 draft-model framework, and credits the original creators through citations in the main text and references. We follow the original licenses and terms of use of these assets.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes].

Justification: This paper introduces CATS and releases the corresponding code as a new asset, including the implementation of hard-token dynamic selection, vision-critical token selection, sparse refinement, and evaluation scripts. The code release includes documentation, configuration files, dependency information, and license terms for reproducibility; no new dataset or trained model checkpoint is released.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A].

Justification: This work does not involve crowdsourcing experiments, human-subject studies, user studies, or newly collected human annotations. All experiments are conducted using existing public datasets and open-source models, so participant instructions, screenshots, and compensation details are not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A].

Justification: This work does not involve crowdsourcing, user studies, newly collected human-subject data, or experiments with human participants. Therefore, participant risk disclosure and IRB or equivalent approval are not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A].

Justification: The core method development of CATS does not involve LLMs as important, original, or non-standard components beyond the conventional use of open-source MLLMs as target/backbone models for evaluation. As stated in the Disclosure of LLM Usage, any general-purpose LLM usage was limited to minor copyediting and did not affect the methodology, experiments, code, citations, or scientific conclusions.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.