BOW: Training Language Models to Reason Over Plausible Next Words

Ming Shen*, **Zhikun Xu***, **Jacob Dineen**, **Xiao Ye**, **Ben Zhou**
Arizona State University

arXiv:2506.13502v4 [cs.CL] 31 Aug 2026

Abstract

Next-word prediction (NWP) trains language models against a single observed continuation, even though many contexts admit multiple plausible next words. Recent RL-based next-word reasoning methods make this tension explicit: they reward a model for producing a rationale that supports one context-conditioned continuation, which can turn a pre-existing preference into a confident, self-justifying trajectory. We introduce BOW, an RL framework that instead trains models to produce self-contained, neutral, and comprehensive descriptions of the plausible next-word space. The policy generates a next-word reasoning trajectory from the full context, but a frozen scorer computes the core reward from that trajectory alone, without separately receiving the context. BOW-Reg adds a lightweight breadth regularizer to this core reward to discourage premature collapse. On two model backbones, BOW remains competitive with the original models and often outperforms trained baselines across ten general reasoning benchmarks. On benchmarks testing ambiguous references and word meanings, BOW-Reg achieves the highest SharedRef correctness and the lowest HoWN-Simple single-sense collapse on both backbones. Human evaluation further shows that BOW-Reg produces broader next-word reasoning trajectories, while direct next-word-prediction evaluation shows that these trajectories remain predictive.[1]

1 Introduction

Large language models (LLMs) are primarily trained with next-word prediction (NWP) during pretraining and post-training (Devlin et al., 2019; Raffel et al., 2020; Brown et al., 2020; OpenAI, 2023). Yet natural language is inherently underdetermined: a context often admits multiple plausible next words, whereas NWP rewards only the single continuation observed in the data. This mismatch can contribute to memorization-related bias (Lee et al., 2022; Hans et al., 2024; Carlini et al., 2023). This tension becomes more consequential in recent RL-based next-word reasoning methods such as Reinforcement Pre-Training (RPT) (Dong et al., 2025) and Reinforcement as a Pretraining Objective (RLP) (Hatamizadeh et al., 2026). Although effective, these methods remain answer-centric: they reward reasoning trajectories that support one context-conditioned continuation under the full input context. When the base model already prefers one plausible next word over others, such training can reinforce that skew[2] by turning a mild preference into a confident, self-justifying rationale.

Fig. 1 gives an intuitive example. Consider the context *Mary works in a hospital and interacts with patients and other medical staff throughout the day. Mary is a*. The observed next word is *nurse*, which the base model already favors over *doctor*. The word *nurse* is a valid continuation, but the context also supports other healthcare roles. An answer-centric objective can turn the model's prior preference for *nurse* into a rationale that presents it as uniquely implied. The same failure can arise whenever several continuations remain plausible but the model already favors one of them. We refer to this as a bias-reinforcing mode of RL-based next-word reasoning.

This failure is especially likely because the policy is conditioned on a context that already induces a skewed next-token distribution, while the reward is tied to the observed continuation. A trajectory that turns the favored word into a confident explanation can therefore be reinforced even when

*Equal contribution.

[1]Code available at https://github.com/ARC-ASU/BOW.

[2]Here and throughout the paper, we use bias broadly to mean a pre-existing skew in the model's next-word distribution toward one plausible continuation, including corpus-frequency and lexical-shortcut effects; it is not restricted to demographic or social bias.

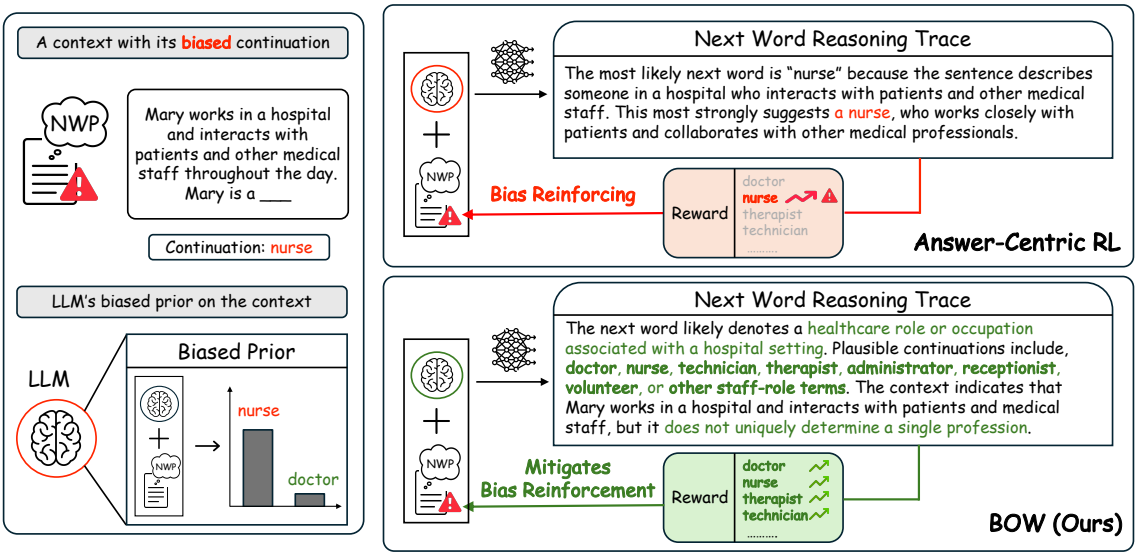


Figure 1: In an underdetermined context, the vanilla Qwen-based model assigns a higher probability to *nurse* than to *doctor*. Answer-centric RL rewards a trajectory that treats this favored continuation as uniquely implied, while BOW trains a neutral description of the broader plausible next-word space. Probabilities and trajectories are actual Qwen-based outputs.

the context still permits alternatives. Our goal is not to suppress the observed word, but to prevent its supervision from forcing the reasoning trajectory to treat one plausible continuation as uniquely licensed. Prior work has shown that reducing inductive bias can generally improve LLM reasoning capabilities (Wang et al., 2023a; Zhang et al., 2025; Wang et al., 2025). We hypothesize that removing this bias-reinforcing mode improves the model's general reasoning. To avoid this mode, we train reasoning that preserves uncertainty over the plausible continuation space rather than collapsing immediately to a single answer.

We therefore propose BOW (Bottlenecked Next-Word Exploration). BOW changes the target of the reasoning trajectory: instead of justifying a single favored continuation, the policy produces a self-contained, neutral, and comprehensive description of the plausible next-word space. To make the trajectory responsible for the core reward, we remove the scorer's direct access to the original context in the core scoring pass. The policy still conditions on the context, and a self-contained trajectory may restate the contextual constraints needed for prediction. The scorer, however, computes the core reward from the trajectory alone, without receiving context as a separate input. Context can therefore affect this reward only through information carried by the trajectory; there is no separate context-to-scorer path that bypasses the trajectory. App. A formalizes this distinction. The core reward is the scorer's probability assigned to the gold next token, so partially informative trajectories can receive graded credit. This better matches linguistic uncertainty than hard exact-match supervision. We prompt the model to follow this trajectory form rather than directly name a single answer. Prompting alone may not ensure this behavior, so BOW-Reg adds a lightweight breadth regularizer that discourages premature collapse to narrow answer naming. We treat BOW-Reg as the full method and report BOW-Core as an analysis variant that uses the core reward without the breadth regularizer.

Empirically, we adapt Qwen2.5-7B-Instruct and LLaMA3.1-8B-Instruct. Across ten general reasoning benchmarks, BOW remains competitive with the original instruction models and often outperforms both trained continual pretraining and RL baselines. Additional NWP-style adaptation on raw text can degrade instruction following, so matching or improving the original models indicates that BOW preserves broad capability. Targeted ambiguity and human evaluations test the collapse-mitigation claim directly. To evaluate the targeted failure mode, we further test on SharedRef (Ellinger and Groh, 2025) and HoWN-Simple (Ellinger et al., 2025), which assess referential ambiguity and lexical sense collapse. BOW-Reg is strongest on SharedRef and achieves the lowest HoWN-Simple single-sense collapse on both backbones. Human evaluation further shows that BOW-Reg elicits broader and more comprehensive next-word reasoning trajectories, whereas BOW-Core is more willing to verbalize a specific answer.

Our contributions are: (i) characterizing answer-centric collapse as a limitation of next-word RL, (ii) proposing a core reward computed from the generated trajectory alone, together with a breadth regularizer, and (iii) showing across general reasoning, targeted ambiguity, intrinsic NWP, and human evaluation that this training principle improves reasoning quality while reducing premature collapse.

2 Related Work

Recent work has made intermediate reasoning explicit during language modeling. ToW (Xu et al., 2025) and Rationalyst (Jiang et al., 2025) add supervised intermediate reasoning, while STaR (Zelikman et al., 2022) and Quiet-STaR (Zelikman et al., 2024) use self-generated reasoning to improve prediction. These methods establish the value of reasoning before output, but their objectives do not directly train intermediate reasoning to preserve uncertainty over the next-word candidate space. BOW instead uses RL to train trajectories that emphasize contextual constraints and plausible continuations rather than solely supporting one realized word.

Closest to our setting, recent RL-based next-word reasoning methods such as RPT (Dong et al., 2025) and RLP (Hatamizadeh et al., 2026) show that RL can be an effective post-training strategy for language modeling, but they remain answer-centric, rewarding trajectories for supporting a single favored continuation under the full context. More broadly, language models often exploit shortcuts, spurious correlations, or plausible-looking rationales rather than robust inference (Li et al., 2024; Zhou et al., 2024; Yu et al., 2024). We study this failure mode in next-word RL, where answer-centric training can turn an existing preference among plausible continuations into a more confident, self-justifying trajectory. BOW instead trains trajectories to describe the plausible next-word space rather than solely justify one realized continuation.

3 Method

We now formalize BOW. Let C denote the input context and w^* the supervised next token, i.e., the first token of the ground-truth next word; we use "next word" informally throughout. A policy language model π_θ first generates an intermediate next-word reasoning trajectory $\tau \sim \pi_\theta(\cdot \mid C)$. The policy conditions on the full context C but never observes w^*; supervision arrives only through a scalar reward assigned after τ is produced.

3.1 Reasoning Goal and Setup

The goal of τ is not to name one answer as early as possible. Instead, it should reason over the plausible next-word space in a way that is informative enough to support prediction, yet broad enough to avoid premature answer commitment. This distinguishes BOW from recent answer-centric next-word RL: we train the model to reason over the continuation space rather than justify one favored continuation.

We ask each trajectory to satisfy three properties. Self-contained: it restates the contextual constraints needed for prediction rather than leaving them implicit in C. Neutral: it does not present one favored continuation as uniquely implied unless the candidate space is genuinely small. Comprehensive: it captures the main factors shaping the next word without exhaustively listing an open-ended candidate set. Explicit enumeration is allowed only when the plausible set is small and concrete.

Note that self-contained does not mean context-free. Because τ is generated from C, it may restate or summarize relevant parts of C. The bottleneck concerns the input to the core scoring pass, not whether τ repeats words from C. The prompt specifies this trajectory form, while the learning objective determines which trajectories are reinforced. Fig. 4 gives the exact policy prompt.

3.2 BOW-Core: Core Reward from the Trajectory Alone

We place the bottleneck in *credit assignment*: the policy conditions on the full context C, while the core scorer computes reward from τ alone. Any context-derived evidence used for this reward must first be represented in the trajectory.

We use a frozen scorer J_ϕ. Given a reasoning trajectory τ, the scorer is prompted to predict the immediate next token from τ alone, yielding a next-token distribution $P_{J_\phi}(\cdot \mid \tau)$. Here, J_ϕ is not a separately trained judge or reward model; it is a frozen off-the-shelf LM, and the gold next token is the only task-specific supervision used to compute the core reward. The core reward is

$$r_{\text{core}}(\tau) := P_{J_\phi}(w^* \mid \tau), \tag{1}$$

where w^* is the supervised next token. Because r_{core} is computed from $P_{J_\phi}(w^* \mid \tau)$, contextual evidence can affect gold-token credit only after the

policy writes it into τ. A scorer that separately received C in the core scoring pass could assign high probability to w^* while ignoring the trajectory, allowing the reward to bypass τ. BOW-Core removes this route. This restriction does not prohibit overlap between C and τ; it makes the trajectory the required carrier of context-derived evidence used by the core scorer. We call this objective BOW-Core, which lets us evaluate the effect of computing the core reward from the generated trajectory alone. App. A formalizes its reward-path bottleneck. Fig. 3 illustrates how candidate next words encoded in τ can appear among the scorer's high-ranked tokens, and Fig. 6 gives the scorer prompt.

BOW-Core also yields a soft, distributional reward rather than a binary one. A trajectory that substantially narrows the plausible continuation space, even without identifying the exact gold next word, can still receive graded credit. This better matches linguistic underdetermination and provides a denser learning signal than exact-match rewards. App. B further motivates the partial-credit advantage of the soft reward.

3.3 BOW-Reg: The Full Objective with Lightweight Breadth Regularization

BOW-Core encourages the intended trajectory form, but it does not fully prevent collapse to narrow, answer-like trajectories. We sometimes observe trajectories that enumerate only a few candidate words or wrap the current favorite answer in slightly longer prose during training. This can be appropriate when the candidate set is genuinely small, but not when the context still supports a broader range of continuations. BOW-Reg therefore adds a lightweight breadth regularizer to discourage this behavior.

Using the same frozen scorer, we take $P_{J_\phi}(\cdot \mid C)$ as a reference distribution and define

$$R_{\text{reg}}(\tau) := r_{\text{core}}(\tau) - \alpha \left\| P_{J_\phi}(\cdot \mid \tau) - P_{J_\phi}(\cdot \mid C) \right\|_1 . \quad (2)$$

where $\alpha \geq 0$ controls the strength of the regularizer. The regularizer shapes rather than replaces the core objective; by itself, it does not guarantee unbiased behavior. It discourages trajectories whose induced next-token distribution drifts too far from the reference; in practice, such drift often corresponds to premature answer naming. We interpret Eq. 2 as a *breadth regularizer*: the prompt specifies the desired trajectory form, while the regularizer helps preserve it when optimization under the core reward becomes too narrow. As a result, BOW-Core uses only the gold-token term $P_{J_\phi}(w^* \mid \tau)$, whereas BOW-Reg adds the breadth regularizer in Eq. 2.

3.4 Policy Optimization

Given a training pair (C, w^*), the policy samples a group of reasoning trajectories $\{\tau_1, \ldots, \tau_N\}$ from $\pi_\theta(\cdot \mid C)$. For each τ_i, we compute either $r_{\text{core}}(\tau_i)$ for BOW-Core or $R_{\text{reg}}(\tau_i)$ for BOW-Reg. The policy is then trained to maximize expected reward. In our experiments, we optimize this objective with Group Relative Policy Optimization (GRPO) (Shao et al., 2024). For each context, rewards are normalized within the sampled group to obtain relative advantages, and the policy is updated with PPO-style steps (Schulman et al., 2017). Group normalization is useful because the absolute reward scale can vary across contexts with different levels of next-word uncertainty. The method is not tied to GRPO; standard policy-gradient methods such as PPO (Schulman et al., 2017) or REINFORCE (Williams, 2004) can also be used.

4 Experimental Setup

We evaluate BOW on four task-facing axes: general zero-shot reasoning, targeted ambiguity and collapse, human judgments of candidate-space reasoning, and intrinsic next-word prediction. We additionally test robustness to scorer family, scorer size, and reward sharpness.

4.1 Training Data

We construct the RL training set to emphasize reasoning-critical next-word prediction: contexts where multiple next words are plausible but deeper contextual reasoning is needed to narrow the candidate space. Starting from long-form narrative prose drawn from prior work (Del and Fishel, 2023), we apply a reasoning-centric filter that removes locally trivial continuations, such as function-word completions and highly formulaic collocations. This yields about 45K context–next-word pairs, where accurate prediction typically requires tracking entities, events, and implicit world dynamics across multiple sentences. Although the source texts are narrative, the selection criterion is task-driven rather than genre-specific: we retain only positions where the next word requires non-trivial inference

from the preceding text. App. C.1 describes the filter, and App. E reports an ablation against random filtering. The filter provides modest but consistent gains by concentrating updates on contexts where next-word reasoning is needed.

4.2 Models and Optimization

We use Qwen2.5-7B-Instruct (Yang et al., 2024) and LLaMA3.1-8B-Instruct (Dubey et al., 2024) as policy models. For both model families, we train for one epoch with a total batch size of 1024, a mini-batch size of 256, and a rollout size of 5. We disable the KL term used in standard GRPO following recent works (Liu et al., 2025; Yu et al., 2025).

We use LLaMA3.1-8B-Instruct as the default frozen scorer J_ϕ because its trajectory-conditioned distributions are less sharply concentrated. Sec. 5.5 analyzes this choice and tests robustness to scorer family, scorer size, and reward sharpness. To obtain $P_{J_\phi}(\cdot \mid \tau)$ in the core reward, we use temperature $T = 5$ and retain the top 100 tokens of the scorer distribution. If the first token of the gold next word does not appear in this set, the trajectory receives reward 0. For BOW-Reg, we use the same top-100 truncation for both distributions in Eq. 2 and set the regularization strength to $\alpha = 0.1$.[3] Additional implementation details, including training-time compute, are reported in App. C.4.

4.3 Baselines

We compare BOW against two continual-pretraining baselines, two answer-centric RL baselines, and the original instruction-tuned models for reference. For fair comparison, all trainable baselines use the same filtered supervision positions as BOW. The continual-pretraining baselines are SLM (Lin et al., 2024), which applies selective language modeling to the same context–next-word pairs, and ToW (Xu et al., 2025), which augments those positions with distilled word-level thoughts from a stronger annotator model before standard causal language modeling.

The answer-centric RL baselines are BR (Binary-Reward RL), an RPT-style (Dong et al., 2025) objective that reasons and then commits to a single next-word prediction with exact-match reward on the first token, and IGR (Information-Gain RL), a simplified RLP-style (Hatamizadeh et al., 2026) objective that uses a frozen reference scorer to reward the difference between reasoned and baseline log-evidence. We also report the original Qwen2.5-7B-Instruct and LLaMA3.1-8B-Instruct models without additional training, since further NWP-style adaptation can affect instruction-following behavior.

4.4 Evaluation Protocol

For general reasoning evaluation, we use ten benchmarks spanning commonsense, world knowledge, and multi-hop reasoning: CSQA (Talmor et al., 2019), PIQA (Bisk et al., 2020), StrategyQA (Geva et al., 2021), TruthfulQA (Lin et al., 2022), ARC-Challenge (Clark et al., 2018), WinoGrande (Sakaguchi et al., 2020), BBH (Suzgun et al., 2023), MMLU (Hendrycks et al., 2021), MMLU-Pro (Wang et al., 2024), and GPQA (Rein et al., 2024). We use each benchmark in its original multiple-choice QA format. We perform zero-shot inference by prompting the policy to think step by step before outputting the final answer letter. We use Math-Verify[4] to extract the answer letter. Fig. 8 gives the inference prompt.

For targeted ambiguity evaluation, we evaluate the English SharedRef split under the Normal prompt (Ellinger and Groh, 2025) and the English HoWN-Simple setting (Ellinger et al., 2025). SharedRef reports official judge-based Correctness under referential ambiguity. HoWN-Simple evaluates homonym definitions in response to a simplified-language request; we report the official Covered and single-sense Collapse. These metrics assess whether a model preserves plausible referents or senses rather than prematurely committing to a single one. App. C.2 provides additional dataset and evaluation details.

For intrinsic next-word prediction evaluation, we convert five benchmarks into multiple-choice next-word prediction tasks while preserving the original reasoning requirement: CSQA, PIQA, StrategyQA, ARC-Challenge, and WinoGrande. The policy first generates a next-word reasoning trajectory; the frozen scorer then scores the candidate next words using the same trajectory-conditioned prompt as in training, and the highest-probability candidate is taken as the prediction. Following prior work, we do not report intrinsic NWP for IGR. For BOW, we report only BOW-Reg because

[3] Set once from the observed training-time magnitudes of the two terms, without tuning on evaluation performance, so the regularizer remained influential without dominating the core reward.

[4] https://github.com/huggingface/Math-Verify

	CSQA	PIQA	SQA	TQA	ARC-c	WG	BBH	MMLU	MMLU-p	GPQA
Qwen2.5-7B-Instruct										
Vanilla	80.51	87.21	64.21	64.65	92.06	64.48	77.76	75.28	58.53	31.92
SLM	75.51	84.55	62.99	53.98	88.05	59.43	54.92	69.53	48.11	29.24
ToW	81.16	88.41	**67.69**	57.16	**93.00**	66.14	69.21	69.52	55.16	32.14
BR	77.64	87.05	60.53	61.08	89.76	62.59	76.82	70.82	46.17	31.25
IGR	78.71	87.27	63.39	65.73	92.41	68.43	**78.01**	75.60	56.94	29.69
BOW-Core	81.49	87.21	64.83	63.28	92.75	64.56	76.78	75.69	**58.22**	32.37
BOW-Reg	**81.90**	**88.68**	63.19	**66.83**	92.32	**70.96**	77.21	**76.51**	57.51	**33.26**
LLaMA3.1-8B-Instruct										
Vanilla	77.07	85.15	69.33	60.47	89.25	59.59	74.71	72.90	50.50	32.14
SLM	36.12	67.85	53.78	9.55	45.82	52.17	22.01	39.32	12.80	24.55
ToW	22.28	60.07	46.83	28.64	34.30	45.07	22.71	43.77	21.16	25.00
BR	76.66	83.73	66.46	53.12	84.47	57.14	65.89	66.33	38.56	22.32
IGR	77.81	86.62	67.89	58.63	90.02	60.06	73.98	73.27	49.78	31.03
BOW-Core	**78.71**	**87.11**	**68.92**	58.75	**90.36**	**61.96**	74.10	**74.21**	**53.13**	**35.04**
BOW-Reg	77.97	86.67	67.28	**60.22**	89.16	59.75	**74.56**	73.59	52.71	32.14

Table 1: General reasoning evaluation of BOW-Core, BOW-Reg, and baselines. Bold indicates the best result among trained methods. Vanilla instruction models are shown as reference points and are not included in comparisons among trained methods. We use SC@10 (self-consistency) for all benchmarks. TQA = TruthfulQA, SQA = StrategyQA, ARC-c = ARC-Challenge, WG = WinoGrande, and MMLU-p = MMLU-Pro.

this evaluation assesses the full BOW-Reg objective; BOW-Core is an analysis variant that omits the breadth regularizer. Tab. 5 shows two transformed examples, and Apps. C.3 and C.5 provide additional transformation and baseline details.

For the general reasoning and intrinsic next-word prediction evaluations, we use vLLM (Kwon et al., 2023) and report self-consistency (Wang et al., 2023b) with 10 samples (SC@10). We sample 10 reasoning trajectories at temperature 0.8 and take a majority vote over the resulting predictions. Neither SharedRef nor HoWN-Simple is a multiple-choice evaluation; for both, we generate one response per instance and apply the official evaluation schema.

5 Results

5.1 General Zero-Shot Reasoning

Tab. 1 reports SC@10 on ten reasoning benchmarks. Across the two model families, a BOW variant is the strongest trained method on 17 of 20 model–benchmark pairs. These results show that training over the plausible next-word space preserves general reasoning performance and often improves over trained baselines, even though BOW is not optimized for downstream QA.

On Qwen, BOW-Reg achieves the best average score among trained methods, with BOW-Core close behind; together the two variants are strongest on 7 of the 10 benchmarks. On LLaMA, BOW-Core achieves the best average score, and the two variants are strongest on all of 10 benchmarks. In both model families, BOW remains competitive with and often slightly above the original instruction-tuned model despite the risk that additional next-word adaptation can weaken instruction following.

The continual-pretraining baselines are less stable across model families. On Qwen, ToW remains competitive on a few tasks but trails BOW on average despite using distilled supervision from a stronger annotator model. On LLaMA, SLM and ToW degrade sharply due to abstentions and degenerate generations, whereas both BOW variants remain stable. BOW-Core shows that the core reward provides a strong training signal even without breadth regularization. BOW-Reg preserves this performance while more directly targeting the ambiguity-collapse behavior evaluated next.

5.2 Targeted Ambiguity and Premature-Collapse Evaluation

Tab. 2 reports targeted ambiguity results. On SharedRef, BOW-Reg reaches 78.85 with Qwen, outperforming the strongest non-BOW baseline, ToW at 66.52, as well as BR at 56.39 and BOW-Core at 50.66. With LLaMA, BOW-Reg reaches 29.07, slightly above SLM at 28.40 and above BOW-Core at 21.15, IGR and ToW at 15.86, and BR at 13.66.

On HoWN-Simple, BOW-Reg achieves the lowest single-sense collapse on both backbones: 31.90 with Qwen and 7.36 with LLaMA. It does not uni-

	SharedRef	HoWN-Simple	
	Correctness ↑	Covered ↑	Collapse ↓
`Qwen2.5-7B-Instruct`			
Vanilla	36.12	48.64	48.15
SLM	36.12	47.48	41.10
ToW	66.52	48.65	37.68
BR	56.39	**52.20**	48.47
IGR	15.42	45.77	61.11
BOW-Core	50.66	51.19	44.44
BOW-Reg	**78.85**	50.84	**31.90**
`LLaMA3.1-8B-Instruct`			
Vanilla	14.10	59.20	28.22
SLM	28.40	13.20	85.37
ToW	15.86	49.75	36.20
BR	13.66	56.38	25.00
IGR	15.86	55.67	28.22
BOW-Core	21.15	**65.86**	8.59
BOW-Reg	**29.07**	63.91	**7.36**

Table 2: Targeted ambiguity evaluation on English SharedRef (Normal) and HoWN-Simple. All scores are percentages. Bold indicates the best result among trained methods.

formly maximize coverage: BR is highest with Qwen, while BOW-Core is highest with LLaMA. Even so, BOW-Reg remains competitive on coverage while most consistently avoiding single-sense collapse. The two HoWN metrics capture different aspects of behavior: Covered measures how many senses are mentioned, while Collapse records whether the response reduces the word to one sense. BOW-Reg does not simply maximize the number of senses mentioned; it most consistently controls the single-sense failure mode targeted by the method.

The non-BOW baselines are less consistent across settings. ToW is the strongest non-BOW method on Qwen SharedRef, while SLM is competitive on LLaMA SharedRef, but neither combines strong referential-ambiguity performance with low HoWN-Simple collapse across both backbones. SLM, for example, reaches 85.37 collapse with LLaMA despite its relatively strong SharedRef result. Among the evaluated methods, BOW-Reg is the only one that achieves both the highest SharedRef correctness and the lowest HoWN-Simple collapse for both model families. This consistency matters because the targeted failure mode spans both referential and lexical ambiguity, rather than performance on a single benchmark.

Together, the two benchmarks separate the roles of the variants. BOW-Core can preserve strong sense coverage, whereas BOW-Reg more strongly controls premature commitment. We therefore use

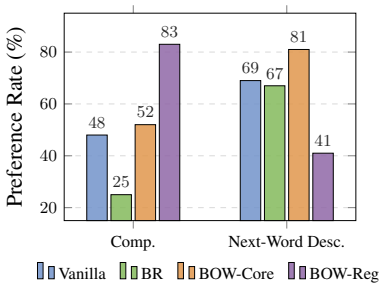


Figure 2: Human-evaluation preference rates for comprehensive next-word reasoning and expected next-word description over 150 contexts. Ties were allowed, so rates need not sum to 100%.

BOW-Reg as the main method for the failure mode studied here, while BOW-Core serves as the analysis variant without breadth regularization.

5.3 Human Evaluation of Next-Word Reasoning

Benchmark accuracy does not reveal the form of the next-word reasoning trajectories. We therefore complement Tab. 1 and 2 with a human study of the trajectories themselves.

We curate 150 contexts where several immediate next words are superficially plausible but deeper contextual reasoning narrows the continuation space. We compare four Qwen-based models: the vanilla instruction model, BR, BOW-Core, and BOW-Reg. Annotators view anonymized, shuffled trajectories and judge which one best reasons over the plausible next-word space in a self-contained and comprehensive way, and which next-word description best matches their expected continuation. Ties are allowed. App. D provides the full curation and annotation protocol.

Fig. 2 provides behavioral evidence. The comprehensiveness dimension evaluates the broader trajectory form targeted by BOW. BOW-Reg receives an 83% preference rate on this dimension, compared with 52% for BOW-Core, 48% for the vanilla model, and 25% for BR. For expected next-word description, BOW-Core leads at 81%, followed by the vanilla model at 69% and BR at 67%, while BOW-Reg receives 41%. Because ties are allowed, the preference rates need not sum to 100.

This reversal reflects the two objectives. BOW-Core more readily states a likely continuation, which aligns with the direct answer an annotator expects. BOW-Reg preserves a broader candidate-space description and is therefore less likely to verbalize one answer immediately. BOW-Core's gain over BR on comprehensiveness also indicates

	CSQA	PIQA	SQA	TQA	ARC-c	WG	BBH	MMLU	MMLU-p	GPQA	Avg.
BOW-Core	81.49	87.21	64.83	63.28	92.75	64.56	76.78	75.69	58.22	32.37	69.72
① $J_\phi :=$ `Q2.5-7B-I`	−0.49↓	−0.27↓	+0.41↑	+3.31↑	−0.69↓	+6.32↑	+0.59↑	+0.55↑	−0.32↓	−0.67↓	+0.87↑
② $J_\phi :=$ `L3.2-1B-I`	+0.16↑	+0.66↑	−0.82↓	+0.86↑	+0.34↑	+4.03↑	+0.43↑	+0.37↑	+0.02↑	−3.35↓	+0.27↑
③ $T = 2$, top-50	+1.07↑	+1.69↑	+0.81↑	+0.49↑	+0.42↑	+6.47↑	+0.59↑	+0.61↑	+0.76↑	−0.90↓	+1.20↑

Table 3: Scorer robustness for BOW-Core with a Qwen2.5-7B-Instruct policy. The default configuration uses LLaMA3.1-8B-Instruct with $T = 5$ and top-100 truncation. The BOW-Core row reports absolute scores; ablation rows report changes relative to this configuration. Q2.5-7B-I and L3.2-1B-I denote Qwen2.5-7B-Instruct and LLaMA3.2-1B-Instruct.

	CSQA	PIQA	SQA	ARC-c	WG
`Qwen2.5-7B-Instruct`					
Vanilla	84.26	85.46	50.92	80.52	67.70
SLM	79.73	80.50	59.51	75.68	55.49
ToW	87.55	87.16	57.06	80.12	67.81
BR	84.00	86.89	63.80	**83.24**	**74.92**
BOW-Reg	**88.40**	**89.06**	**64.21**	80.02	66.81
`LLaMA3.1-8B-Instruct`					
Vanilla	84.74	83.56	54.19	73.16	56.94
SLM	40.66	59.65	49.28	37.54	50.28
ToW	87.79	88.65	58.28	79.02	66.26
BR	82.30	88.86	57.06	78.00	**70.14**
BOW-Reg	**87.91**	**89.27**	**62.17**	**79.11**	64.71

Table 4: Intrinsic next-word recovery of BOW-Reg and various baselines. All scores are SC@10. Bold indicates the best result among trained methods.

that computing the core reward from the trajectory alone changes the learned reasoning behavior, even without the breadth regularizer. App. D provides representative examples from ordinary ambiguous, medical, and finance contexts.

5.4 Intrinsic Next-Word Prediction

Tab. 4 reports intrinsic next-word prediction on five benchmarks converted into multiple-choice NWP form. We evaluate only BOW-Reg because this setting assesses the full method.

BOW-Reg achieves the best result in 7 of the 10 model–benchmark settings. On LLaMA, it is the strongest method on 4 of the 5 tasks and also has the best average score. On Qwen, it leads on CSQA, PIQA, and StrategyQA, while trailing BR on ARC-Challenge and WinoGrande. A breadth-oriented objective could, in principle, make the trajectories too diffuse for accurate prediction. Here, candidate scores are computed from the generated trajectory rather than the original context, so the evaluation directly tests whether BOW-Reg's broader trajectories remain predictive representations. The results show that reducing early commitment does not remove task-relevant information: the trajectories still support accurate continuation recovery.

5.5 Scorer Robustness

Because BOW uses a frozen scorer to assign the core reward, we test whether the resulting training signal depends on the scorer configuration. We run this ablation with the Qwen2.5-7B-Instruct policy and BOW-Core, which excludes the breadth regularizer so that the comparison focuses on the core reward. Our default scorer is LLaMA3.1-8B-Instruct. In the qualitative example in Fig. 3 (App. B), both LLaMA3.1-8B-Instruct and Qwen2.5-7B-Instruct rank *water* first, but Qwen concentrates nearly all probability mass on it (0.9995), whereas LLaMA remains less concentrated (0.7575) and assigns non-negligible mass to other candidates. We therefore use LLaMA as the default because this scoring behavior better matches the goal of rewarding neutral descriptions of the plausible next-word space.

Tab. 3 compares the default configuration with three variants: a Qwen2.5-7B-Instruct scorer, a smaller LLaMA3.2-1B-Instruct scorer, and a sharper reward distribution using $T = 2$ and top-50 truncation instead of $T = 5$ and top-100. The variants change average performance by $+0.87$, $+0.27$, and $+1.20$ points, respectively, although individual benchmarks vary. Thus, the core reward is robust on average to the tested changes in scorer family, model size, and reward sharpness; it does not rely on the default scorer configuration.

6 Conclusion

We introduced BOW, an RL framework that trains next-word reasoning trajectories to describe the plausible continuation space rather than justify a single observed continuation. BOW-Core isolates the effect of computing the core reward from the generated trajectory alone, and BOW-Reg adds breadth regularization to reduce premature collapse while preserving general reasoning and intrinsic NWP performance. Across both backbones, BOW-Reg achieves the highest SharedRef correctness and the lowest HoWN-Simple collapse; in the

Qwen-based human study, it receives the highest comprehensiveness preference. The scorer ablation further shows that the core reward is robust on average across the tested scorer configurations. Together, these findings support training next-word reasoning to preserve plausible alternatives until the context justifies commitment.

Limitations

BOW targets a specific failure mode in RL-based next-word reasoning: when several continuations remain plausible, answer-centric rewards can push the model to commit too early to one favored continuation and generate reasoning that supports it. BOW should therefore not be interpreted as a general-purpose method for maximizing performance on every downstream QA benchmark. We use general QA and intrinsic NWP primarily to verify that the intervention preserves broad capability, while SharedRef, HoWN-Simple, and the human study provide more direct tests of the premature-collapse behavior that BOW is designed to address.

Our experiments cover two 7–8B instruction backbones, English ambiguity benchmarks, a Qwen-based human study, and reasoning-heavy next-word positions selected by filtering. They test the proposed failure mode in a controlled setting but do not establish how BOW behaves in larger-scale pretraining, multilingual data, specialized domains, or longer-form generation.

References

Yonatan Bisk, Rowan Zellers, Ronan Le bras, Jianfeng Gao, and Yejin Choi. 2020. Piqa: Reasoning about physical commonsense in natural language. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7432–7439.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2023. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv*, abs/1803.05457.

Maksym Del and Mark Fishel. 2023. True detective: A deep abductive reasoning benchmark undoable for GPT-3 and challenging for GPT-4. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 314–322, Toronto, Canada. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Qingxiu Dong, Li Dong, Yao Tang, Tianzhu Ye, Yutao Sun, Zhifang Sui, and Furu Wei. 2025. Reinforcement pre-training. *ArXiv*, abs/2506.08007.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, and 1 others. 2024. The llama 3 herd of models.

Lukas Ellinger, Miriam Anschütz, and Georg Groh. 2025. Simplifications are absolutists: How simplified language reduces word sense awareness in LLM-generated definitions. In *Proceedings of the 15th International Conference on Recent Advances in Natural Language Processing - Natural Language Processing in the Generative AI Era*, pages 342–351, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Lukas Ellinger and Georg Groh. 2025. It depends: Resolving referential ambiguity in minimal contexts with commonsense knowledge. In *Proceedings of the 2nd Workshop on Uncertainty-Aware NLP (UncertaiNLP 2025)*, pages 229–246, Suzhou, China. Association for Computational Linguistics.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

Abhimanyu Hans, John Kirchenbauer, Yuxin Wen, Neel Jain, Hamid Kazemi, Prajwal Singhania, Siddharth Singh, Gowthami Somepalli, Jonas Geiping, Abhinav Bhatele, and Tom Goldstein. 2024. Be like a goldfish, don't memorize! mitigating memorization in generative LLMs. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Ali Hatamizadeh, Syeda Nahida Akter, Shrimai Prabhumoye, Jan Kautz, Mostofa Patwary, Mohammad Shoeybi, Bryan Catanzaro, and Yejin Choi. 2026. RLP: Reinforcement as a pretraining objective. In

The Fourteenth International Conference on Learning Representations.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Dongwei Jiang, Guoxuan Wang, Yining Lu, Andrew Wang, Jingyu Zhang, Chuyu Liu, Benjamin Van Durme, and Daniel Khashabi. 2025. RATIONALYST: Pre-training process-supervision for improving reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26547–26566, Vienna, Austria. Association for Computational Linguistics.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, SOSP '23, page 611–626, New York, NY, USA. Association for Computing Machinery.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022. Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8424–8445, Dublin, Ireland. Association for Computational Linguistics.

Bangzheng Li, Ben Zhou, Fei Wang, Xingyu Fu, Dan Roth, and Muhao Chen. 2024. Deceptive semantic shortcuts on reasoning chains: How far can models go without hallucination? In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7675–7688, Mexico City, Mexico. Association for Computational Linguistics.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, yelong shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, and Weizhu Chen. 2024. Not all tokens are what you need for pretraining. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025. Understanding r1-zero-like training: A critical perspective. In *Second Conference on Language Modeling*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

OpenAI. 2023. Gpt-4 technical report.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2024. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8732–8740.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *ArXiv*, abs/1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junmei Song, Mingchuan Zhang, Y. K. Li, Yu Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *ArXiv*, abs/2402.03300.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2025. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, EuroSys '25, page 1279–1297, New York, NY, USA. Association for Computing Machinery.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Fei Wang, Wenjie Mo, Yiwei Wang, Wenxuan Zhou, and Muhao Chen. 2023a. A causal view of entity bias in (large) language models. In *Findings of the Association for Computational Linguistics: EMNLP*

2023, pages 15173–15184, Singapore. Association for Computational Linguistics.

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

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024. MMLU-pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Ziqi Wang, Hanlin Zhang, Xiner Li, Kuan-Hao Huang, Chi Han, Shuiwang Ji, Sham M. Kakade, Hao Peng, and Heng Ji. 2025. Eliminating position bias of language models: A mechanistic approach. In *The Thirteenth International Conference on Learning Representations*.

Ronald J. Williams. 2004. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256.

Zhikun Xu, Ming Shen, Jacob Dineen, Zhaonan Li, Xiao Ye, Shijie Lu, Aswin Rrv, Chitta Baral, and Ben Zhou. 2025. ToW: Thoughts of words improve reasoning in large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3057–3075, Albuquerque, New Mexico. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2024. Qwen2.5 technical report. *ArXiv*, abs/2412.15115.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, YuYue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, LingJun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, and 17 others. 2025. DAPO: An open-source LLM reinforcement learning system at scale. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Xiaodong Yu, Ben Zhou, Hao Cheng, and Dan Roth. 2024. Reasonagain: Using extractable symbolic programs to evaluate mathematical reasoning. *ArXiv*, abs/2410.19056.

Eric Zelikman, Georges Raif Harik, Yijia Shao, Varuna Jayasiri, Nick Haber, and Noah Goodman. 2024. Quiet-STar: Language models can teach themselves to think before speaking. In *First Conference on Language Modeling*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. STar: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*.

Congzhi Zhang, Linhai Zhang, Jialong Wu, Yulan He, and Deyu Zhou. 2025. Causal prompting: Debiasing large language model prompting based on front-door adjustment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(24):25842–25850.

Ben Zhou, Hongming Zhang, Sihao Chen, Dian Yu, Hongwei Wang, Baolin Peng, Dan Roth, and Dong Yu. 2024. Conceptual and unbiased reasoning in language models. *ArXiv*, abs/2404.00205.

A Reward-Path Bottleneck Analysis of the Core BOW Reward

Motivation. A central goal of the core BOW reward is to make the generated trajectory τ responsible for credit assignment in next-word learning. The policy still conditions on C, so τ may restate or summarize information from C. In the core scoring pass, however, the scorer receives only τ, without receiving C as a separate input. This appendix formalizes why this design pressures τ to remain predictive of the next token, whereas joint scoring from (C, τ) can bypass the trajectory. The analysis concerns the core reward in Eq. 1, which BOW-Reg retains; the breadth term in Eq. 2 is outside its scope.

A.1 Setup and Notation

Let $C := x_{1:t}$ denote the context and let $Y := x^*_{t+1}$ denote the gold next token (or first token of the gold next word, consistent with our implementation). Let τ be the next-word reasoning trajectory sampled from the policy:

$$\tau \sim \pi_\theta(\cdot \mid C).$$

Since τ is generated from C and does not influence the data-generating process of Y, we have the Markov structure

$$\tau \leftarrow C \rightarrow Y,$$
$$\text{equivalently} \qquad I(Y; \tau \mid C) = 0.$$

We first contrast two scoring schemes:

Core scoring from the trajectory alone (BOW-Core). A frozen scorer J_ϕ receives only τ and outputs a next-token distribution $P_{J_\phi}(\cdot \mid \tau)$. The core BOW reward is

$$r_{\text{BOW}}(\tau; Y) := P_{J_\phi}(Y \mid \tau).$$

Joint context-and-trajectory scoring. A probabilistic scorer $\tilde{J}_\phi$ receives the original context C in addition to τ, and assigns reward via

$$r_{\text{CTX}}(C, \tau; Y) := \log P_{\tilde{J}_\phi}(Y \mid C, \tau).$$

This family captures reward designs in which the evaluator receives both C and τ and can in principle ignore τ.

A.2 Predictive Sufficiency of the Core BOW Reward

The core BOW reward targets predictive sufficiency rather than minimal or lexically compressed representations. Because the core scoring pass receives only τ, reward depends on whether τ preserves information useful for predicting Y. This does not require τ to be context-free, information-minimal, or lexically disjoint from C.

For analysis, we use a log-likelihood-shaped surrogate of the core reward,

$$\tilde{r}_{\text{BOW}}(\tau; Y) := \log P_{J_\phi}(Y \mid \tau),$$

which is used only in this section.

Theorem 1 (Predictive sufficiency under a log-reward surrogate)**.** *Assume the scorer is approximately calibrated in the sense that*

$$P_{J_\phi}(Y \mid \tau) \approx P(Y \mid \tau).$$

Then maximizing the expected shaped reward

$$\max_\theta \; \mathbb{E}_{(C,Y)} \, \mathbb{E}_{\tau \sim \pi_\theta(\cdot|C)} \big[\log P_{J_\phi}(Y \mid \tau) \big]$$

approximately minimizes the conditional entropy $H(Y \mid \tau)$, equivalently maximizes the mutual information $I(\tau; Y)$.

Moreover, under the Markov structure $\tau \leftarrow C \rightarrow Y$, we have the identity

$$I(Y; C \mid \tau) = H(Y \mid \tau) - H(Y \mid C),$$

so minimizing $H(Y \mid \tau)$ also minimizes the remaining predictive information in C beyond τ. This conditional mutual information is the predictive-sufficiency gap: it vanishes exactly when

$$I(Y; C \mid \tau) = 0 \quad \Longleftrightarrow \quad P(Y \mid C) = P(Y \mid \tau).$$

Proof sketch. If $P_{J_\phi}(Y \mid \tau) = P(Y \mid \tau)$, then

$$\begin{aligned} \mathbb{E}[\log P_{J_\phi}(Y \mid \tau)] &= \mathbb{E}[\log P(Y \mid \tau)] \\ &= -H(Y \mid \tau). \end{aligned}$$

Thus maximizing the expected shaped reward is equivalent to minimizing $H(Y \mid \tau)$ and therefore maximizing

$$I(\tau; Y) = H(Y) - H(Y \mid \tau).$$

Next, using $\tau \leftarrow C \rightarrow Y$ (equivalently $I(Y; \tau \mid C) = 0$), we have $H(Y \mid C, \tau) = H(Y \mid C)$. Therefore

$$\begin{aligned} I(Y; C \mid \tau) &= H(Y \mid \tau) - H(Y \mid C, \tau) \\ &= H(Y \mid \tau) - H(Y \mid C). \end{aligned}$$

Since $H(Y \mid C)$ is fixed by the data distribution, minimizing $H(Y \mid \tau)$ directly minimizes $I(Y; C \mid \tau)$. The achievable minimum depends on the policy class; a zero gap corresponds to predictive sufficiency, $P(Y \mid C) = P(Y \mid \tau)$.

Interpretation. Because the core scoring pass receives only τ, improving the core reward requires τ to preserve information useful for predicting Y. Under the assumptions above, the objective encourages predictive sufficiency. This is not a claim that τ excludes or minimally encodes information from C.

A.3 Two Reward-Bypass Mechanisms

We consider two ways in which the learning signal for τ can be bypassed. Under joint context-and-trajectory scoring, the evaluator can predict Y directly from C. Under answer-only rewards, the policy can predict $\hat{y}$ from C without using τ. In either case, improving reward need not make the trajectory predictive.

Theorem 2 (Reward bypass under joint context-and-trajectory likelihood scoring)**.** *Assume a Bayes-optimal joint scorer, so that*

$$P_{\tilde{J}_\phi}(Y \mid C, \tau) = P(Y \mid C, \tau).$$

Under $\tau \leftarrow C \rightarrow Y$, we have

$$P(Y \mid C, \tau) = P(Y \mid C),$$

hence

$$\begin{aligned} r_{\text{CTX}}(C, \tau; Y) &= \log P_{\tilde{J}_\phi}(Y \mid C, \tau) \\ &= \log P(Y \mid C). \end{aligned}$$

The resulting reward is independent of τ. Consequently, the policy-gradient learning signal for shaping $\pi_\theta(\tau \mid C)$ collapses:

$$\mathbb{E}_{\tau \sim \pi_\theta(\cdot|C)} \Big[\nabla_\theta \log \pi_\theta(\tau \mid C) \, r_{\text{CTX}}(C, \tau; Y) \Big] = 0.$$

Proof sketch. Because τ is generated from C and does not affect Y, we have $Y \perp \tau \mid C$, and thus $P(Y \mid C, \tau) = P(Y \mid C)$. Therefore the reward is constant with respect to τ for fixed (C, Y). The score-function identity then gives

$$\mathbb{E}_\tau[\nabla_\theta \log \pi_\theta(\tau \mid C)] = 0,$$

so multiplying by a constant reward yields zero expected gradient.

Interpretation. If the evaluator separately receives C, it can recover the next-token distribution without using τ. The trajectory is then no longer necessary for reward, so optimization need not increase $I(\tau; Y)$.

Theorem 3 (Reward bypass under answer-only rewards)**.** *A policy π_θ generates a reasoning trajectory τ and then a predicted next token or word $\hat{y}$:*

$$(\tau, \hat{y}) \sim \pi_\theta(\cdot \mid C),$$
$$\pi_\theta(\tau, \hat{y} \mid C) = \pi_\theta(\tau \mid C)\, \pi_\theta(\hat{y} \mid C, \tau).$$

Consider any answer-only reward $g(\hat{y}, Y)$ that depends on the predicted answer $\hat{y}$ and target Y but does not directly score τ. This includes BR, where $g(\hat{y}, Y) = \mathbf{1}\{\hat{y} = Y\}$.

Define the rollout objective

$$J(\theta) := \mathbb{E}_{(\tau,\hat{y})\sim\pi_\theta(\cdot|C)}[g(\hat{y}, Y)]$$

under the data distribution of (C, Y) and the policy. Then the policy gradient decomposes as

$$\nabla_\theta J(\theta) = \mathbb{E}_{\substack{(\tau,\hat{y})\sim \\ \pi_\theta(\cdot|C)}}\big[g(\hat{y}, Y)\nabla_\theta \log \pi_\theta(\hat{y} \mid C, \tau)\big] + \mathbb{E}_{\substack{\tau\sim \\ \pi_\theta(\cdot|C)}}\big[Q_\theta(C, \tau)\nabla_\theta \log \pi_\theta(\tau \mid C)\big].$$

Here, the post-τ value is

$$Q_\theta(C, \tau) := \mathbb{E}_{\hat{y}\sim\pi_\theta(\cdot|C,\tau)}[g(\hat{y}, Y)].$$

Moreover, if the policy reaches (or approximates) a bypass regime in which the answer distribution becomes τ-invariant given C, i.e.

$$\pi_\theta(\hat{y} \mid C, \tau) \approx \pi_\theta(\hat{y} \mid C),$$
for all τ in the policy support,

then $Q_\theta(C, \tau) \approx Q_\theta(C)$ is approximately constant in τ, and the learning signal specifically for shaping the reasoning policy collapses:

$$\begin{aligned}&\mathbb{E}_{\tau\sim\pi_\theta(\cdot|C)}\big[Q_\theta(C)\nabla_\theta \log \pi_\theta(\tau \mid C)\big] \\ &\quad = Q_\theta(C)\, \mathbb{E}_{\tau\sim\pi_\theta(\cdot|C)}\big[\nabla_\theta \log \pi_\theta(\tau \mid C)\big] \\ &\quad = 0.\end{aligned}$$

The answer term

$$\mathbb{E}\big[g(\hat{y}, Y)\nabla_\theta \log \pi_\theta(\hat{y} \mid C, \tau)\big]$$

can remain nonzero and continue improving prediction accuracy.

Finally, because the objective depends on τ only through its effect on $\pi_\theta(\hat{y} \mid C, \tau)$, there exist reward-optimal solutions in which $\hat{y}$ implements a Bayes-optimal predictor from C (under g) while τ is arbitrary, even independent of C. Hence maximizing $J(\theta)$ does not in general force τ to become an information-sufficient statistic for Y.

Proof sketch. By the policy gradient,

$$\nabla_\theta J(\theta) = \mathbb{E}_{\substack{(\tau,\hat{y})\sim \\ \pi_\theta(\cdot|C)}}\big[g(\hat{y}, Y)\nabla_\theta \log \pi_\theta(\tau, \hat{y} \mid C)\big].$$

Using the autoregressive factorization

$$\log \pi_\theta(\tau, \hat{y} \mid C) = \log \pi_\theta(\tau \mid C) + \log \pi_\theta(\hat{y} \mid C, \tau),$$

we obtain the stated decomposition. In the bypass regime where $\pi_\theta(\hat{y} \mid C, \tau) \approx \pi_\theta(\hat{y} \mid C)$, the post-$\tau$ value becomes approximately constant in τ, so the reasoning-gradient term vanishes by the score-function identity, while the answer-gradient term can remain nonzero and continue improving the answer distribution.

Interpretation. BR-style training can improve next-token prediction through the answer term without making τ self-contained or causally responsible. Once the policy predicts $\hat{y}$ from C without relying on τ, the gradient shaping $\pi_\theta(\tau \mid C)$ can vanish and the trajectory can become a decorative byproduct. In BOW-Core, the core scorer does not receive C as a separate input, so the reward depends on what is present in τ.

A.4 Scope of the Bottleneck Claim

The analysis establishes a structural property: the core scoring pass cannot predict directly from C while ignoring τ. It does not require τ to be context-free or to avoid restating contextual information. Any context-derived evidence used by the core scorer must first be present in τ. This is the reward-path bottleneck claimed by BOW.

B Why Soft Partial-Credit Rewards Give Finer Credit Than Binary Hit/Miss Rewards

This appendix provides an informal justification for why the *core* BOW reward in Eq. 1 can yield denser

	Original Instance	Transformed Instance
PIQA	*Question*: Choose the most sensible solution given the physical goal: To cream butter and sugar together, you can *Options*: (a) Place it in a bowl and use a hand warmer (b) Place in a bowl and use a hand mixer	*Context*: To cream butter and sugar into a light, fluffy mixture, place them in a bowl and use a hand *Candidate Next Words*: (a) warmer (b) mixer
CSQA	*Question*: Where could you find a shark before it was caught? *Options*: (a) pool hall (b) tomales bay (c) marine museum (d) business (e) desert	*Context*: Before being caught, a shark naturally lives in the *Candidate Next Words*: (a) hall (b) bay (c) museum (d) business (e) desert

Table 5: Examples of the multiple-choice-to-next-word transformation.

learning signal than a binary exact-match reward such as BR. The discussion concerns BOW-Core; the auxiliary BOW-Reg shaping term in Eq. 2 is omitted for clarity.

B.1 Setup and Notation

Let $x_{1:t}$ denote the context and let $x^*_{t+1} \in V$ denote the gold next token, where V is the model vocabulary. The policy samples a reasoning trajectory

$$\tau \sim \pi_\theta(\cdot \mid x_{1:t}).$$

We compare two reward types:

Binary hit/miss reward. The model predicts a next token or word $\hat{x}_{t+1}$ and receives

$$R_{\text{BINARY}} = \mathbf{1}\{\hat{x}_{t+1} = x^*_{t+1}\} \in \{0, 1\}.$$

Soft partial-credit reward. A frozen scorer that observes only the reasoning trajectory assigns

$$R_{\text{SOFT}} = P_{J_\phi}(x^*_{t+1} \mid \tau) \in [0, 1].$$

The corresponding information flow can be summarized informally as:

$$x_{1:t} \xrightarrow{\pi_\theta} (\tau, \hat{x}_{t+1}) \xrightarrow{\mathbf{1}\{\hat{x}_{t+1}=x^*_{t+1}\}} R_{\text{BINARY}},$$

versus

$$x_{1:t} \xrightarrow{\pi_\theta} \tau \xrightarrow{P_{J_\phi}(x^*_{t+1}|\tau)} R_{\text{SOFT}}.$$

B.2 Soft Rewards Preserve Partial Credit

Unlike binary supervision, the soft reward can assign positive and graded credit before the exact gold token is identified. Whenever $P_{J_\phi}(x^*_{t+1} \mid \tau) > 0$, the trajectory receives positive reward. A trajectory that narrows the continuation to a plausible class can therefore receive partial credit without identifying the exact token, provided the scorer retains nonzero mass on the gold token. Binary exact-match supervision assigns zero reward unless the model predicts the gold token exactly.

Because the reward equals $P_{J_\phi}(x^*_{t+1} \mid \tau)$, it also preserves the ordering of partially informative trajectories: a trajectory for which the scorer assigns more probability to the gold token. Binary supervision cannot distinguish two trajectories when both miss the exact token.

B.3 Implication for Exploration

Soft rewards provide nonzero and graded feedback on a broader set of rollouts, which can reduce dependence on rare exact-match successes. This is an argument about the granularity of credit assignment, not a guarantee that soft rewards dominate binary rewards in every optimization regime.

C More Experimental Details

C.1 Training Data Filtering

We filter the training data to remove context–next-word pairs whose next token can be derived without meaningful reasoning over the context. Specifically, we discard tokens that are: (i) purely functional (e.g., determiners, punctuation), (ii) syntactically or semantically deterministic based on surface cues, or (iii) explainable without invoking latent knowledge or contextual abstraction. This filtering strategy is inspired by selective language modeling methods such as RHO-1 (Lin et al., 2024), which demonstrates that focusing training on informative or "reasoning-heavy" tokens improves learning efficiency and model generalization. We use `gpt-4.1-mini-2025-04-14`[5] to evaluate each context–next-word pair using these criteria. Fig. 7 shows the filtering prompt. We retain only pairs whose next word requires non-trivial reasoning.

C.2 Targeted Ambiguity Evaluation Details

For SharedRef, we evaluate on the English SharedRef split under the Normal prompt. This setting preserves referential ambiguity while avoiding the additional simplified-language constraint. We report the benchmark's official judge-based Correct-

[5] https://openai.com/index/gpt-4-1/

```
The next word set will likely contain nouns that represent beverages, as the runner is at an aid station and is likely to grab a
drink to rehydrate.

The set of possible next words can be described as:
- Water
- Sports drink
- Juice
- Electrolyte solution

These are the most common types of beverages found at aid stations during marathons, designed to help runners rehydrate and
replenish electrolytes lost through sweat.
```

		water	*sports*	*electro*	*juice*	*drink*	*energy*
LLaMA3.1-8B-Instruct	**Rank**	1	5	7	10	11	16
	Prob.	0.7575	0.0228	0.0074	0.0024	0.0011	0.0007
Qwen2.5-7B-Instruct	**Rank**	1	2	N/A	13	N/A	N/A
	Prob.	0.9995	0.0004	N/A	0.0000	N/A	N/A

Figure 3: This example shows that several expected next tokens appear near the top of $P_{J_\phi}(\cdot \mid \tau)$. Given the context *The marathon runner felt dizzy from dehydration, so at the aid station she grabbed a bottle of*, the trajectory shown above supports beverage-related continuations. Both scorers rank *water* first; Qwen2.5-7B-Instruct concentrates nearly all probability mass on it (0.9995), whereas LLaMA3.1-8B-Instruct remains less concentrated (0.7575) and assigns non-negligible mass to additional plausible candidates.

ness metric. For HoWN-Simple, we evaluate English homonyms under the Simple prompt, which asks for definitions in simple language without the Multi-Sense-Aware hint. We report Covered and Collapse. Covered is the percentage of coarse-grained WordNet senses covered by the response. Collapse is the rate of responses that provide only one definition without helpful sense awareness. We omit Complete, FKGL, and other HoWN diagnostics from the main table because Covered measures actual sense coverage, while Collapse directly measures the premature-commitment failure mode targeted by BOW.

C.3 NWP Format Transformation

We use `gpt-4.5-preview-2025-02-27`[6] to jointly transform each multiple-choice QA instance into a context and a set of single-word candidate continuations, with one gold candidate. Each candidate must complete the context naturally, and the transformed task must preserve the original knowledge, reasoning process, and approximate difficulty. We then use GPT-4.5 to filter out invalid transformations. Figs. 9 and 10 show the transformation and validation prompts.

C.4 More Optimization Details

We conduct RL training on four NVIDIA H200 GPUs using the VeRL[7] (Sheng et al., 2025) repository. We use the AdamW (Loshchilov and Hutter, 2019) optimizer with an initial learning rate of 1×10^{-6}, $(\beta_1, \beta_2) = (0.9, 0.999)$, and a weight decay of 1×10^{-2}.

For representative Qwen2.5-7B-Instruct runs, one epoch on four NVIDIA H200 GPUs took approximately 6 wall-clock hours for BR, 9 for BOW-Core, and 12 for BOW-Reg, corresponding to about 24, 36, and 48 H200 GPU-hours, respectively. BOW-Core and BOW-Reg therefore used roughly $1.5\times$ and $2.0\times$ the GPU-hours of BR. The additional cost comes from frozen-scorer inference over the sampled trajectories; BOW-Reg also computes the context-based reference distribution used by the breadth regularizer.

C.5 More Baseline Details

For fair comparison, all trainable baselines use the same filtered supervision positions as BOW.

ToW. For ToW annotations, we use `gpt-4o-2024-11-20`.[8] Following Xu et al. (2025), we provide the context, three in-context examples, and the gold next word, then ask the model to generate a thought of about 20 words. We annotate the same supervision positions used for RL rather than applying the original three-stage generation, filtering, and shortening pipeline. As in prior work, the model may state that the next word is unpredictable from the context.

BR. BR is designed to match BOW as closely as possible while remaining answer-centric. The

[6] https://openai.com/index/introducing-gpt-4-5/

[7] https://github.com/volcengine/verl

[8] https://openai.com/index/hello-gpt-4o/

	CSQA	PIQA	SQA	TQA	ARC-c	WG	BBH	MMLU	MMLU-p	GPQA	AVG.
					Qwen2.5-7B-Instruct						
BOW-Reg	81.90	88.68	63.19	66.83	92.32	70.96	77.21	76.51	57.51	33.26	70.84
w/ random filt.	0.00	−0.84↓	−1.02↓	−1.22↓	+0.60↑	−0.24↓	+0.21↑	−0.34↓	+0.27↑	0.00	−0.26↓
					LLaMA3.1-8B-Instruct						
BOW-Reg	77.97	86.67	67.28	60.22	89.16	59.75	74.56	73.59	52.71	32.14	67.41
w/ random filt.	+0.41↑	−0.87↓	−0.20↓	−0.86↓	+0.35↑	+0.94↑	−3.01↓	−1.41↓	−3.68↓	−2.45↓	−1.08↓

Table 6: Effect of reasoning-centric training-data filtering. BOW-Reg rows report absolute scores; random-filtering rows report changes relative to BOW-Reg.

policy is prompted to reason about the most likely next word and then output a single predicted next word wrapped in `\boxed{}`; the exact prompt is shown in Fig. 5. Reward is 1 if the first token of the boxed word matches the first token of the gold next word, and 0 otherwise.

IGR. Because the original RLP implementation is not publicly available, we implement a simplified information-gain baseline. We use the initial policy model as a frozen reference scorer and compute reward as the difference between the log-evidence assigned to the gold next token with reasoning and the corresponding baseline log-evidence without reasoning.

Intrinsic NWP evaluation details. BR directly emits a boxed next word rather than a candidate-scored distribution. For a fair comparison in intrinsic NWP evaluation, we remove the boxed answer and score the benchmark candidate continuations from the preceding reasoning prefix.

D Human Evaluation Details and Representative Examples

This appendix describes the curation and annotation of the 150 human-evaluation contexts, followed by three representative examples of premature collapse from a broader plausible next-word space to a single favored continuation.

D.1 Context Curation

We curate 150 contexts in which several immediate next words are superficially plausible but contextual reasoning narrows the continuation space. GPT-4.5 selects contexts with two properties: (i) several continuations are plausible from local statistical, grammatical, or commonsense cues, and (ii) broader contextual reasoning narrows the space to one or a few especially plausible candidates. The selected next words include nouns, verbs, and adjectives.

D.2 Annotation Protocol

We compare four Qwen-based models: the vanilla instruction model, BR, BOW-Core, and BOW-Reg. For each curated context, we sample one next-word reasoning trajectory from each model. Annotators are shown the original context together with four anonymized and randomly ordered trajectories. Each set of trajectories is evaluated along two dimensions:

1. **Comprehensive next-word reasoning.** Which trajectory best reasons over the plausible next-word space in a self-contained, neutral, and comprehensive way, using clues expressed or implied by the context without collapsing too early to a narrow answer?

2. **Human-expected next-word description.** Which trajectory's next-word description best matches the annotator's own expected continuation?

Ties are allowed when no clear winner is apparent, so the reported preference rates do not sum to 100.

D.3 Interpreting the Annotation Dimensions

The first dimension directly measures the trajectory form targeted by BOW. The second is diagnostic: it measures whether the trajectory states the continuation an annotator expects most directly. A trajectory may score highly on the second dimension by committing early, while another may score more highly on comprehensiveness by preserving a broader candidate set.

D.4 Representative Examples

Tabs. 7, 8, and 9 illustrate the qualitative differences behind the quantitative results. In these tables, the original context is shown alongside the trajectories only for the reader. The policy conditions on that context when generating each self-contained trajectory, so a trajectory may restate

parts of it; the core scoring pass receives only the trajectory.

Ordinary ambiguous context. For the context *The thunderstorm was getting closer, so I rolled up the . . .*, Vanilla and BR commit quickly to *window*. BOW-Core names a somewhat broader but still explicit set, while BOW-Reg remains less committed and describes a wider range of weather- and roll-up-related possibilities. Its broader wording also helps explain its lower preference on expected next-word description.

Medical example. In Tab. 8, BR and BOW-Core move quickly toward *antibiotics* or *antibiotic*. BOW-Reg stays at a broader treatment-related level rather than immediately committing to one continuation.

Finance example. In Tab. 9, BR and BOW-Core commit to *transfers*. BOW-Reg discusses broader concepts around control transfer instead of treating that continuation as uniquely licensed.

Across the three examples, BOW-Core is more willing to state a likely continuation directly, while BOW-Reg more strongly resists early commitment.

E Effect of Training Data Filtering

Tab. 6 compares the reasoning-centric filter with random filtering that retains the same number of supervision positions. Random filtering lowers average performance by 0.26 points on Qwen2.5-7B-Instruct and 1.08 points on LLaMA3.1-8B-Instruct. The filter is not the main source of BOW's gains; it concentrates updates on contexts where next-word reasoning is needed.

F Use of AI Assistants

We used AI assistants to polish language and presentation throughout the manuscript based on author-provided research ideas, experimental design, analyses, and results. AI assistants were not used to originate the research questions or generate plots; the authors reviewed and take responsibility for all content.

As part of the experimental pipeline, we used `gpt-4.1-mini-2025-04-14` for training-data filtering, `gpt-4.5-preview-2025-02-27` for evaluation-data transformation and human-evaluation context curation, and `gpt-4o-2024-11-20` for ToW annotations. We also used LLM-based judging for SharedRef and HoWN-Simple, as described above.

Policy rollout prompt

```
The next word prediction problem of language modeling generally involves first searching for a comprehensive but small enough set of possible next words, and then estimating each possible word's probability within this set. Given a specific context, your job is to output a reasoning process that generally follows this illustration.

Here is one important rule you MUST follow. Since different candidate sets are different in size, instead of enumerating all possible words, you must be smart and innovative enough to describe the word set in concise and unbiased natural language, without missing any possibilities. This is particularly important when the smallest candidate set you can find is still not small or concrete enough. For example, given the context "I [next-word]", the best you can do is to tell me that the next word set will likely contain verbs, and those verbs should be actions performed by humans rather than other verbs. You should not give a concrete list of words since it is impossible to enumerate all the verbs humans can perform.

However, in other contexts where information is rich, you should perform more in-depth reasoning. For example, given the context "I like fruit, so as lunch I had two [next-word]", you should reason that the next word set will likely contain nouns of fruit given the logical clue that "I like fruit" and the quantitative adjective "two" in the context. Furthermore, you should reason that this set is more likely to contain larger fruits to fill me up for lunch instead of small fruit. In other scenarios, such as "1+1=[next-word]", where the next word set can be small enough to be listed out, you should actually list out all possible words in the set you find.

Now, given the below context, please reason on possible next words following all the rules described above. Make your output self-contained without needing to reference the given context.
Context:
{context}
```

Figure 4: Policy rollout prompt.

Binary-reward rollout prompt

```
The next word prediction problem of language modeling generally involves first searching for a comprehensive but small enough set of possible next words, and then estimating each possible word's probability within this set to pick the most likely next word. Given a specific context, your job is to output a reasoning process that generally follows this illustration to predict the most likely next word.

Here is one important rule you MUST follow. Since different candidate sets are different in size, instead of enumerating all possible words, you must be smart and innovative enough to describe the word set in concise and unbiased natural language, without missing any possibilities. This is particularly important when the smallest candidate set you can find is still not small or concrete enough. For example, given the context "I like fruit, so as lunch I had two [next-word]", you should reason that the next word set will likely contain nouns of fruit given the logical clue that "I like fruit" and the quantitative adjective "two" in the context. Furthermore, you should reason that this set is more likely to contain larger fruits to fill me up for lunch instead of small fruit. Then you should choose the most likely next word based on the set description.

Now, given the below context, please follow the rule described above to first reason and then wrap your predicted most likely next word within \boxed{}. Make your output self-contained without needing to reference the given context.
Context:
{context}
```

Figure 5: Binary-reward rollout prompt.

Frozen-scorer prompt

```
Task Instruction: Given a thought for the next word of a certain context, you need to predict this next word.

Now please give me your prediction of the next word based on the following thought:

{thought}

Next Word:
```

Figure 6: Frozen-scorer prompt.

Training-data filtering prompt

```
Given a context and its completion, you need to decide if the context's immediate next word, which is the first word in the completion, requires non-trivial reasoning to derive. Below are two examples:

<Context>
During her last visit, my mom gave me a pair of socks and a swimsuit. ......(words omitted here). This winter, to stay warm, I put on a pair of
</Context>
<Completion>
socks that my mom gave me last time.
</Completion>
In this example, the immediate next word to the context is "socks", and it requires non-trivial reasoning to derive. It is common sense to know that when people feel cold, they will put on warm clothes to keep warm. Some possible next words to the context could be "jeans", "socks", or "gloves", etc. However, from the context, we know that mom gave me a pair of socks last time, so it is more likely for me to put on a pair of socks instead of gloves or jeans to keep warm, since it is known that I have this pair of socks.

<Context>
During her last visit, my mom gave me a pair
</Context>
<Completion>
of socks and a swimsuit.
</Completion>
In this example, the immediate next word to the context is "of", and it does not require reasoning to derive. Given "a pair", English grammar already tells us the next word is highly likely to be "of" to complete the phrase "a pair of".

The two examples above are simplified examples to illustrate your goal; the context you are about to see is longer and more complex. Now, I am providing you with a context and its completion. Please decide if the immediate next word to the context, which is the first word in the completion, requires non-trivial or complex reasoning to derive:
<Context>
{context}
</Context>
<Completion>
{completion}
</Completion>

Note that you must respond in the format of a JSON object with two keys. The first key is named "requires_reasoning" with Boolean type to decide if reasoning is required. True means required, and false means not required. The second key is called "explanation" with String type, and it should record your explanation of why or why not non-trivial reasoning is needed.
```

Figure 7: Training-data filtering prompt.

MCQA inference prompt

```
Question: {question}
Options: {options}

You should ONLY choose the letters from the options and answer in the format of 'So the answer is X', where X is the option letters A, B, C, etc. Please think step by step.
```

Figure 8: MCQA inference prompt.

Evaluation transformation prompt

```
# Task: Given a question, its candidate answers, and the correct answer, your task is to transform them into a final word prediction problem.

# Transformation rules:
1. The transformation output should contain a transformed context, a set of candidate final words (must be WORD instead of phrase), and the gold final word, which is much more likely to conclude the transformed context than other candidate final words.
2. Concatenating each candidate final word to the transformed context must be able to form a COMPLETE sentence. This complete sentence must be NATURAL and FLUENT.
3. The transformed final word prediction problem must maintain the SAME CONTEXT from the original question, so that it can evaluate the SAME or roughly the same KNOWLEDGE and REASONING process from the original question at the SAME DIFFICULTY level.

# Output format requirements:
You must output in JSON format with three keys. The first key is named “transformed_context” with the data type of string to store the transformed context. The second key is named “final_word_candidates” with the data type of a list of strings to store the transformed candidate final words. The third key is named “ gold_final_word_index” with the data type of integer to store the gold final word index in the “final_word_candidates” list.

Now, you are given the following question, its candidate answers, and the gold answer. Please follow the above transformation rules, example transformations, and output format requirements to perform the transformation.

<Original Question>
{question}
</Original Question>
<Original Candidate Options>
{options}
</Original Candidate Options>
<Original Ground Truth>
{ground_truth}
</Original Ground Truth>
```

Figure 9: Evaluation transformation prompt.

Transformation validation prompt

```
# Task: Last word prediction is the task of predicting the last word of a given context. Given an instance of last word prediction, there is a context, a set of candidate last words for the context, and the gold last word. Your task is to verify the quality of this instance based on the following verification dimensions:

# Verification rules:
1. You need to ensure that the gold last word is more plausible than other candidates without any ambiguity.
2. All the candidate’s last words must be able to be directly concatenated to the context to form a semantically complete sentence.
3. The semantically complete sentence from #2 must be semantically fluent and as natural as possible, just like how we humans would talk in real life. Minor grammatical errors are acceptable.

# Output format rules:
You must output in JSON format with two keys. The first key is named “is_valid” with the data type of boolean to store whether the given instance is valid based on the verification rules. The second key is called “rationale” with the data type of string to store the rationale of why or why not the instance is valid.

Now, given a last word prediction instance and the above verification rules, the instance is not valid if a single rule is violated. Please verify this instance:

<Context>
{context}
</Context>
<Candidate last words>
{candidate_last_words}
</Candidate last words>
<Gold last word>
{gold_last_word}
</Gold last word>
```

Figure 10: Transformation validation prompt.

Context	The thunderstorm was getting closer, so I rolled up the
Vanilla	Given the context "The thunderstorm was getting closer, so I rolled up the," the next word set will likely contain nouns that refer to parts of a window or curtains, as the action of rolling up suggests something that can be rolled. The most probable nouns in this context are "window" and "curtains," as these are the most common items that one would roll up to protect oneself from a thunderstorm. Therefore, the concise and unbiased set of possible next words can be described as "window" or "curtains."
BR	Reasoning: To predict the most likely next word, we need to consider the context provided: "The thunderstorm was getting closer, so I rolled up the". Given that a thunderstorm is approaching, the action of rolling something up is likely in response to this weather event. The most probable object to be rolled up in this context would be a window or a door, as these are common actions taken to protect oneself from a storm. Thus, the most likely next word set would be: "window" or "door". \boxed{window}
BOW-Core	Given the context "The thunderstorm was getting closer, so I rolled up the [next-word]", the next word in the sentence could be "window," "car window," "shade," "curtain", or "blanket," among others. The most common and logical next word in this context would be "window" or "car window," as it continues the phrase "rolled up the" in a way that makes sense in the context of a thunderstorm. The phrase "rolled up the window" is a common expression used when someone closes a car window by rolling it up. Other possibilities like "shade" or "curtain" are also plausible in certain contexts, such as rolling up a window shade or a curtain, but "window" is the most direct and commonly used option. In concise and unbiased natural language, the possible next words for "The thunderstorm was getting closer, so I rolled up the [next-word]" can be "window," "car window," "shade," "curtain", or similar items that can be rolled up, such as a blanket or a curtain. However, "window" is the most likely and commonly used next word in this context.
BOW-Reg	Given the context "The thunderstorm was getting closer, so I rolled up the," the next word set can be reasoned as follows: 1. **Continuation of the sentence**: The sentence is describing an action related to a thunderstorm, so the next words are likely to continue the narrative or provide more details about the situation. 2. **Context clues**: The phrase "The thunderstorm was getting closer, so I rolled up the" suggests that the speaker is taking some action in response to the approaching thunderstorm. The word "rolled up" is incomplete and could be referring to something that can be rolled up, such as a window, a curtain, or a tent. 3. **Topic relevance**: Since the context mentions a thunderstorm, the next words are likely to be related to weather-related actions or objects, or they might provide more details about the setting or the speaker's actions. Based on these observations, the next word set could likely contain: - Nouns or noun phrases related to weather or outdoor activities (e.g., window, curtain, tent, car window) - Nouns or noun phrases related to the action of rolling up (e.g., window, curtain, tent) - Nouns or noun phrases related to the setting or the speaker's environment (e.g., car, door, side) - Phrases or sentences that continue the narrative or provide more details about the thunderstorm or the speaker's actions Therefore, a concise and unbiased description of the next word set could be: "The next word set will likely contain nouns or noun phrases related to weather, outdoor activities, the action of rolling up, the setting or the speaker's environment, or phrases that continue the narrative or provide more details about the thunderstorm or the speaker's actions."

Table 7: A representative example from the 150 contexts curated in Sec. 5.3. The backbone model is Qwen2.5-7B-Instruct.

Context	In the ED with fever, hypotension, and suspected sepsis; after blood cultures, initiate broad-spectrum
Vanilla	Given the context "In the ED with fever, hypotension, and suspected sepsis; after blood cultures, initiate broad-spectrum", the next word set will likely contain nouns or adjectives that describe antibiotics or antimicrobial agents, as these are typically used in such clinical scenarios to treat suspected infections, especially in cases of sepsis. The reasoning is as follows: 1. The context mentions "fever, hypotension, and suspected sepsis," which indicate a severe infection requiring urgent medical intervention. 2. "After blood cultures" suggests that the healthcare provider is awaiting the results of the cultures before proceeding with treatment. 3. "Initiate broad-spectrum" indicates that the next word should describe an antibiotic or antimicrobial agent that covers a wide range of potential pathogens. Therefore, the next word set will likely include terms such as "antibiotics," "antimicrobials," or specific names of broad-spectrum antibiotics like "ciprofloxacin," "meropenem," "vancomycin," etc. However, since listing all possible broad-spectrum antibiotics would be impractical, we can describe the set concisely as "broad-spectrum antibiotics or antimicrobial agents."
BR	Reasoning: Given the context "In the ED with fever, hypotension, and suspected sepsis; after blood cultures, initiate broad-spectrum", the most likely next word is "antibiotics". This is because in a medical setting where a patient presents with fever, hypotension, and suspected sepsis, the standard protocol is to start broad-spectrum antibiotics immediately after obtaining blood cultures. The phrase "broad-spectrum" indicates that the next word should be a noun that can be used in a medical context to treat a wide range of bacterial infections, and "antibiotics" is the most appropriate and commonly used term in such a scenario. \boxed{antibiotics}
BOW-Core	Given the context "In the ED with fever, hypotension, and suspected sepsis; after blood cultures, initiate broad-spectrum [next-word]", the next word set is likely to be a noun that describes a type of antibiotic or treatment. The phrase "broad-spectrum" indicates that the next word should be a noun that refers to a medication capable of treating a wide range of bacterial infections, which is common in cases of suspected sepsis. The most appropriate and commonly used next word in this context would be "antibiotic." This noun fits the logical flow of the sentence, as it is the typical next step in managing a patient with suspected sepsis after obtaining blood cultures. While "antibiotic" is the most direct and common choice, the set of possible next words could also include other medical terms that are relevant in this context, such as "therapy," "antimicrobial," or "treatment." However, these are less specific and less commonly used in the immediate context of initiating a broad-spectrum approach in the emergency department. Therefore, the most reasonable and concise next word set for this context is "antibiotic," as it is the most direct and commonly used term in medical practice for initiating broad-spectrum treatment in cases of suspected sepsis.
BOW-Reg	Given the context "In the ED with fever, hypotension, and suspected sepsis; after blood cultures, initiate broad-spectrum", the next word set can be reasoned as follows: 1. **Medical Context**: The context is clearly in a medical setting, specifically in an Emergency Department (ED) dealing with a patient who has fever, hypotension, and suspected sepsis. This sets the tone for the next words to be related to medical treatment or procedures. 2. **Action Following Blood Cultures**: The phrase "after blood cultures" indicates that the next words should describe an action or treatment that follows the completion of blood cultures. This could include starting a specific type of treatment, ordering further tests, or documenting the patient's condition. 3. **Broad-Spectrum Treatment**: The phrase "initiate broad-spectrum" suggests that the next words will likely describe a type of medication or treatment that is broad-spectrum in nature, meaning it is effective against a wide range of pathogens. Based on these points, the next word set will likely contain: - **Medical treatments or medications**: Such as antibiotics, antivirals, or other broad-spectrum drugs. - **Procedures or actions**: Such as starting an IV, ordering additional tests, or documenting the patient's condition. - **Medical terms**: Such as the names of specific conditions, treatments, or medications. Therefore, a concise and unbiased description of the next word set could be: "The next word set will likely contain medical treatments or medications, procedures or actions following the completion of blood cultures, and medical terms related to the patient's condition or the treatment being initiated."

Table 8: A representative medical example using Qwen2.5-7B-Instruct as the backbone model.

Context	Under IFRS 15, revenue is recognized when control
Vanilla	Given the context "Under IFRS 15, revenue is recognized when control", the next word set will likely contain nouns or adjectives that complete the phrase "when control [next-word]". The most probable next words would be related to the transfer of risk and reward, which are key elements in determining when revenue is recognized under IFRS 15. Therefore, the next word set can be described as containing terms such as "transfers", "is transferred", "has been transferred", "has transferred", or "will transfer". These terms capture the idea of the transfer of control, which is central to the revenue recognition criteria under IFRS 15. To summarize, the next word set is best described as: "terms indicating the transfer of control, such as 'transfers', 'is transferred', 'has been transferred', 'has transferred', or 'will transfer'."
BR	Reasoning: Given the context "Under IFRS 15, revenue is recognized when control", the most likely next word is "transfers". This is because IFRS 15, the International Financial Reporting Standard for revenue from contracts with customers, specifies that revenue should be recognized when control of the promised goods or services is transferred to the customer. The word "transfers" directly aligns with the concept of control transfer in the context of revenue recognition under IFRS 15. \boxed{transfers}
BOW-Core	Given the context "Under IFRS 15, revenue is recognized when control [next-word]", the next word in the sentence is likely to be a noun that completes the phrase "when control [next-word]". The most logical and common completion for this phrase in the context of accounting and financial reporting standards would be "transfers" to "the customer." It fits grammatically and semantically, as it is used in the context of when revenue is recognized according to IFRS 15. The candidate set for the next word is small and concrete enough to be listed out, and the most appropriate and commonly used word in this context is "transfers." Therefore, the next word in the sentence is most likely "transfers." Output: transfers
BOW-Reg	Given the context "Under IFRS 15, revenue is recognized when control", the next word set will likely contain nouns, verbs, or phrases that logically follow the discussion of revenue recognition under IFRS 15. The phrase "Under IFRS 15, revenue is recognized when control" suggests that the following words will provide more details about the criteria for recognizing revenue or possibly introduce a new aspect of the standard. To reason further, IFRS 15 (International Financial Reporting Standards 15) is specifically about revenue recognition. Therefore, the next word set will likely include: 1. **Nouns or phrases related to revenue recognition criteria**: Such as "transferred" (indicating the transfer of control), "to a customer" (specifying the recipient of the revenue), or "to the customer" (further clarifying the point of transfer). 2. **Verbs or phrases that continue the explanation**: Such as "is transferred" (completing the sentence about revenue recognition), "to the customer" (further detailing the process), or "is determined" (introducing a new aspect of the standard). 3. **Phrases that introduce a new aspect of the standard**: Such as "of the good or service" (further specifying what is being recognized as revenue), or "is transferred" (continuing the explanation of the criteria). Thus, a concise and unbiased description of the next word set could be: "The next word set will likely contain nouns, verbs, or phrases related to the criteria for recognizing revenue under IFRS 15, specifically focusing on the transfer of control, the recipient of the revenue, and the process of revenue recognition."

Table 9: A representative finance example using Qwen2.5-7B-Instruct as the backbone model.