FILD: RELATIONAL STATE FIELDS FOR EVIDENCE-GATED REVISION IN LOOKAHEAD DECODING

Anonymous authors
Paper under double-blind review

ABSTRACT

While greedy decoding provides a strong default, bounded lookahead can reveal better continuations beyond the locally preferred token. The challenge is to use these partial futures to decide when to revise the model's native trajectory. We introduce FILD, a training-free decoder that builds evidence for revision from the model's own lookahead computation. Using the same frozen model, FILD uses self-verification to assess continuation quality and combines these assessments with native token likelihoods to identify a challenger. This proposal reflects a fixed-horizon assessment; the unfolding candidate futures reveal how support for it develops relative to the native continuation. FILD organizes their hidden states into a relational state field, assessing whether the challenger is closer than the native continuation to the same reference branches across model depths and continuation horizons. For unfinished candidate futures, revision additionally requires support that agrees across depths and persists across adjacent horizons. If all branches terminate within the lookahead horizon, the proposal suffices for revision. This procedure reuses states already computed during exploration, requiring no auxiliary trained probe or router. Across four language models and six generation benchmarks, FILD achieves the strongest aggregate performance among the evaluated decoders and the highest six-task average for each model. Component analysis shows that self-verified proposals improve factual question answering, with relationally guided revision providing further gains on reasoning tasks.

1 INTRODUCTION

A language model's most likely next token need not lead to its best answer. Lookahead decoding addresses this mismatch by exploring candidate continuations before deciding how to proceed (Lu et al., 2022; Huang et al., 2025). These continuations can be scored using likelihood estimates or model self-evaluation to guide generation (Kadavath et al., 2022; Xie et al., 2023). Intrinsic self-correction can also degrade reasoning accuracy (Huang et al., 2024), making the decision to depart from the native path consequential. Scores summarize candidate quality at a given horizon, leaving open how support for that alternative develops as generation continues.

The same candidate exploration provides a way to follow this developing support. Branches start from a shared prefix and represent competing ways to continue it; as they grow, their hidden states evolve across layers and generation steps. Prior work on layer-based decoding and representation-guided inference shows that such states carry useful signals for generation (Chuang et al., 2024; Zhang et al., 2024; Li et al., 2023a; Yin et al., 2026). Within lookahead, the candidate ensemble gives these signals a common frame of reference: each continuation can be assessed in relation to the other futures explored from the same prefix.

These evolving relationships offer a basis for deciding when to revise the decoding trajectory. We organize the candidate states into a **relational state field** spanning branches, model depth, and continuation time. Comparing the model's preferred continuation and a proposed challenger to the same reference branches reveals which is more closely aligned with the surrounding candidate futures. Following this comparison across layers and generation steps reveals whether the preference is shared across different depths of internal processing and whether it persists across adjacent continuation horizons. The exploration that suggests a promising departure thus also reveals how support for that departure develops.

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

We introduce **FILD (Field-Informed Lookahead Decoding)**, a training-free decoder that uses this evolving evidence to guide selective revision. FILD asks the same model to assess whether candidate continuations are likely to lead to a correct answer, then combines these self-verification scores with native token likelihoods. A challenger must offer a continuation advantage that offsets its likelihood deficit relative to the native action. For unfinished candidate futures, the relational state field then tests whether support for this challenger agrees across depths and persists across adjacent horizons; if all branches terminate within the lookahead horizon, FILD retains the lookahead-selected challenger. The field reuses hidden states computed during exploration, allowing the entire procedure to operate with a single frozen model and no auxiliary trained probe or router.

We evaluate FILD across four language models and six benchmarks spanning factual question answering, resistance to memorized completions, and reasoning. FILD achieves the highest aggregate performance among the evaluated decoders and the highest six-task average for every model. Resource measurements show that these gains come with modest runtime and peak-memory overhead.

Our contributions are threefold:

- **A relational view of lookahead:** we formulate candidate support as a paired comparison against shared reference branches, tracked across model depth and continuation time.
- **A training-free decoder:** we develop FILD, a single-model procedure for proposing and selectively committing revisions within a bounded lookahead budget.
- **Findings on selective revision:** component analyses show gains from self-verified proposals on factual question answering and additional gains from relationally guided revision on reasoning tasks. Trajectory analyses reveal distinct support patterns across models.

2 Related Work

Decoding through distributional contrast. Contrasting predictive distributions provides a way to steer generation. DExperts combines expert and anti-expert models for attribute control (Liu et al., 2021), while Contrastive Decoding contrasts an expert with a smaller amateur under a plausibility constraint (Li et al., 2023b). Context-aware decoding contrasts predictions with and without context to improve faithfulness (Shi et al., 2024). Within one model, DoLa contrasts mature and premature layers (Chuang et al., 2024), and SLED uses their logit evolution to refine the final distribution (Zhang et al., 2024). Another line constructs negative signals: ICD induces a factually weaker model (Zhang et al., 2025), HICD disperses selected heads' attention (Jiang et al., 2025), and NDAD masks influential heads and weights their hallucination-oriented predictions (Qiu et al., 2026). FILD instead expands candidate futures and separates their likelihood-aware evaluation from the relational decision to adopt a proposed revision.

Lookahead and self-evaluated search. Additional inference computation can improve generation through sampling, evaluation, and revision. Self-consistency aggregates answers across sampled reasoning paths (Wang et al., 2023); trained outcome and process verifiers score candidate solutions or intermediate steps (Cobbe et al., 2021; Lightman et al., 2024). Same-model correctness assessments (Kadavath et al., 2022) also guide beam search (Xie et al., 2023). NeuroLogic A*esque uses lookahead heuristics for constraint satisfaction (Lu et al., 2022), while DeAL searches for alignment objectives (Huang et al., 2025). Tree of Thoughts explores and evaluates intermediate thoughts (Yao et al., 2023), and Self-Refine iterates feedback and revision (Madaan et al., 2023). The effectiveness of allocating computation to search or revision depends on problem difficulty (Snell et al., 2024). FILD uses bounded local token branches: continuation scores nominate a challenger, and evolving branch relationships guide its admission while candidate futures remain unfinished.

Representations for inference-time decisions. Internal activations can reveal information beyond the final output. Contrast-Consistent Search extracts knowledge through logical consistency constraints (Burns et al., 2023), and the tuned lens decodes intermediate states into vocabulary predictions (Belrose et al., 2023). Contrastive search uses representation similarity to discourage degeneration (Su et al., 2022); activation decoding uses contextual activation sharpness to adjust token probabilities (Chen et al., 2024). ITI steers activations along learned truth-related directions (Li et al., 2023a). Internal-consistency calibration weights reasoning paths by agreement among latent predictions (Xie et al., 2024), while Local Branch Routing trains a router on candidate-future states

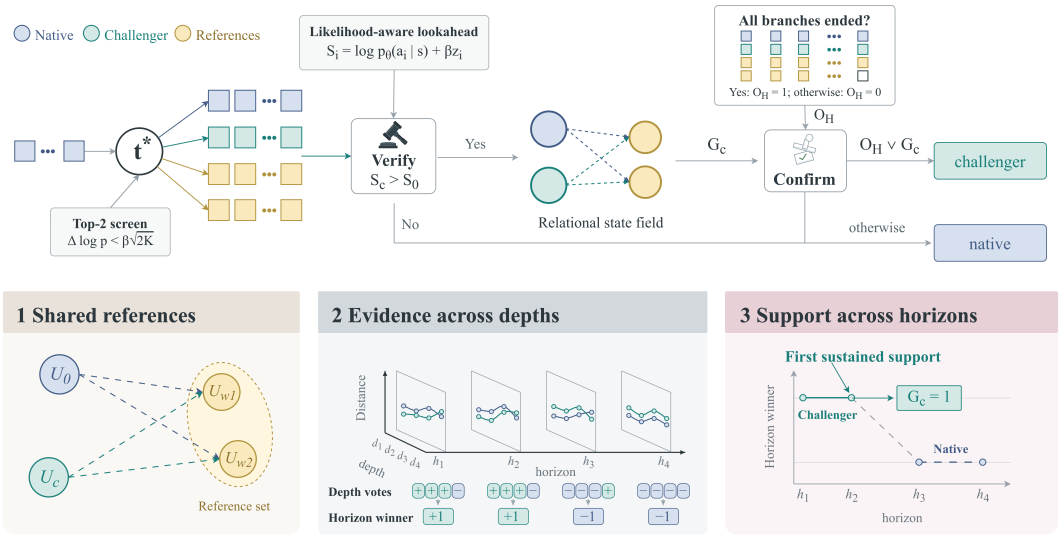

Figure 1: Overview of FILD. After the proposal gate, revision follows terminal resolution or positive field support; otherwise, native is retained. The lower panels show shared-reference comparisons, depth agreement, and the first direction sustained across adjacent horizons, which determines the field signal even if later support reverses.

(Yin et al., 2026). FILD compares native and challenger against shared reference branches. Agreement across depths and persistence across horizons provide a revision signal without training a probe or router.

3 METHOD

Given input x, a frozen language model generates a token sequence $y = (y_1, \ldots, y_T)$ over vocabulary $\mathcal{V}$ according to $p_\theta(y \mid x) = \prod_{t=1}^{T} p_\theta(y_t \mid s_t)$, where $s_t = (x, y_{<t})$ is the current prefix. A decoding policy selects tokens from these conditional distributions. We use greedy decoding as the native policy, $a_0(s_t) = \arg\max_{a \in \mathcal{V}} p_\theta(a \mid s_t)$, with deterministic tie-breaking.

FILD uses the same lookahead computation to propose a challenger and decide whether to adopt it (Figure 1). Each local probe combines same-model self-verification with native likelihood to form a proposal, then follows the candidate branches for terminal resolution or positive relational corroboration while they remain open. Probe positions are selected online from the observed prefix, with at most B probes per response.

3.1 LIKELIHOOD-AWARE LOOKAHEAD

Proposal scoring. For a prospective probe at t, let $\mathcal{A}_t = \{a_0, a_1, \ldots, a_{K-1}\}$ be the top-K actions ordered by native probability, with $K \geq 3$. Each continuation $\widetilde{y}_i^{h_p}$ starts at a_i and extends to horizon h_p or termination, counting tokens from the root action. A fixed query $\mathcal{Q}_i = \mathcal{Q}(x, y_{<t} \circ \widetilde{y}_i^{h_p})$ asks the same model whether naturally continuing the response is likely to yield a correct answer, requesting Yes or No. We standardize the resulting logit differences across candidates:

$$q_i = \lambda_\theta(v^+ \mid \mathcal{Q}_i) - \lambda_\theta(v^- \mid \mathcal{Q}_i), \qquad z_i = \frac{q_i - \mu_q}{\sigma_q}. \tag{1}$$

Here λ_θ denotes output logits, v^+, v^- the Yes/No verbalizers, and μ_q, σ_q the mean and population standard deviation of the K scores; $z_i = 0$ when σ_q is numerically zero. Combining this evidence with native likelihood gives

$$S_i = \log p_\theta(a_i \mid s_t) + \beta z_i, \qquad c = \arg\max_i S_i, \tag{2}$$

where $\beta \geq 0$ weights continuation evidence and native wins ties. For any non-native candidate i,

$$S_i - S_0 = \beta(z_i - z_0) - \underbrace{\left[\log p_\theta(a_0 \mid s_t) - \log p_\theta(a_i \mid s_t)\right]}_{\text{native likelihood deficit}}. \tag{3}$$

The continuation advantage must therefore offset the native likelihood deficit.

Pivot selection. Standardization bounds $z_i - z_0$ by $\sqrt{2K}$, while a_1 has the smallest non-native likelihood deficit. Thus no candidate can beat native when

$$\log p_\theta(a_0 \mid s_t) - \log p_\theta(a_1 \mid s_t) \geq \beta\sqrt{2K}. \tag{4}$$

This lossless screen skips positions without opening branches. FILD generates the provisional native trajectory in a forward-only buffer. At predefined checkpoints, the scheduler tracks the earliest entropy maximum in the observed search prefix. The checkpoint rule selects it as $t^\star$ when it passes the screen and remains unchanged for the required number of consecutive checkpoints.

Challenger formation. At $t^\star$, FILD expands the K branches to h_p and scores them as a batch. Each probe consumes one budget unit regardless of outcome, and later probes use the committed prefix. The proposal gate admits c as the **challenger** when it is non-native, $S_c - S_0 > 0$, and the conditions $H_\theta(s_{t^\star}) \geq \eta$, $z_c - z_0 > \delta_z$, and $q_c - q_0 > \delta_q$ are satisfied, where H_θ is next-token entropy. The thresholds require uncertainty and both standardized and raw self-verification gains. Otherwise, FILD retains native.

The same rollouts reveal the challenger's relationships to other candidate futures, which FILD uses to assess support for revision while branches remain open.

3.2 Relational state fields

To follow support for a formed challenger, FILD compares it and the native continuation against the same remaining candidate branches. Their relative hidden-state distances define a paired preference at each depth and horizon. FILD continues the same branches over predefined horizons $\mathcal{H} = \{h_1 < \cdots < h_J = H\}$, where $h_p \in \mathcal{H}$. States up to h_p are retained from the proposal rollout, and unfinished branches are then advanced incrementally toward H. The same challenger is evaluated over the full horizon sequence, including retained states.

Let $B_i^{(h)}$ denote the tokens produced after advancing branch i to horizon h, stopping early at a terminal event, and let $\rho_i(h)$ indicate whether that branch remains unfinished. For model depth ℓ, define

$$U_{i,\ell,h} = \phi_\ell\left(x, y_{<t^\star}, B_i^{(h)}\right), \tag{5}$$

where ϕ_ℓ returns the layer-ℓ hidden vector at the last real branch token. If a branch terminates before h, its final real-token state is carried forward. The collection over branch, depth, and horizon forms the **state field**.

For challenger c, the remaining candidates form the shared-reference set $W_c = \{0, \ldots, K-1\} \setminus \{0, c\}$, with $|W_c| = K - 2 \geq 1$.

These shared references give both continuations the same comparison context. Using cosine distance d, we average the paired distance differences to obtain

$$M_{\ell,h}(c) = \frac{1}{|W_c|} \sum_{w \in W_c} \left[d(U_{0,\ell,h}, U_{w,\ell,h}) - d(U_{c,\ell,h}, U_{w,\ell,h})\right]. \tag{6}$$

A positive margin places the challenger closer than native, on average, to the same references; a negative margin favors native.

The paired construction makes the comparison local to the alternatives available at the pivot. A reference branch need not be a correct continuation: it supplies the same comparison point for both candidates. Averaging the distance differences summarizes their relative alignment with these shared alternatives, rather than assigning either candidate an absolute quality score. The likelihood-aware proposal and the relational comparison therefore contribute different evidence to the revision decision.

FILD next asks whether this preference is shared across model depths. For selected depths $\mathcal{L}$, a direction is assigned to horizon h when enough depth-wise margins agree:

$$g_h(c) = \begin{cases} +1, & |\{\ell \in \mathcal{L} : M_{\ell,h}(c) > 0\}| \geq r, \\ -1, & |\{\ell \in \mathcal{L} : M_{\ell,h}(c) < 0\}| \geq r, \\ 0, & \text{otherwise}, \end{cases} \tag{7}$$

where $r > |\mathcal{L}|/2$ is the required depth agreement. FILD then follows these horizon-level directions as the branches unfold. The first direction that persists across two adjacent predefined horizons determines the field signal:

$$\tau_{\pm}(c) = \min\{j \geq 2 : g_{h_{j-1}}(c) = g_{h_j}(c) = \pm 1\}, \qquad G_c = \mathbf{1}[\tau_+(c) < \tau_-(c)], \tag{8}$$

with $\min \varnothing = +\infty$. Thus $G_c = 1$ when the first sustained direction favors the challenger, even if later horizons change direction. If neither direction persists, $G_c = 0$. The final revision rule below combines this signal with branch termination.

Depth agreement and temporal persistence act at different levels. The former requires a preference to recur across representations of the same candidate futures; the latter requires that preference to survive an additional observation horizon. A positive margin at one layer or one horizon is thus insufficient for open-field admission. The first sustained direction resolves the temporal comparison without requiring all later horizons to agree.

3.3 Binary revision rule

Terminal resolution at the maximum horizon is denoted by $O_H = \mathbf{1}[\forall i : \rho_i(H) = 0]$; thus $O_H = 1$ means that all candidate branches have ended by H.

Terminal resolution. If $O_H = 1$, later field columns under last-real-state carry-forward would only duplicate the terminal states and provide no new temporal evidence. FILD therefore retains the challenger already selected by likelihood-aware lookahead.

Open-field corroboration. If $O_H = 0$, at least one candidate future remains unresolved, and revision additionally requires $G_c = 1$.

For a challenger that passes the proposal gate, the final action is

$$\widehat{a}(s_{t^\star}) = \begin{cases} a_c, & O_H \vee G_c, \\ a_0, & \text{otherwise}. \end{cases} \tag{9}$$

Illustrative decision trace. Figure 2 traces two open-field decisions ($O_H = 0$) after a challenger has passed the proposal gate. In the left panel, replacing addition with multiplication yields the correct answer, 48. Challenger support persists across the first two horizons, admitting the revision under Eq. (9). In the right panel, the proposed continuation misreads the working days. Its early support is interrupted, and the first sustained direction favors native, preserving the correct answer, 120.

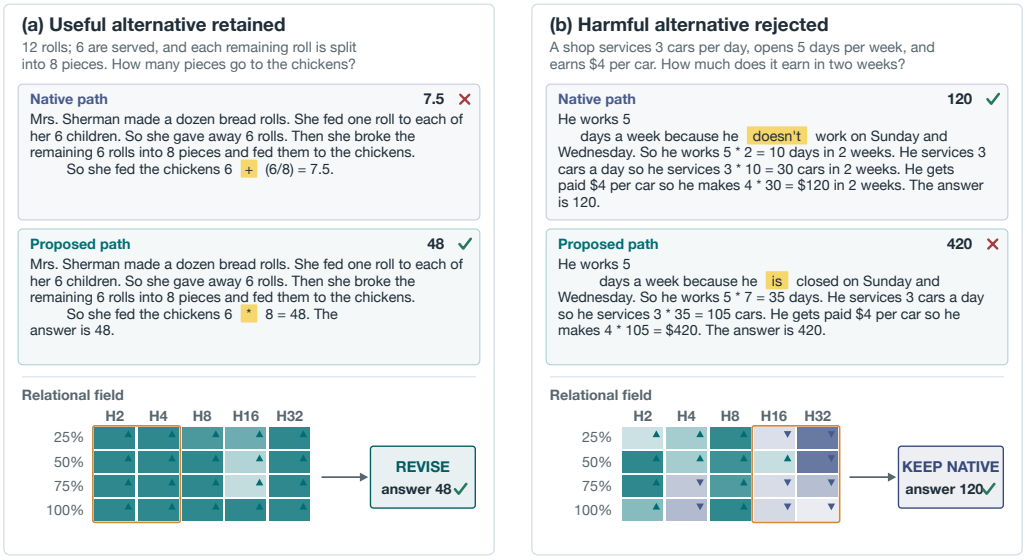

Figure 2: Illustrative decision traces for revision and native retention. Each panel links the pivot token and candidate continuations to the depth–horizon field and final action. Yellow marks the proposed token change; teal/upward and blue/downward cells favor challenger and native. Rows denote depths, columns denote horizons, and orange outlines mark the first direction sustained across adjacent horizons.

3.4 Efficient realization

FILD initializes the candidate batch from cloned copies of the native-prefix KV cache. The selected hidden states are recorded during the same branch forwards, so constructing the state field requires no additional model pass beyond candidate generation. The continuation queries are also evaluated in a single batch.

At the decision horizon, FILD commits the tokens and KV-cache state already materialized for the selected branch. If that branch remains unfinished, greedy decoding resumes directly from its cache; accepted lookahead tokens are neither discarded nor regenerated. The Top-2 screen avoids unnecessary probes, and terminal resolution stops branch expansion once all candidates have terminated. With fixed K and H, each probe expands at most KH branch-token positions and evaluates the K continuation queries as one batch. The budget therefore bounds the number of such local expansions, while caching preserves the computation on the committed path.

4 Experiments

4.1 Experimental setup

Benchmarks. The six benchmarks cover factual QA, context-sensitive completion, and reasoning. NQ-Open (Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017), and PopQA (Mallen et al., 2023) cover open-domain QA. MemoTrap (Liu & Liu, 2023) tests context-sensitive completion, GSM8K (Cobbe et al., 2021) multi-step numerical reasoning, and StrQA (Geva et al., 2021) knowledge-intensive Boolean reasoning.

Models and baselines. We evaluate Qwen3-8B, Qwen3-14B, Mistral-7B, and Falcon3-7B. Baselines include Greedy and three methods that refine next-token distributions using signals from within the model. DoLa (Chuang et al., 2024) contrasts final and dynamically selected premature layers, while SLED (Zhang et al., 2024) uses logit evolution across layers to refine the final distribution. NDAD (Qiu et al., 2026) uses attention-head masking to expose hallucination-leaning predictions as a negative decoding signal.

Table 1: Generation accuracy (%; higher is better) across four models and six benchmarks. Each model block starts with Greedy, followed by alternative decoders applied to that model. Bold and underlined values mark the best and second-best results within each model–task pair.

Method	NQ-Open	TriviaQA	PopQA	MemoTrap	GSM8K	StrQA
Qwen3-8B	22.47	43.35	11.24	59.57	91.74	74.28
+DoLa	21.05	41.83	11.25	59.33	91.21	72.40
+SLED	21.91	43.19	11.19	60.17	91.58	73.58
+NDAD	21.97	43.21	11.20	60.29	91.81	73.49
+FILD	**23.21**	**45.76**	**11.98**	**60.89**	**91.89**	**74.32**
Qwen3-14B	23.35	53.49	12.43	60.29	93.56	76.68
+DoLa	21.86	50.42	10.63	**62.92**	93.71	**77.77**
+SLED	23.43	53.43	12.17	60.05	**94.09**	76.59
+NDAD	23.41	53.32	12.13	60.29	93.86	76.59
+FILD	**24.24**	**55.57**	**13.10**	60.53	93.48	77.16
Mistral-7B	23.21	55.73	**23.14**	69.62	33.21	60.57
+DoLa	20.17	50.42	18.79	69.02	27.98	45.15
+SLED	23.82	55.71	23.11	69.98	32.90	60.92
+NDAD	23.60	55.79	22.97	69.38	34.27	60.52
+FILD	**24.88**	**57.59**	22.76	**70.22**	**34.34**	**61.48**
Falcon3-7B	20.69	49.34	16.39	40.31	78.92	76.72
+DoLa	17.34	44.99	14.10	39.47	72.86	75.15
+SLED	20.58	49.53	16.54	40.79	79.53	76.86
+NDAD	20.50	49.39	16.49	40.31	77.71	76.81
+FILD	**21.75**	**51.80**	**17.99**	**41.51**	**80.59**	**76.90**

Evaluation protocol. All methods use the same model, prompt, generation limit, and evaluator for each model–task pair, isolating the effect of decoding. Open-domain QA and MemoTrap use relaxed normalized answer match, GSM8K uses numeric-answer accuracy, and StrQA uses format-normalized Boolean accuracy. Complete settings appear in the appendix.

4.2 Overall performance

Across benchmarks. FILD achieves the highest aggregate accuracy, 49.75%, exceeding the strongest competing decoder by 0.93 percentage points (Table 1). The most consistent gains appear in open-domain QA: FILD ranks first for every model on NQ-Open and TriviaQA, raising their cross-model averages over Greedy from 22.43% to 23.52% and from 50.48% to 52.68%, respectively. Its cross-model average also leads on PopQA, extending this pattern across all three factual QA benchmarks.

The gains also extend to contextual completion and reasoning. FILD improves over Greedy for every model on MemoTrap, where a familiar completion can conflict with the local context. On GSM8K and StrQA, it leads six of the eight model–task settings and the cross-model average on both benchmarks. These results show that selective revision helps both factual answering and generation that must follow context or develop a reasoning chain.

Across models. FILD leads the six-task average for every model by 0.71–1.12 percentage points over its strongest competitor. That competitor varies across models: Greedy for both Qwen models, NDAD for Mistral, and SLED for Falcon. The gains also have different task profiles. FILD ranks first on all six tasks for Qwen3-8B and Falcon3-7B, while its advantage on Qwen3-14B is concentrated in open-domain QA. The model-level improvement persists across both changing baseline rankings and different task profiles.

4.3 Component analysis

Table 2 separates proposal and admission. Proposal-only shares FILD's pivot, candidates, continuation query, score, and proposal gate, but commits every eligible challenger. We compare it with Greedy on factual QA and with FILD on reasoning tasks to examine these two stages.

Table 2: Accuracy gains from proposal and relational revision (percentage points). Prop. denotes Proposal-only. Left: Proposal-only minus matched Greedy on NQ-Open and TriviaQA. Right: FILD minus matched Proposal-only on GSM8K and StrQA.

Model	Prop. − Greedy		FILD − Prop.	
	NQ-Open	TriviaQA	GSM8K	StrQA
Qwen3-8B	+0.72	+2.41	+0.08	+0.61
Qwen3-14B	+0.89	+2.08	+0.23	+0.48
Mistral-7B	+1.66	+1.86	+0.08	+0.52
Falcon3-7B	+1.05	+2.47	+0.61	+0.17

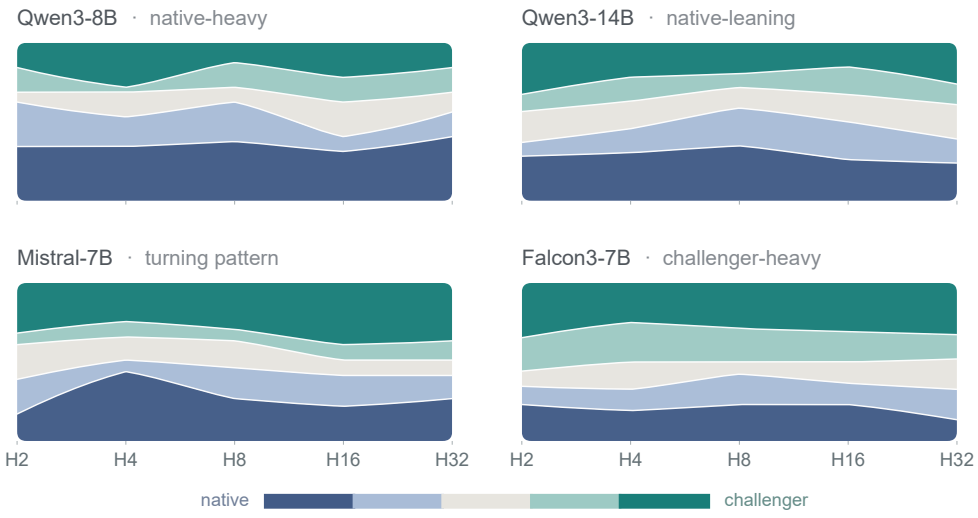

Figure 3: Relational support across probes remaining open at H. At each horizon, band thickness gives the share of probes at each of five depth-vote balances across four sampled layers: unanimous native support (dark blue), native majority, balanced votes, challenger majority, and unanimous challenger support (dark teal). Smooth boundaries connect the observed horizons $2, 4, 8, 16, 32$.

Proposal-only improves all four models by 0.72–1.66 percentage points on NQ-Open and 1.86–2.47 points on TriviaQA. These gains show that self-verification combined with native likelihood can identify useful departures from greedy decoding. On GSM8K and StrQA, FILD further improves on Proposal-only by 0.08–0.61 and 0.17–0.61 points, respectively. With shared proposals, these additional gains arise from selecting which eligible challengers to commit.

Figure 2 illustrates how these roles interact. In one case, the field admits a correct proposal; in the other, it rejects an incorrect proposal and preserves a correct native answer. Selective revision thus enables corrections while avoiding harmful departures. Appendix D also examines failures of this selection: useful proposals left uncommitted and incorrect proposals admitted by the field.

4.4 Model-specific state-field patterns

The state-field profiles reveal differences in both the prevailing direction of support and its evolution as candidate continuations unfold (Figure 3). Qwen3-8B exhibits predominantly native-oriented support, whereas Qwen3-14B shows a more mixed profile with an overall native leaning. Mistral-7B shifts toward greater challenger support later in the continuation. Falcon3-7B maintains a challenger-heavy profile throughout.

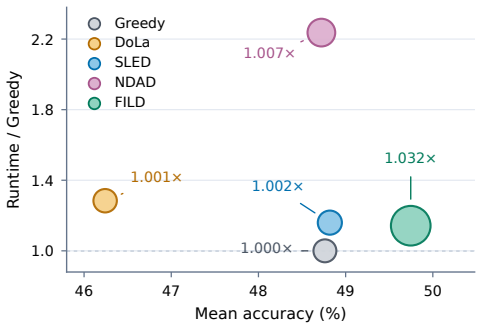

Figure 4: Accuracy versus runtime. Runtime and memory are relative to Greedy; bubbles and labels encode peak memory.

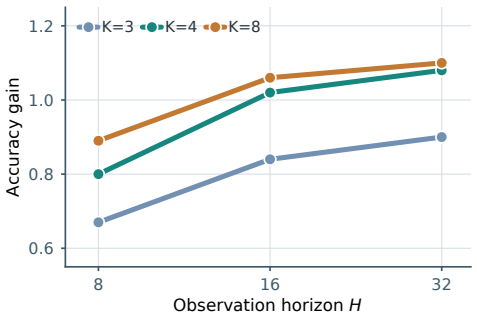

Figure 5: K–H sensitivity. Accuracy gain over Greedy (percentage points) across horizons H and candidate widths $K = 3, 4, 8$.

These profiles distinguish proposal quality from relational support. In a native-heavy profile, a challenger that passes the proposal gate still frequently faces depth-wise comparisons favoring native. A challenger-heavy profile reflects more frequent alignment between the proposed alternative and the shared references. Mistral's changing profile shows that these relationships can evolve as branches develop, revealing variation that a proposal score at a single horizon leaves unresolved.

This variation gives depth agreement and temporal persistence complementary roles. Agreement across depths identifies a preference shared by different levels of representation, while persistence distinguishes a briefly expressed preference from one sustained across successive observations. The population profiles summarize how these preferences are distributed; individual decisions depend on each probe's sequence. A later population shift toward challenger support can therefore coexist with native retention where native support persisted first.

4.5 Quality–resource trade-off

On the resource-evaluation subset (Appendix A.3), FILD uses $1.142\times$ the runtime and $1.032\times$ the peak memory of Greedy, with the lowest runtime overhead among the evaluated decoding interventions (Figure 4).

Candidate width K and observation horizon H control the breadth and length of lookahead, and hence the computation allocated to each probe. Figure 5 examines the accuracy gains from these choices. All three widths improve with longer horizons, and wider candidate sets yield larger gains at each horizon. The gap between $K = 4$ and $K = 8$ narrows as the horizon grows: exploring more candidates brings a smaller additional accuracy gain when their continuations are observed for longer. This pattern makes the joint choice of K and H central to balancing lookahead quality and cost.

5 Conclusion

We presented FILD, a training-free decoder that uses the same lookahead computation to propose alternatives and assess the evolving support for revision. Self-verification combined with native token likelihoods identifies a challenger. For unfinished candidate futures, a relational state field guides revision through support that agrees across model depths and persists across continuation horizons. Across four models and six benchmarks, FILD achieves the strongest aggregate performance among the evaluated decoders with modest runtime and memory overhead on the measured subset. Component analysis shows gains from self-verified proposals on factual question answering and further gains from relationally guided revision on reasoning tasks. Reusing candidate states and cached computation makes this decision process practical within a bounded lookahead budget. Together, these findings show how candidate exploration can both identify promising continuations and reveal, through their evolving relationships, when to depart from the native trajectory.

AI USE STATEMENT

Generative AI tools assisted literature discovery, manuscript drafting and revision, articulation of the problem formulation, and preparation of the LaTeX project. The authors are responsible for the final content of the paper.

REFERENCES

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. arXiv preprint arXiv:2303.08112, 2023.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *The Eleventh International Conference on Learning Representations*, 2023.

Shiqi Chen, Miao Xiong, Junteng Liu, Zhengxuan Wu, Teng Xiao, Siyang Gao, and Junxian He. In-context sharpness as alerts: An inner representation perspective for hallucination mitigation. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 7553–7567, 2024.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. DoLa: Decoding by contrasting layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168, 2021.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. arXiv preprint arXiv:2101.02235, 2021.

James Y. Huang, Sailik Sengupta, Daniele Bonadiman, Yi-An Lai, Arshit Gupta, Nikolaos Pappas, Saab Mansour, Katrin Kirchhoff, and Dan Roth. DeAL: Decoding-time alignment for large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 26280–26300, 2025.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations*, 2024.

Xinyan Jiang, Hang Ye, Yongxin Zhu, Xiaoying Zheng, Zikang Chen, and Jun Gong. HICD: Hallucination-inducing via attention dispersion for contrastive decoding to mitigate hallucinations in large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 7764–7786, 2025.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know. arXiv preprint arXiv:2207.05221, 2022.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems*, 2023a.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 2023b.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2024.

Alisa Liu and Jiacheng Liu. The MemoTrap dataset. Inverse Scaling Prize dataset, 2023.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. DExperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6691–6706, 2021.

Ximing Lu, Sean Welleck, Peter West, Liwei Jiang, Jungo Kasai, Daniel Khashabi, Ronan Le Bras, Lianhui Qin, Youngjae Yu, Rowan Zellers, Noah A. Smith, and Yejin Choi. NeuroLogic A*esque decoding: Constrained text generation with lookahead heuristics. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 780–799, 2022.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, 2023.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 2023.

Panjia Qiu, Mingyuan Fan, Cen Chen, and Daixin Wang. NDAD: Negative-direction aware decoding for large language models via controllable hallucination signal injection. In *The Fourteenth International Conference on Learning Representations*, 2026.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. Trusting your evidence: Hallucinate less with context-aware decoding. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 783–791, 2024.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. arXiv preprint arXiv:2408.03314, 2024.

Yixuan Su, Tian Lan, Yan Wang, Dani Yogatama, Lingpeng Kong, and Nigel Collier. A contrastive framework for neural text generation. In *Advances in Neural Information Processing Systems*, 2022.

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

Yuxi Xie, Kenji Kawaguchi, Yiran Zhao, Xu Zhao, Min-Yen Kan, Junxian He, and Qizhe Xie. Self-evaluation guided beam search for reasoning. In *Advances in Neural Information Processing Systems*, 2023.

Zhihui Xie, Jizhou Guo, Tong Yu, and Shuai Li. Calibrating reasoning in language models with internal consistency. In *Advances in Neural Information Processing Systems*, 2024.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, 2023.

Yutong Yin, Mingyu Jin, Jin Pan, Changyi Yang, Zijie Xia, Dhruv Pai, Shuming Hu, Zhen Zhang, Chenyang Zhao, Jinman Zhao, Wujiang Xu, Raymond Li, Xin Eric Wang, Julian McAuley, and Zhaoran Wang. Efficient and trainable language model test-time scaling via local branch routing. arXiv preprint arXiv:2606.25354, 2026.

Jianyi Zhang, Da-Cheng Juan, Cyrus Rashtchian, Chun-Sung Ferng, Heinrich Jiang, and Yiran Chen. SLED: Self logits evolution decoding for improving factuality in large language models. In *Advances in Neural Information Processing Systems*, 2024.

Yue Zhang, Leyang Cui, Wei Bi, and Shuming Shi. Alleviating hallucinations of large language models through induced hallucinations. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 8233–8247, 2025.

APPENDIX

A EXPERIMENTAL SETTINGS

We evaluate FILD on factual question answering, contextual completion, and reasoning. This section summarizes the comparison setup; Section B evaluates linguistic quality, Section C presents the decoding procedure, and Section D illustrates its decisions through individual cases.

A.1 MODELS, TASKS, AND EVALUATION

We use Qwen3-8B, Qwen3-14B, Mistral-7B, and Falcon3-7B. Within each model–task pair, all decoding methods share the model, prompt, generation limit, stopping rules, and evaluator. Table 3 summarizes the six benchmarks.

Table 3: Dataset-level evaluation protocol. "Chat/plain" uses the model's chat template for Qwen and the plain task prompt for Mistral and Falcon.

Dataset	Examples	Shots	Rendering	Max. tokens	Evaluator
NQ-Open	3610	6	chat/plain	32	short answer
TriviaQA	3610	6	chat/plain	32	short answer
PopQA	14267	6	chat/plain	32	short answer
MemoTrap	836	0	raw completion	16	category-aware
GSM8K	1319	4	chat/plain	512	numeric
StrategyQA	2290	6	chat/plain	256	Boolean

Qwen models use their chat template with thinking disabled; Mistral and Falcon receive the plain task prompt. MemoTrap uses raw completion. The factual-QA prompts request a literal answer and allow abstention. We evaluate these answers using normalized equality or containment. MemoTrap uses category-aware matching, with explicit A/B selection for history-of-science questions. GSM8K uses numeric-answer accuracy, and StrategyQA uses Boolean-answer accuracy.

A.2 DECODING SETTINGS

FILD uses one revision opportunity per response. Each probe considers four candidate actions ($K = 4$), forms its proposal after four tokens ($h_p = 4$), and observes the continuations over horizons $\{2, 4, 8, 16, 32\}$ up to a task-dependent maximum H. The field samples four evenly spaced model depths, with three-depth agreement required for a directional vote.

The proposal weight is $\beta = 0.5$, with standardized and raw margin thresholds $\delta_z = 0.25$ and $\delta_q = 0.125$; Qwen3-8B on MemoTrap uses $\beta = 1$. The entropy threshold is $\eta = 0$ for Qwen and $\eta = 1$ for Mistral and Falcon. Maximum horizons range from 4 to 32 tokens. The scheduler uses a shorter window for short-form tasks and a longer window for reasoning, with persistence and window $(m, w) = (2, 32)$ and $(3, 64)$, respectively.

Proposal-only shares FILD's pivot, candidates, continuation query, and proposal gate, but commits an eligible challenger without relational admission.

The model–task settings were developed using the complete benchmark populations, so the comparisons characterize performance on these populations. Reference answers and evaluation scores are used only after generation.

A.3 RESOURCE MEASUREMENT

Resource ratios cover Qwen3-8B, Mistral, and Falcon on NQ-Open and GSM8K, using a single NVIDIA H20 with BF16 inference. Each model–task cell uses 16 requests, with the first excluded from timing as warm-up. Wall time and peak GPU memory, including model weights and runtime allocations, are normalized to Greedy within each cell and then averaged.

B LINGUISTIC QUALITY EVALUATION

We assess whether selective revision preserves the naturalness and readability of the generated responses. Gemini 2.5 Pro scores matched Qwen3-14B outputs from five decoding methods on 128 GSM8K questions. The three dimensions are fluency, coherence, and comprehensibility, each on a 0–10 scale. The judge receives neither method identities nor reference answers and evaluates presentation rather than numerical correctness.

Table 4: Linguistic quality scored by Gemini 2.5 Pro on a fixed 128-example Qwen3-14B GSM8K panel. Each dimension ranges from 0 to 10.

Method	Fluency	Coherence	Comprehensibility
Greedy	9.85	9.45	9.57
+DoLa	9.80	9.20	9.39
+SLED	9.82	9.34	9.50
+NDAD	9.71	9.39	9.55
+FILD	**9.97**	**9.76**	**9.84**

FILD receives the highest mean score in all three dimensions (Table 4). Its paired bootstrap intervals relative to Greedy are positive in each dimension, whereas the corresponding intervals for the other baselines include zero. With scores near the top of the scale, the result primarily shows that revision maintains language quality on this model–task panel. The judge uses temperature zero and the following prompt.

Linguistic-quality evaluation prompt

```
You are evaluating only the linguistic quality of a model response.
Do not score whether the numerical answer is correct.  Do not use a
reference answer.  Do not infer or discuss which decoding method
produced the response.

Score each dimension with an integer from 0 to 10:  fluency
(grammatical, natural, and readable language); coherence (internally
connected reasoning without confusing jumps, contradictions, or
needless repetition); and comprehensibility (how easily a reader can
understand the response).

Question:  [question]
Response:  [response]

Return exactly one JSON object with this schema:
{"fluency":  0, "coherence":  0,
"comprehensibility":  0,
"reason":  "one concise sentence"}
```

C FILD ALGORITHM AND SUPPORTING PROPERTIES

This section connects the proposal, field construction, and commitment steps into a complete decoding procedure. Algorithm 1 specializes the budgeted method in the main text to $B = 1$, the setting used in our experiments. We then explain how the two decision stages use the same candidate continuations, derive the screening bound, and describe what a local revision preserves.

Algorithm 1 FILD with one probe per response ($B = 1$)

Input: frozen model, prompt, generation limits, and FILD settings.
Output: one completed response.

1. Follow the provisional native trajectory. At checkpoints, track the earliest entropy maximum in the observed search prefix. Select $t^\star$ when it passes the screen in Eq. (C.2) and satisfies the checkpoint persistence rule. If no pivot is selected, finish native.
2. Expand the top-K actions to h_p or termination, retaining states at the observation horizons. Score the continuations and identify the highest-scoring candidate. If it fails the proposal gate, finish native; otherwise designate it as challenger c.
3. Advance unfinished branches through the remaining horizons. If all branches have terminated, select c and proceed to commitment.
4. If a branch remains open at H, compute the depth votes and the first persistent field direction. Select c when $\tau_+ < \tau_-$; otherwise select the native branch.
5. Commit the selected continuation and resume greedy generation if unfinished. A handled probe failure returns to the native path.

C.1 FROM A PROPOSAL TO A COMMITTED REVISION

Selecting and scoring a local alternative. Pivot selection uses the observed native prefix. At predefined checkpoints, the scheduler tracks its earliest maximum-entropy position and selects it when it passes the likelihood screen and persists for the required number of checkpoints. Candidate exploration then starts from the prefix immediately before this pivot. Thus the scheduler locates a revision opportunity before the continuation evidence is used to assess the available alternatives.

Let a_0 denote the native action and $a_1, \ldots, a_{K-1}$ the remaining top-K actions in descending native probability. Each branch starts at its candidate action and extends greedily to the proposal horizon h_p or termination. The same model evaluates these partial responses using the prompt below. Its final-layer Yes–No logit difference gives q_i; population standardization gives $z_i = (q_i - \mu_q)/\sigma_q$, with $z_i = 0$ when σ_q is numerically zero. The proposal score is

$$S_i = \log p_\theta(a_i \mid s_{t^\star}) + \beta z_i, \qquad \beta \geq 0. \tag{C.1}$$

Native wins ties. A challenger must be non-native, beat S_0, and pass the entropy, standardized-margin, and raw-margin thresholds in Section A.2. These checks connect two scales of evidence: $z_c - z_0$ measures separation within the current candidate set, while $q_c - q_0$ retains the magnitude of the self-verification difference.

Candidate-continuation scoring prompt

```
Evaluate the continuation without rewriting it.  Decide whether the
response shown is on a path to a correct answer to the question.  For
an incomplete response, judge whether continuing it naturally is
likely to end correctly.  Answer exactly Yes or No.

Question:  [question]
Response so far:  [partial response]
Verdict:
```

Following a fixed challenger. Once the proposal gate admits c, the remaining lookahead follows this same challenger. The other $K - 2$ candidates provide shared references for comparing it with

native. A positive mean paired cosine-distance margin means that the challenger is closer to these references; a negative margin favors native. Comparing both branches to the same reference set makes the margin a relative preference within the candidate ensemble.

The branches are extended incrementally, retaining states already observed at horizons up to h_p. A branch that terminates contributes its last real-token state at subsequent horizons. At each horizon, three of the four sampled depths must agree to produce a directional vote; otherwise the vote is zero. The first two adjacent predefined horizons with matching nonzero votes establish the persistent direction. A zero vote interrupts a run, and a later reversal does not replace the first persistent direction.

Separating field support from commitment. The field signal and the final action answer different questions. The signal records whether challenger support persists before native support, $G_c = \mathbf{1}[\tau_+ < \tau_-]$, with a missing passage assigned $+\infty$. Commitment also depends on whether any candidate branch remains open at H. If all branches terminate, FILD adopts the eligible challenger. Otherwise, adoption requires $G_c = 1$; a negative or unresolved field retains native.

This ordering matters when a directional passage occurs early. It determines the field signal, but the branches may still reach terminal resolution before H. FILD therefore continues unfinished branches until all have ended or the maximum horizon is reached. Once every branch has ended, carrying states forward would simply repeat the terminal column. Terminal resolution uses the likelihood-aware proposal to choose among these completed branches. The open-field rule instead uses the evolving relational evidence when at least one candidate future remains unfinished.

C.2 Why screening preserves the proposal

The proposal score yields a bound on how much self-verification can compensate for a candidate's native likelihood deficit. This bound can be evaluated before opening any branches.

Theorem C.1 (Standardized-gap bound)**.** *If $K \geq 2$, $\sum_i z_i = 0$, and $K^{-1}\sum_i z_i^2 = 1$, then $\max_{i,j}(z_i - z_j) \leq \sqrt{2K}$, and the bound is tight.*

Proof. The variance condition gives $\|z\|_2 = \sqrt{K}$. For distinct indices i, j, the vector $e_i - e_j$ has norm $\sqrt{2}$, so Cauchy–Schwarz gives

$$z_i - z_j = \langle z, e_i - e_j \rangle \leq \|z\|_2 \|e_i - e_j\|_2 = \sqrt{2K}.$$

Equality holds for $z_i = \sqrt{K/2}$, $z_j = -\sqrt{K/2}$, and zero elsewhere. This vector also has zero mean and unit population variance. □

Write the likelihood deficit of candidate i as $\Delta_i = \log p_\theta(a_0 \mid s_t) - \log p_\theta(a_i \mid s_t)$. Because candidates are ordered by native probability, $\Delta_i \geq \Delta_1$ for every non-native i. Their score advantages therefore satisfy

$$S_i - S_0 = \beta(z_i - z_0) - \Delta_i \leq \beta\sqrt{2K} - \Delta_1.$$

Consequently, no candidate can beat native when

$$\log p_\theta(a_0 \mid s_t) - \log p_\theta(a_1 \mid s_t) \geq \beta\sqrt{2K}. \tag{C.2}$$

Equality is safe because native wins ties. When the score variance is numerically zero, all z_i are set to zero and the proposal follows native likelihood, so the same screening conclusion holds.

The screen certifies that a position cannot yield an eligible challenger under the proposal score. Passing it leaves the decision to the actual continuation scores and proposal gate. The theorem itself needs only two candidates; FILD's relational comparison requires $K \geq 3$ so that native and challenger have at least one shared reference.

C.3 PRESERVATION AND COMPUTATIONAL REUSE

Effect on the generated response. In the $B = 1$ procedure, rejection leaves the response on the native trajectory under deterministic execution. An accepted revision keeps the prefix before $t^\star$, changes the action at that pivot, and continues greedily along the selected branch. This is a local change to the decoding path: candidate exploration leaves the preceding tokens intact, while the changed action can affect the entire suffix. The committed lookahead tokens become part of that suffix.

For the general budgeted method, the same reasoning applies relative to the prefix at each probe. Each attempted probe consumes one budget unit regardless of acceptance, and later probes use the committed prefix. Rejection at a later probe preserves the path reached so far, which may already include an earlier revision.

What the relational comparison preserves. Cosine distance makes the field depend on relative directions of the candidate states. A common orthogonal transformation or positive rescaling of individual states leaves the distances, depth votes, and field signal unchanged. Permuting the shared references also leaves their mean paired margin unchanged. Exchanging native and challenger reverses the margins when the reference set is held fixed. These properties describe the geometry of the field; native likelihood and the proposal gate separately determine which candidate is proposed.

Reusing exploration at commitment. Each probe expands at most KH branch-token positions and evaluates the K continuation queries in one batch. Selected hidden states are recorded during the branch forwards, so field construction reuses candidate generation. At commitment, FILD keeps the selected tokens and their cache state and resumes generation from that continuation. The screen saves exploration at positions where the proposal cannot change native, while terminal resolution stops expansion when every branch has ended. The KH bound covers branch expansion; query evaluation and the native response remain separate parts of decoding.

D CASE STUDIES

The following GSM8K cases illustrate four outcomes of relational admission: retaining a useful proposal, rejecting a harmful one, missing a useful correction, and accepting an incorrect revision. They come from the diagnostic base-policy panel used in the original ablation campaign. Each case presents a condensed question, a reference calculation, and the recorded final answers and field outcome. Reference calculations explain the task; they are not excerpts from model-generated reasoning. Proposal-only commits the challenger without relational admission.

D.1 SUCCESSFUL ADMISSION DECISIONS

Salary allocation: retaining a useful correction

Question. Zaid earns \$6,000. He spends one quarter on rent and one third on fuel, then donates half of the remaining money. After giving \$200 to his daughter and \$700 to his wife, how much does he have left?

Reference calculation. Rent and fuel leave $6000 - 1500 - 2000 = 2500$. The donation leaves $2500/2 = 1250$, and the final household payments leave $1250 - 200 - 700 = 350$. The donation is calculated from the money remaining after rent and fuel.

Recorded answers. Greedy: 3850; Proposal-only: 350; FILD: 350. The reference answer is 350.

Admission and outcome. The recorded field outcome favors the challenger, and FILD retains the correct proposal. Here, proposal formation supplies a useful alternative to the native answer, and relational admission preserves its benefit.

Equal donations: rejecting a harmful correction

Question. Ronnie receives \$5 and Rissa receives three times as much. Both give the same amount to their younger sister. Rissa keeps four fifths of her original money. How much does Ronnie have left?

Reference calculation. Rissa starts with \$15 and gives away $15(1 - 4/5) = 3$. Ronnie gives the same \$3, leaving $5 - 3 = 2$. The shared quantity is the amount donated, rather than the fraction of each person's starting money.

Recorded answers. Greedy: 2; Proposal-only: 5; FILD: 2. The reference answer is 2.

Admission and outcome. The recorded field outcome favors native. FILD rejects the proposed change and preserves the correct answer. This case illustrates the value of admission when a challenger passes the proposal gate but would introduce an error into an already correct response.

Where the benefit enters. The two successes locate the benefit at different stages. In the salary case, the proposal already contains the correct answer: admission preserves a gain created by lookahead. In the equal-donations case, native is already correct, so the benefit of admission is avoiding the error that Proposal-only would introduce. These cases distinguish finding a better answer from deciding whether to replace the current one. A successful decoder needs both the opportunity to depart from native and the ability to retain it when a proposed departure is harmful.

D.2 MISSED CORRECTIONS AND ERRONEOUS REVISIONS

Perfume sales: missing a correct proposal

Question. A company sells five units of each of four vanilla fragrances and two units of each of eight fruity fragrances. How many more vanilla units than fruity units does it sell?

Reference calculation. The totals are $4 \times 5 = 20$ vanilla units and $8 \times 2 = 16$ fruity units. Their difference is $20 - 16 = 4$. The comparison requires the total sales of each category.

Recorded answers. Greedy: 3; Proposal-only: 4; FILD: 3. The reference answer is 4.

Admission and outcome. The field is recorded as unresolved. Under the open-field rule, the absence of a persistent direction retains native, so the correct proposal is lost. This outcome shows the cost of requiring relational corroboration: a useful alternative can be available without meeting the condition for commitment.

Overlapping subject groups: accepting an incorrect proposal

Question. A class has 20 students. Five are good only at mathematics, eight are good only at English, and the rest are good at both. How many students are good at mathematics?

Reference calculation. The overlap contains $20-5-8 = 7$ students. Adding the mathematics-only group gives $5 + 7 = 12$, equivalently $20 - 8 = 12$. Both the mathematics-only and the overlapping groups count.

Recorded answers. Greedy: 12; Proposal-only: 13; FILD: 13. The reference answer is 12.

Admission and outcome. The recorded field outcome favors the challenger, and FILD adopts the incorrect proposal. Persistent relational support can therefore accompany an erroneous answer. Agreement within the candidate field does not by itself establish that the selected continuation satisfies the question's constraints.

Two distinct failure modes. The perfume case contains a correct alternative, so its failure occurs at admission rather than in the availability of a useful proposal. Retaining native under unresolved support forgoes that correction. The subject-groups case has the opposite structure: the proposal is wrong, and positive relational support allows it through. These two outcomes expose different requirements for an admission signal: recognizing useful departures and screening out harmful ones. Simply accepting more proposals would not address both failures, because the subject-groups error is already an accepted revision.

Relational support and answer correctness. The salary and subject-groups cases both record challenger-favoring support, yet their accepted answers have opposite correctness. Relational support measures how a continuation compares with the other candidate branches; the reference calculation checks whether its answer satisfies the task. Agreement within the field can be useful for selection without being a certificate of correctness. Together with the equal-donations and perfume cases, this comparison shows why the quality of admission depends on which proposals it accepts and rejects, rather than the acceptance rate alone. The four cases illustrate these distinctions; their selection does not estimate how often each outcome occurs.