CONTINUOUS LANGUAGE BRIDGE

Anonymous authors
Paper under double-blind review

ABSTRACT

Improving predictive accuracy remains a central goal of language modeling, with scaling model capacity, training data, and computation driving much of the progress. Pursuing this goal also motivates model-native self-correction: allowing a model to use its tentative prediction as feedback and reconsider the same next-token decision before emission. We introduce Continuous Language Bridge (CLB), which connects the observed predecessor to a draft predicted continuation in embedding space. A shared second pass reassesses the same next-token target at one selected bridge state, and a categorical mixture produces the final prediction. The framework retains exact autoregressive likelihood. We validate the CLB across different benchmark datasets and also from dense to sparse MOE structures at different scales. Specifically, CLB beats the naive AR and looped model by 1.27% and 0.47% on LM1B dataset, respectively. Across seven external datasets, CLB reduces perplexity relative to AR by an average of 8.79%, 5.79%, and 13.76% at the 130M, 700M, and 2.1B scales, respectively, with the MoE transfer result measured at 70K updates. The project code will be open-sourced.

1 INTRODUCTION

Scaling has been a central driver of progress in language modeling. Larger models trained on more data can achieve lower predictive loss, with the attainable improvement depending on the available training compute. Kaplan et al. (2020) characterize these relationships through empirical scaling laws, describing predictable trends in cross-entropy loss across model and data scales. These laws also guide the allocation of a fixed compute budget between model capacity and data exposure.

Moreover, post-training further develops behavior beyond the pretraining objective. InstructGPT combines demonstrations and preference feedback (Ouyang et al., 2022); SCoRe trains response-level self-correction through multi-turn reinforcement learning (Kumar et al., 2025). These approaches make feedback part of learning rather than merely an external check.

Meanwhile, at inference time, chain-of-thought prompting makes intermediate reasoning explicit (Wei et al., 2022). Self-refine alternates responses, feedback, and revision (Madaan et al., 2023), while reflection carries verbal feedback across trials (Shinn et al., 2023). Such processes allow reconsideration before a final answer, at the level of reasoning steps, responses, or actions.

Besides those scaling methods, our complementary objective is to make reconsideration native to each next-token decision. We study whether exposing the model's own draft proposal to further computation improves that conditional distribution before emission. We make this proposal available for reassessment through a continuous interpolation between its draft embedding and the observed-token embedding. The interpolation coordinate regulates the strength of predictive feedback, allowing the model to revisit the same next-token decision without committing the proposed continuation to the observed context. This motivates *continuous verification*, meaning learned reassessment rather than a correctness test.

In *Continuous Language Bridge* (CLB), the first-pass probabilities define a draft token embedding. Bridging this embedding with the last observed token produces a continuous family of inputs. During decoding, a shared Transformer evaluates one selected input for the same next-token target, using earlier clean-context states and its own current state. Both training and inference construct the bridge from the observed prefix and the model's predictions, while the normalized readout preserves exact autoregressive likelihood.

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.

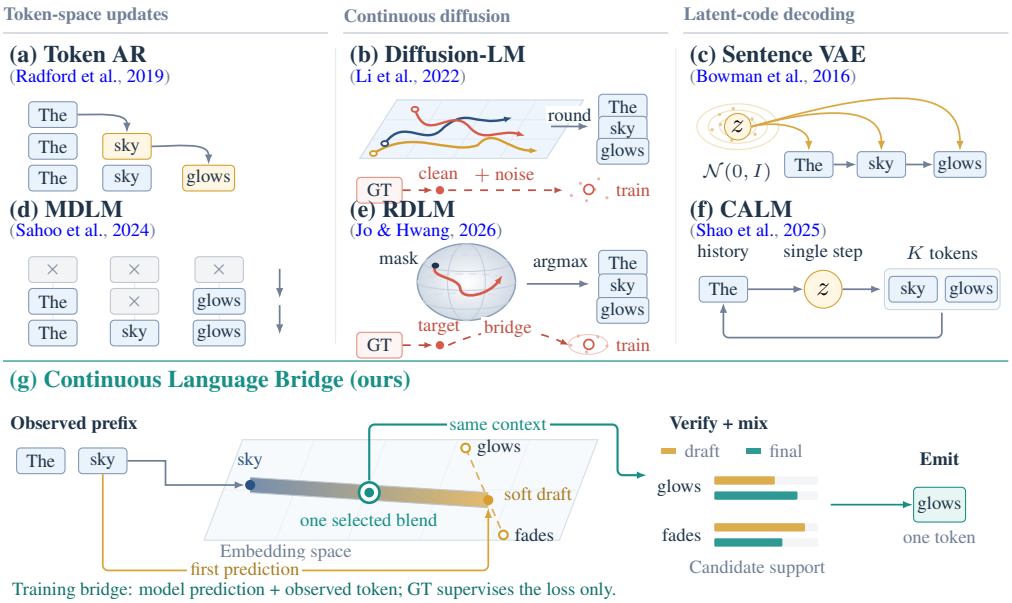


Figure 1: Representative generation mechanisms. Solid paths depict generation; dashed coral insets in (b,e) show training states built from interpolation with *ground truth (GT)*: Gaussian-corrupted embeddings in Diffusion-LM and endpoint-conditioned spherical bridges in RDLM. CLB instead constructs its bridge from prefix-derived predictions during both training and generation; GT supervises the loss but never enters that bridge. Its ribbon shows available blends, not an executed trajectory. Boxes are tokens ($\times$: mask), points are continuous states.

We evaluate CLB on LM1B and OpenWebText with dense and sparse backbones. The LM1B comparison trains CLB, AR, and a repeated-computation control on the same 10B tokens, where CLB achieves lower test perplexity than both controls. OWT experiments cover three model scales and seven external corpora. Further experiments examine feedback strength, noise, and expert routing. Our contributions are threefold:

- we introduce a within-token feedback framework combining an observed-to-predicted embedding bridge, clean-context reassessment of the same target, and a normalized readout preserving exact autoregressive likelihood;
- we instantiate CLB with dense and mixture-of-experts backbones, including a sparse variant that reuses selected expert identities and recomputes their combination weights;
- we conduct extensive experiments on LM1B and OpenWebText, together with evaluations on seven external datasets, demonstrating CLB's perplexity improvements over autoregressive baselines across multiple scales and both dense and mixture-of-experts architectures.

In the following, Sec. 2 summarizes different language models and make a comparison of the similarity and differences between them. Sec. 3 instructs the method in detail, and Sec. 4 gives a diverse evaluation of the proposed method. The appendices collect proofs, implementation details, numerical records, and additional ablations.

2 Related Work

Continuous language representations. Sentence VAEs condition an autoregressive decoder on a global continuous latent (Bowman et al., 2016). Diffusion-LM denoises continuous word embeddings (Li et al., 2022), while the Riemannian Diffusion Language Model (RDLM) evolves categorical representations on hyperspheres (Jo & Hwang, 2026). Projected Autoregression predicts next-token vectors (Naparstek, 2026); CALM predicts a chunk vector with a single-step noise-conditioned head, decodes it to tokens, and feeds those tokens back autoregressively (Shao et al., 2025). CLB feeds a soft draft predicted embedding back into the model to reassess the next-token distribution. Fig. 1 shows how these models use continuous states during generation.

Continuous reasoning and predictive feedback. Coconut feeds hidden states back to extend a latent reasoning sequence (Hao et al., 2024). Soft Thinking instead advances that sequence with probability-weighted token embeddings, without additional training (Zhang et al., 2025). CLB places feedback *within one next-token decision*: both passes address the same target at the same position, and only the final emitted token extends the context. The observed-token anchor and clean-context reuse define this interface, learned through next-token likelihood. The distinction concerns the organization of generation, not merely the choice of a draft representation. CoCoMix interleaves predicted sparse-autoencoder concepts during pretraining (Tack et al., 2026); Analog Bits and self-conditioned embedding diffusion reuse clean-state predictions during denoising (Chen et al., 2023; Strudel et al., 2022).

Diffusion, hybrids, and consistency. Discrete diffusion spans transition-based D3PM (Austin et al., 2021), discrete-score estimation in SEDD (Lou et al., 2024), and masked denoising in MDLM (Sahoo et al., 2024). Block Diffusion and Eso-LMs connect autoregressive and diffusion constructions (Arriola et al., 2025; Sahoo et al., 2026). Separately, Consistency Models, improved consistency training, and continuous-time sCM learn endpoint agreement along a generative trajectory (Song et al., 2023; Song & Dhariwal, 2024; Lu & Song, 2025). CLB uses neither token corruption nor a cross-coordinate consistency loss. Its coordinate sets the interpolation between an observed token embedding and a prefix-derived draft prediction, not like other diffusion-based methods that use the GT token. *Moreover, we have also evaluated that the diffusion-based GT-utilized blending process actually cannot help the language generation (please refer to Sec. 4.4 and Fig. 3).*

Shared computation and categorical readouts. Universal transformers and recurrent-depth language models reuse transformations to refine hidden states (Dehghani et al., 2019; Geiping et al., 2026). A mixture of softmaxes increases output expressiveness by combining multiple softmax distributions derived from the same context (Yang et al., 2018). CLB obtains its second component by passing the observed-to-predicted bridge state through the shared backbone to reassess the same next-token target. Sec. 4.4 compares CLB with a tied-layer Loop baseline that repeats computation without this predictive feedback.

Sparse expert computation. Switch Transformers scale capacity with sparse activation (Fedus et al., 2022), while DeepSeekMoE distinguishes shared and routed experts (Dai et al., 2024). CLB retains the first pass's selected expert identities and recomputes their outputs and weights from the bridge input.

Fig. 1 compares the representations and update steps used by representative language models. Token AR extends the prefix sequentially, while masked diffusion progressively fills masked positions. Diffusion-LM and RDLM evolve continuous token representations; sentence VAEs condition an autoregressive decoder on a global latent, and CALM decodes predicted latent vectors into token chunks. CLB uses a continuous representation to revisit a prediction within one next-token decision. Its bridge connects the last observed token embedding to a probability-weighted mean of candidate embeddings from the initial prediction. With the observed prefix held fixed, a shared second pass reassesses the same target at one selected bridge state, and a learned mixture combines both predictions before token emission.

3 Proposed Method

CLB connects the observed predecessor to a soft predicted continuation (Fig. 2). Before emitting the next token, the model evaluates one state on this bridge using a second pass through the shared backbone and the clean context. We call this learned reassessment of the same next-token target *continuous verification*.

3.1 From an observed token to a soft prediction

Let $x_{1:N} \in \mathcal{V}^N$ be a context of length N from a vocabulary of size V. At position $1 \le i < N$, x_i is the observed predecessor, $x_{<i}$ the earlier context, and $y_i = x_{i+1}$ the target. The embedding matrix $\mathbf{E} \in \mathbb{R}^{d \times V}$ has token column $\mathbf{E}_v \in \mathbb{R}^d$. A causal backbone $\mathcal{F}_\theta(\cdot)$ with parameters θ gives hidden

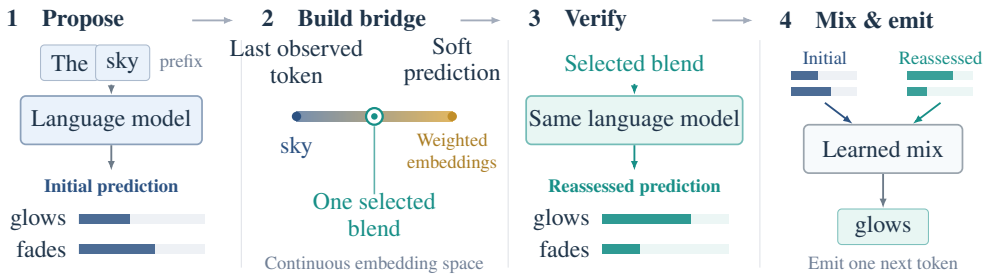


Figure 2: CLB before token emission. Initial probabilities weight candidate embeddings; one blend with the last observed token is reassessed by the same model using earlier clean context and its own current state. A gate from the first-pass hidden state mixes the initial and reassessed probabilities. The soft endpoint is detached during training.

state $\mathbf{H}_i \in \mathbb{R}^d$ and tied-head prediction:

$$\mathbf{H}_i = \mathcal{F}_\theta(x_{\le i}), \qquad \mathbf{L}_i = \mathbf{E}^\top \mathbf{H}_i, \qquad \mathbf{P}_i = \mathrm{softmax}(\mathbf{L}_i).$$

Here $\mathbf{L}_i \in \mathbb{R}^V$ are the vocabulary logits. The next-token probability vector $\mathbf{P}_i \in \Delta^{V-1}$ lies in the probability simplex $\Delta^{V-1} := \{\mathbf{P} \in \mathbb{R}^V \mid P_v \ge 0 \text{ for all } v,\ \sum_{v=1}^V P_v = 1\}$. Its scalar entry $P_{i,v}$ denotes the probability assigned to a vocabulary token v. Let $S_{i,K} = \mathrm{TopK}(\mathbf{P}_i)$ contain the K highest-probability candidates. The soft endpoint retains and renormalizes their support:

$$\mathbf{D}_i^{(K)} = \sum_{v \in S_{i,K}} \bar{P}_{i,v} \mathbf{E}_v, \qquad \bar{P}_{i,v} = \frac{P_{i,v}}{\sum_{u \in S_{i,K}} P_{i,u}} \quad (v \in S_{i,K}). \tag{1}$$

Set $\bar{P}_{i,v} = 0$ outside $S_{i,K}$, so $\bar{\mathbf{P}}_i \in \Delta^{V-1}$ and $\mathbf{D}_i^{(K)} \in \mathbb{R}^d$. The soft endpoint summarizes the retained candidates as a probability-weighted embedding mean. Together with the observed token embedding, it supplies two endpoints computed entirely from the observed prefix.

3.2 A CONTINUOUS BRIDGE WITH A SHARED PREDICTION TARGET

Let $t \in [0, 1]$ denote the continuous interpolation coordinate, distinct from context length N. For a candidate v, the segment $t\mathbf{E}_{x_i} + (1-t)\mathbf{E}_v$ introduces its representation without appending it to the context. Its expectation under $v \sim \bar{\mathbf{P}}_i$ defines the soft-bridge map $\mathcal{B}_{\theta,t}^{(K)}(\cdot)$:

$$\mathcal{B}_{\theta,t}^{(K)}(x_{\le i}) = t\mathbf{E}_{x_i} + (1-t)\mathbf{D}_i^{(K)} \in \mathbb{R}^d. \tag{2}$$

At $t = 1$ the bridge is the observed embedding; at $t = 0$ it is the soft endpoint. Decreasing t increases the proposal's contribution. The observed endpoint supplies a lexical anchor; interpolation controls feedback strength without treating a predicted token as observed text.

Every coordinate addresses the *same* prediction problem: infer y_i from the observed prefix and a partially introduced proposal. Range training applies the same next-token supervision to bridge states sampled at different coordinates. The expectation defining the bridge averages candidate embeddings before they enter the nonlinear verification network.

3.3 CONTINUOUS VERIFICATION WITH CLEAN CONTEXT

The first pass stores clean keys and values at each backbone layer. The shared-weight refinement pass receives the bridge state $\mathcal{B}_{\theta,t}^{(K)}(x_{\le i})$ and coordinate features $\mathcal{C}(t) \in \mathbb{R}^d$. At position i, it attends only to earlier clean entries $j < i$ and its own current verification state. It excludes the current clean entry and all other verification positions. With head and layer indices suppressed, this is

$$\widetilde{\mathbf{O}}_i = \mathcal{A}\Big(\widetilde{\mathbf{Q}}_i,\ [\mathbf{K}_{<i}; \widetilde{\mathbf{K}}_i^\top],\ [\mathbf{V}_{<i}; \widetilde{\mathbf{V}}_i^\top]\Big), \tag{3}$$

where $\mathcal{A}(\cdot,\cdot,\cdot)$ is scaled dot-product attention with query, key, and value arguments, $\widetilde{\mathbf{Q}}_i, \widetilde{\mathbf{K}}_i, \widetilde{\mathbf{V}}_i, \widetilde{\mathbf{O}}_i \in \mathbb{R}^{d_h}$, $d_h = d/n_h$ for n_h heads, and $\mathbf{K}_{<i}, \mathbf{V}_{<i} \in \mathbb{R}^{(i-1)\times d_h}$. The semicolon concatenates rows. The same mask permits parallel teacher-forced training and causal cached decoding.

For L layers and n cached tokens, standard full-head attention requires $2Lnd$ persistent KV elements per sequence, as in the corresponding AR backbone. Verification reuses this clean history and stores its current-position keys and values transiently. The second pass adds computation and temporary activations; the $2Lnd$ count covers persistent KV storage only.

The refinement hidden state $\widetilde{\mathbf{H}}_i \in \mathbb{R}^d$ gives vocabulary logits $\widetilde{\mathbf{L}}_i = \mathbf{E}^\top \widetilde{\mathbf{H}}_i \in \mathbb{R}^V$ and probabilities $\widetilde{\mathbf{P}}_i = \mathrm{softmax}(\widetilde{\mathbf{L}}_i)$. A first-pass-conditioned gate mixes the two distributions:

$$\mathbf{\Pi}_i = \mathrm{softmax}(\mathbf{W}_g \mathbf{H}_i + \mathbf{B}_g), \qquad \widehat{\mathbf{P}}_i = \Pi_{i,1} \mathbf{P}_i + \Pi_{i,2} \widetilde{\mathbf{P}}_i, \tag{4}$$

where $\mathbf{W}_g \in \mathbb{R}^{2\times d}$, $\mathbf{B}_g \in \mathbb{R}^2$, $\mathbf{\Pi}_i \in \Delta^1$, and $\widehat{\mathbf{P}}_i \in \Delta^{V-1}$. Probability mixing preserves normalization and gives $\widehat{P}_{i,v} - P_{i,v} = \Pi_{i,2}(\widetilde{P}_{i,v} - P_{i,v})$: support can rise or fall before emission. The gate computes its mixture weights from the first-pass hidden state and is trained through the likelihood objective.

For fixed evaluation t and K, $\widehat{\mathbf{P}}_i$ is a normalized function of $x_{\le i}$ and the model parameters. Hence

$$\mathcal{P}_\theta(x_{2:N} \mid x_1) = \prod_{i=1}^{N-1} \widehat{P}_{i,x_{i+1}} \tag{5}$$

is an exact autoregressive likelihood. The bridge and clean cache are prefix-derived, and the gate combines normalized distributions with nonnegative weights summing to one.

We analyze whether mixing the first-pass and verification predictions can reduce the expected next-token log loss. Both predictors are held fixed throughout this analysis. Fix both predictors and let $(\mathbf{C}, Y)$ denote the observed context token vector and true next token drawn from the evaluation distribution. Write $P_Y(\mathbf{C})$ and $\widetilde{P}_Y(\mathbf{C})$ for the probabilities assigned to the class Y by the first-pass and verification predictors, respectively. The gate uses $\mathbf{H} = \mathcal{F}_\theta(\mathbf{C})$, the first-pass hidden vector at the last observed position, to choose the verification weight $a(\mathbf{H})$ without observing Y. Define $U = \widetilde{P}_Y(\mathbf{C})/P_Y(\mathbf{C}) - 1$ as the relative change in the probability assigned to Y, with conditional mean $D(\mathbf{H}) = \mathbb{E}[U \mid \mathbf{H}]$ and second moment $S(\mathbf{H}) = \mathbb{E}[U^2 \mid \mathbf{H}]$. All expectations use the evaluation distribution. Assume that both predictors assign strictly positive probability to every token almost surely, have finite expected log losses, and satisfy $\mathbb{E}[U^2] < \infty$. For $b \in [0, 1]$, let $\mathcal{A}_b$ contain all measurable gate functions with values in $[0, b]$. Let $\mathcal{R}(a)$ denote the expected next-token log loss of mixing the verification and first-pass distributions with weights $a(\mathbf{H})$ and $1 - a(\mathbf{H})$, respectively. Thus, $\mathcal{R}(0)$ is the first-pass log loss.

Theorem 1 (Mixture risk and gate error)**.** *Under the preceding assumptions, every $a \in \mathcal{A}_{1/2}$ satisfies*

$$\mathcal{R}(a) - \mathcal{R}(0) \le -\mathbb{E}[aD] + 2\mathbb{E}[a^2 S]. \tag{6}$$

Let $D_+ = \max(D, 0)$ and $a^\star = \min\{\frac{1}{2}, D_+/(4S)\}$, with $a^\star = 0$ when $S = 0$. If $\widehat{a} \in \mathcal{A}_1$ satisfies $\mathcal{R}(\widehat{a}) \le \inf_{a\in\mathcal{A}_{1/2}} \mathcal{R}(a) + \varepsilon$ for $\varepsilon \ge 0$, then $\mathcal{R}(\widehat{a}) \le \mathcal{R}(0) - \frac{1}{2}\mathbb{E}[a^\star D_+] + \varepsilon$. The improvement is strict whenever $\frac{1}{2}\mathbb{E}[a^\star D_+] > \varepsilon$.

The theorem characterizes probability-mixture readouts through the gain available from the second predictor and the error in choosing its weight. Its $1/2$ cap belongs to the reference class; CLB uses an uncapped linear softmax gate. For this learned gate, the bound is conditional on the stated comparison inequality. The proof and a successive-update corollary are given in Supplementary Sec. A.

3.4 Training and Inference

The scalar-valued objective $\mathcal{L}(\theta)$ combines mixture negative log-likelihood, first-pass supervision, and a logit z-loss. With $z_i = \log \sum_{v\in\mathcal{V}} \exp(\widetilde{L}_{i,v})$, it is

$$\mathcal{L}(\theta) = \frac{1}{N-1} \sum_{i=1}^{N-1} \left[-\log \widehat{P}_{i,y_i} - \lambda \log P_{i,y_i} + \beta z_i^2 \right]. \tag{7}$$

The scalar weights are $\lambda = 1$ and $\beta = 10^{-4}$ for the OWT clean-KV runs, and $\lambda = 1$, $\beta = 0$ for LM1B. The predictive endpoint is detached before verification; the auxiliary loss trains the first pass

directly. The observed anchor and shared backbone remain trainable. For the default OWT bridge, training samples one integer K uniformly from $\{5, \ldots, 50\}$ per forward call; evaluation uses $K = 10$. Supplementary Sec. E.3 records the second-corpus configuration.

Range training samples $t_i \sim \mathrm{Uniform}(0, 1)$ independently per token; default fixed-coordinate training uses $t_i = 0.6$. All coordinates supervise the same y_i without its embedding or an error label. Headline native and cross-corpus readouts use the prespecified $t = 0.6$; historical validation-selected coordinates are identified separately. LM1B uses a separate held-out test set. Generation evaluates one verification state at a fixed coordinate and mixes predictions before selection. Only the emitted token extends the context.

3.5 Extending continuous verification to mixture-of-experts models

The MoE extension replaces dense feed-forward sublayers while preserving the bridge and readout. At each layer, four always-active shared experts $\mathcal{S}_s(\cdot)$ are combined with four of eleven routed experts $\mathcal{G}_j(\cdot)$. The clean pass selects routed identities $\mathcal{R}$; fixed-ID verification reuses those identities but recomputes their outputs and softmax weights $\widetilde{w}_j$ from the bridge-conditioned input $\widetilde{\mathbf{Z}}$. The verification map $\widetilde{\mathcal{M}}(\cdot)$ is

$$\widetilde{\mathcal{M}}(\widetilde{\mathbf{Z}}) = \sum_{s=1}^{4} \mathcal{S}_s(\widetilde{\mathbf{Z}}) + \gamma \sum_{j \in \mathcal{R}} \widetilde{w}_j \mathcal{G}_j(\widetilde{\mathbf{Z}}) \in \mathbb{R}^d. \tag{8}$$

The clean-pass map $\mathcal{M}(\cdot)$ uses the same expression without tildes. A separate Fourier coordinate feature conditions the router, while a count-updated balancing bias affects selection only. The shared contribution is an unweighted sum, and the unconstrained learned scale γ controls the routed contribution.

The shared/reselected control chooses four routed identities again at verification. The no-shared control instead routes to eight of all fifteen experts without a separate shared sum; its clean pass still uses four shared plus four routed experts. Our 2.1B backbone has 36 layers, width 1,280, and 20 heads. The AR and bridge runs have different dense initializations and computation. Supplementary Secs. B.3 and E.1 specify expert maps, router equations, and comparisons.

4 Experiments

We evaluate CLB on LM1B at matched training-token budgets and on OpenWebText at 130M and nominal 700M dense scales and a 2.1B MoE scale (Tables 1–2).

4.1 Datasets, model scales, and comparison protocols

LM1B. LM1B contains shuffled English sentences from WMT11 (Chelba et al., 2013). We train AR, Loop, CLB, and an MDLM adaptation from scratch on the same 10B input tokens, using GPT-2 tokenization, length 128, and global batch 512. The categorical models have approximately 130.7M parameters. We score 2,032,000 targets in 16,000 test blocks; 2,000 disjoint validation blocks select earlier checkpoints' coordinates. Final comparisons fix $t = 0.6$. CLB denotes range training; fixed-coordinate training remains an ablation.

OpenWebText. OWT comprises English Reddit-linked web documents (Gokaslan & Cohen, 2019). We use GPT-2 tokenization (Radford et al., 2019), length 1,024, and effective batch 128. Validation scores in 2K contiguous chunks, each length 1,024 tokens, $\mathrm{PPL} = \exp(\mathrm{CE})$ with CE in nats.

Experimental setup. Each configuration uses one training seed. CLB uses two shared-backbone passes per token versus AR's one. For the training of OWT MOE structure, we utilize the pretrained 700M model to initialize the experts. Specifically, we divide the original parameter of FFN into 4 experts as shown in Eq. (8). And we have also set an additional 11 experts to be routed. During the initialization phase of that 11 experts, we independently sample and copy 1,280 intermediate channels from the original dense FFN. Each sampled channel then has a 50% probability of being reinitialized, replacing both its input and output weights with Gaussian samples whose mean and standard deviation match those of the respective original weight matrices. Thus, each routed expert retains approximately

Table 1: LM1B language modeling.

Our experiments(130M/10B)		
Model	KV ↓	PPL ↓
AR	36	27.85
Loop	72	27.62
CLB	36	**27.49**
MDLM (raw)	–	44.92
MDLM (EMA)	–	41.95
Published reference(130M/33B)		
AR		22.32
SEDD		$\leq$ 32.79
MDLM		$\leq$ 27.04
D3PM		$\leq$ 28.51

Table 2: OpenWebText language modeling.

Our experiments			
Model	Scale / budget	KV	PPL ↓
AR	130M /21B	36	21.64
CLB		36	**21.28**
AR	700M /10B	180	16.99
CLB		180	**16.66**
MOE-AR	2.1B /20B	180	15.30
MOE-CLB		180	**15.04**
Published reference(130M/262B)			
AR			17.54
SEDD			$\leq$ 24.10
MDLM			$\leq$ 23.21

Published values from Sahoo et al. (2024). $\leq$ retains their bound notation; local MDLM values are Monte Carlo estimates, not exact PPL. Published budgets retain the source's masking adjustment for diffusion. KV: analytical BF16 KiB per prefix token. In Table 2, dense budgets count tokens; 60K counts MoE-stage updates after dense initialization.

half of its sampled pretrained channels while receiving newly initialized weights for the remainder. Supplementary Secs. C.1–C.2, E.1, and E.3 give the full training and evaluation protocols. MDLM reports conditional Monte Carlo $\exp(\widehat{\text{NELBO}})$, not exact categorical PPL.

4.2 LM1B AT MATCHED TRAINING-TOKEN BUDGETS

Table 1 illustrates the experimental results on LM1B dataset, where CLB reaches 27.49 test PPL, improving on AR by 1.27% and Loop by 0.47%. All implemented models complete the pre-training process of 10B tokens. CLB's analytical KV footprint equals AR's and is half that of Loop's 24 tied-layer applications, which further indicates the effectiveness and resource-saving characteristics of the proposed method. Supplementary Sec. E.3 gives learning curves, coordinate ablations, and sampled-training-set results.

Tables 1–2 also report MDLM (Sahoo et al., 2024), SEDD (Lou et al., 2024), and a D3PM-equivalent reimplementation (Austin et al., 2021; Sahoo et al., 2024). Note that these published method are trained with BERT-tokenized on LM1B, which are not directly comparable to our GPT-2 results.

4.3 OPENWEBTEXT ACROSS DENSE AND SPARSE SCALES

Table 2 shows consistent perplexity improvements with CLB across dense and sparse backbones, spanning 130M to 2.1B parameters. The 130M pair is nearly parameter-matched. At 70K updates, the nominal 700M dense model reduces perplexity by 1.95%, while the 2.1B fixed-expert MoE at 60K MoE-stage updates achieves 15.04 PPL compared with AR's 15.30, a 1.72% reduction at $t = 0.6$. The MoE result is particularly encouraging: these gains are obtained while both passes share backbone parameters and reuse the same selected experts, without introducing experts dedicated to verification. This makes the current implementation a starting point for a broader architectural exploration. With a larger pool of sparsely activated experts, the two passes could even use distinct routed expert sets, allowing one group to specialize in proposing continuations and another in reassessing them. Such a design would give verification its own learned expertise, beyond reusing the computation developed for initial prediction.

Table 3 tests OWT-trained checkpoints on seven external collections without retraining or corpus-specific tuning. Our AR and CLB evaluations use context 1,024, an observed first token per block, and separator-free streams capped at 2M tokens; CLB uses $t = 0.6$. LAMBADA uses token-level stream PPL, and pretraining overlap has not been assessed. CLB improves over its AR counterpart on all seven datasets at each scale. Equal-weight mean relative PPL reductions are 8.79% at 130M, 5.79%

Table 3: Cross-corpus perplexity of OWT-trained models. Bold marks the lower PPL within each of our same-scale AR/CLB pairs; AR follows (Radford et al., 2019). The last column illustrates the mean reduction, which is $\frac{1}{7}\sum_{d=1}^{7}(\text{score}_d/\text{PPL}_{\text{AR},d} - 1) \times 100\%$ against the corresponding AR baseline and score_d is unrounded CLB PPL, or the published MDLM PPL upper bound on dataset d.

Scale	Model	PTB	Wiki	LM1B	LAMBADA	AG	PubMed	Arxiv	Redu. ↓
130M	AR	109.14	32.72	64.14	59.78	77.31	58.01	55.77	–
	CLB	**101.13**	**30.58**	**63.50**	**55.58**	**72.94**	**47.27**	**47.14**	**-8.79%**
700M	AR	79.77	25.13	52.01	47.23	50.77	40.32	42.72	–
	CLB	**73.55**	**23.01**	**51.65**	**44.61**	**49.08**	**38.12**	**38.76**	**-5.79%**
2.1B	AR	70.16	23.47	48.31	60.01	42.61	49.41	48.09	–
	CLB	**64.27**	**20.85**	**48.11**	**46.14**	**41.12**	**35.80**	**37.40**	**-13.76%**
Published reference: (Sahoo et al., 2024) (trained 524B tokens on OWT)									
AR		82.05	25.75	51.25	51.28	52.09	49.01	41.73	–
MDLM ($\leq$)		95.26	32.83	67.01	47.52	61.15	41.89	37.37	+8.49%

Table 4: Multiple-choice accuracy (%, ↑) of OWT-trained models under full-text-PPL scoring (Supplementary Sec. J.1). Mean weights the five tasks equally; Δ is CLB minus same-scale AR in percentage points. Bold marks the higher score within each scale, including ties.

Scale	Model	PIQA	ARC-E	ARC-C	WinoGrande	HellaSwag	Mean	Δ (pp)
130M	AR	59.96	39.14	**23.46**	**52.33**	29.04	40.79	–
	CLB	**61.04**	**41.12**	**23.46**	52.01	**29.34**	**41.40**	+0.61
700M	AR	63.06	42.55	24.32	50.59	33.99	42.90	–
	CLB	**64.09**	**43.73**	**24.83**	**50.99**	**35.02**	**43.73**	+0.83

Benchmarks: PIQA (Bisk et al., 2020); ARC-E and ARC-C (Clark et al., 2018); WinoGrande (Sakaguchi et al., 2021); HellaSwag (Zellers et al., 2019).

at nominal 700M, and 13.76% at 2.1B MoE. Supplementary Secs. H and I retain all 26 configurations and 316 readouts, coordinate scans, and format diagnostics.

Multiple-choice accuracy. Table 4 complements perplexity with answer-selection accuracy on five benchmarks. CLB increases task-averaged accuracy from 40.79% to 41.40% at 130M and from 42.90% to 43.73% at nominal 700M, gains of 0.61 and 0.83 percentage points, respectively. The 700M model improves on all five tasks; at 130M, CLB improves on three, ties on ARC-Challenge, and slightly declines on WinoGrande. Supplementary Sec. J provides the scoring protocol and selected paired generation examples.

4.4 Ablation Studies

Training and inference coordinates. Fig. 3 (a,b) compares coordinate sensitivity under different training schedules. Among the OWT fixed-coordinate runs trained for the same number of updates, $t = 0.6$ gives the lowest validation-selected PPL; the range-trained checkpoint has a slightly larger training budget. On LM1B, range training gives lower PPL than fixed-0.6 training after 10B tokens. Supplementary Sec. F gives the full scans.

MoE routing and continuation. Fig. 3 (c) shows the MoE continuation runs, with validation-selected bridge results through 60K. Table 2 reports the same 60K endpoint at the prespecified $t = 0.6$. We can see from the training process that MOE-CLB consistently beats the MOE-AR, indicating the stabilization and effectiveness of the proposed method.

Diffusion-like State Interpolation. Diffusion models based on Gaussian noise or masking construct intermediate training states by corrupting ground-truth targets. In contrast, CLB builds its bridge from the observed predecessor and a soft draft predicted continuation, without using the ground-truth

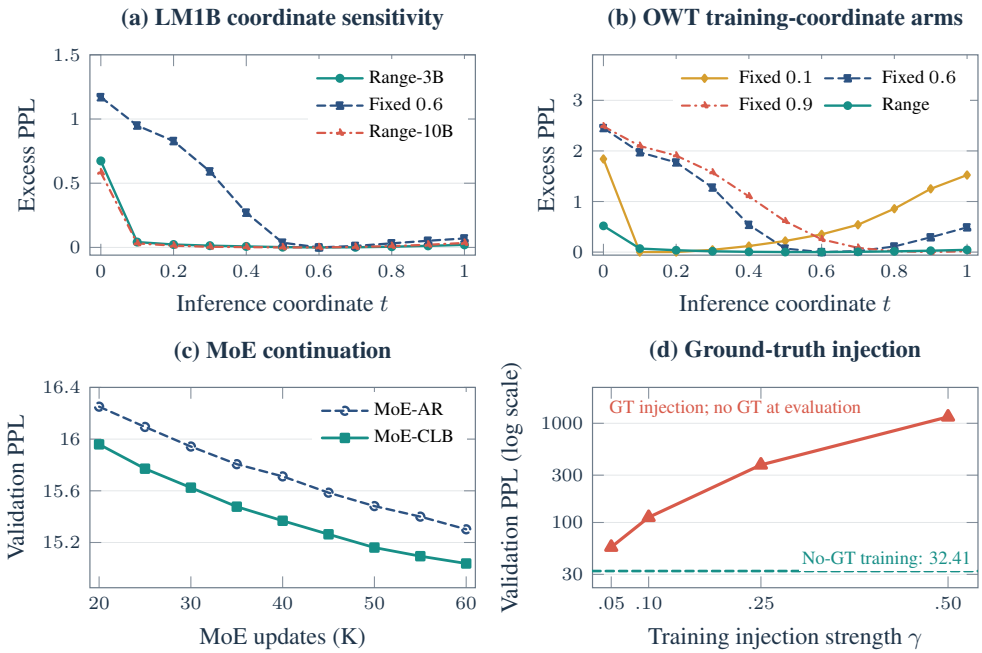


Figure 3: Ablation studies. **(a,b)** indicates PPL above each checkpoint's validation minimum. **(c)** Historical 2.1B fixed-expert MoE with validation-selected coordinates and different initialization/compute. Both curves stop at 60K; checkpoints are not independent replications. **(d)** Top-10 target injection at 10K, evaluated without injection; best $t \in \{0.4, 0.6, 0.7\}$ on 2,000 chunks. Dashed: no-injection baseline. Lines join recorded values. These controls do not establish a general limitation of target-informed training.

next token to construct the bridge state. To examine this design choice, we introduce controls that additionally inject ground-truth target information into the bridge during training. At 10K updates, the four top-10 target-injection runs yield 56.45–1160.37 PPL when evaluated without injection, compared with 32.41 for training without injection (Fig. 3 (d)). These results favor the prediction-based construction in the tested setting, although the controls also change candidate selection and do not constitute a direct comparison with diffusion models. Separately, Brownian-like embedding noise, disabled during evaluation, yields higher PPL than the better deterministic arm at each of seven recorded checkpoints. Supplementary Secs. C.5 and D.3 detail the constructions and training protocols.

5 Conclusion and Discussion

Continuous Language Bridge reassesses tentative next-token predictions through an observed-to-predicted embedding bridge, a shared second pass, and a categorical mixture, preserving exact autoregressive likelihood. After 10B training tokens on LM1B, CLB reduces test perplexity by 1.27% relative to AR and 0.47% relative to the tied-layer Loop baseline. Across seven external datasets, the default OWT-trained CLB models achieve mean relative perplexity reductions of 8.79%, 5.79%, and 13.76% over their AR counterparts at the 130M, nominal 700M, and 2.1B MoE scales, respectively. On OWT itself, the larger dense and MoE models reduce perplexity by 1.95% at 70K dense updates and 1.72% at 60K MoE-stage updates, respectively. These results support within-token predictive feedback across dense and sparse architectures.

Discussion. The current study focuses on predictive quality in English under the protocols of Sec. 4.1. Extending the bridge to multilingual data, longer contexts, and adaptive verification budgets is a natural next step. Lower PPL measures predictive loss rather than token-level or semantic correction; the risk analysis likewise only provides conditional bounds.

AI USE STATEMENT

In this work, we used generative AI tools solely to polish the language of the manuscript, including improving grammar, wording, and readability. These tools were not used for any other research or writing tasks. All AI-assisted edits were reviewed by the authors to ensure accuracy and preserve the intended meaning. The authors take full responsibility for the final content of this work.

ETHICS STATEMENT

Training uses public web text without human-subject experiments. Models may reproduce harmful, biased, or copyrighted content; releases should document data provenance, intended use, and limitations.

REPRODUCIBILITY STATEMENT

Sec. 3 and Appendices A–I document the method, proofs, protocols, numerical records, and trace coverage. Full reproduction additionally requires anonymized code, configurations, data indices, checkpoint metadata, and original logs; their incomplete availability is identified in the appendices.

REFERENCES

Marianne Arriola, Aaron Gokaslan, Justin Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *International Conference on Learning Representations*, volume 2025, pp. 50726–50753, 2025. 3

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021. 3, 7

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 7432–7439, 2020. 8, 40

Samuel Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. In *Proceedings of the 20th SIGNLL conference on computational natural language learning*, pp. 10–21, 2016. 2

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*, 2013. 6

Ting Chen, Ruixiang ZHANG, and Geoffrey Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=3itjR9QxFw. 3

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018. URL https://arxiv.org/abs/1803.05457. 8, 40

Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Yu Wu, et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 1280–1297, 2024. 3

Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Lukasz Kaiser. Universal transformers. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HyzdRiR9Y7. 3

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022. 3

Jonas Geiping, Sean McLeish, Neel Jain, John Kirchenbauer, Siddharth Singh, Brian Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, and Tom Goldstein. Scaling up test-time compute with latent reasoning: A recurrent depth approach. *Advances in Neural Information Processing Systems*, 38: 41340–41391, 2026. 3

Aaron Gokaslan and Vanya Cohen. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus, 2019. 6

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024. 3

Jaehyeong Jo and Sung Ju Hwang. Continuous diffusion model for language modeling. *Advances in Neural Information Processing Systems*, 38:97394–97430, 2026. 2

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 1

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, et al. Training language models to self-correct via reinforcement learning. In *International Conference on Learning Representations*, volume 2025, pp. 54523–54549, 2025. 1

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in neural information processing systems*, 35: 4328–4343, 2022. 2

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024. 3, 7

Cheng Lu and Yang Song. Simplifying, stabilizing and scaling continuous-time consistency models. In *International Conference on Learning Representations*, volume 2025, pp. 50611–50649, 2025. 3

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *Advances in neural information processing systems*, 36:46534–46594, 2023. 1

Oshri Naparstek. Projected autoregression: Autoregressive language generation in continuous state space. *arXiv preprint arXiv:2601.04854*, 2026. 2

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022. 1

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 2, 6, 8, 19, 20, 24, 25, 28, 35, 38

Subham S Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024. 2, 3, 7, 8, 29, 33, 36, 39

Subham Sekhar Sahoo, Zhihan Yang, Yash Akhauri, Johnna Liu, Deepansha Singh, Zhoujun Cheng, Zhengzhong Liu, Eric P. Xing, John Thickstun, and Arash Vahdat. Esoteric language models: A family of any-order diffusion LLMs. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=4C5lHAIWiF. 3

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8732–8740, 2020. doi: 10.1609/aaai.v34i05.6399. URL https://ojs.aaai.org/index.php/AAAI/article/view/6399. 40

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021. 8

Chenze Shao, Darren Li, Fandong Meng, and Jie Zhou. Continuous autoregressive language models. *arXiv preprint arXiv:2510.27688*, 2025. 2

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652, 2023. 1

Yang Song and Prafulla Dhariwal. Improved techniques for training consistency models. In *International Conference on Learning Representations*, volume 2024, pp. 15078–15097, 2024. 3

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023. 3

Robin Strudel, Corentin Tallec, Florent Altché, Yilun Du, Yaroslav Ganin, Arthur Mensch, Will Grathwohl, Nikolay Savinov, Sander Dieleman, Laurent Sifre, et al. Self-conditioned embedding diffusion for text generation. *arXiv preprint arXiv:2211.04236*, 2022. 3

Jihoon Tack, Jack Lanchantin, Jane Dwivedi-Yu, Andrew Cohen, Ilia Kulikov, Janice Lan, Shibo Hao, Yuandong Tian, Jason E Weston, and Xian Li. Llm pretraining with continuous concepts. In *International Conference on Learning Representations*, volume 2026, pp. 54214–54231, 2026. 3

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022. 1

Zhilin Yang, Zihang Dai, Ruslan Salakhutdinov, and William W. Cohen. Breaking the softmax bottleneck: A high-rank RNN language model. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=HkwZSG-CZ. 3, 24

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/. 8, 40

Zhen Zhang, Xuehai He, Weixiang Yan, Ao Shen, Chenyang Zhao, and Xin Eric Wang. Soft thinking: Unlocking the reasoning potential of LLMs in continuous concept space. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=ByQdHPGKgU. 3

SUPPLEMENTARY MATERIAL

Contents

A Verification-risk analysis and causality 14
A.1 Proof of the conditional mixture-risk bound 14
A.2 Causal likelihood and endpoint approximation 16

B Implementation and model variants 16
B.1 Coordinate encoding and clean-KV inputs 16
B.2 Parameter accounting and the low-dimensional variant 16
B.3 MoE implementation and routing policies 17

C Learning curves and full OWT evaluation records 18
C.1 Dense 130M endpoints and recorded comparisons 18
C.2 Nominal 700M dense models and resume accounting 19
C.3 Historical multiple-choice readout comparisons 20
C.4 Repeated-computation controls 21
C.5 Coordinate and Brownian-noise ablations 21

D Historical bridge ablation audits 22
D.1 Historical pass budget 22
D.2 Path shape and explicit time injection 22
D.3 Perturbing the bridge and conditioning on the target 23

E Supplementary campaigns 24
E.1 The 2.1B MoE scale and routing ablations 24
E.2 Masked diffusion likelihood-bound record 27
E.3 LM1B protocol, learning curves, and final evaluation 27

F Additional coordinate scans 30

G Training dynamics and trace coverage 31
G.1 LM1B training measurements 31
G.2 OWT training traces and coverage 32
G.3 Target-conditioned training traces 32

H Final-checkpoint cross-corpus evaluation 33
H.1 Data construction and scoring protocol 33
H.2 Categorical readouts and transfer limitations 34
H.3 Masked-diffusion bounds 36
H.4 Final coordinate scans and within-checkpoint readout 36
H.5 Within-checkpoint and independent-baseline gains 37

I LM1B sentence boundaries and position interpolation 38
I.1 Sentence-isolated vs. packed evaluation 38
I.2 Untrained extension of learned position embeddings 38
I.3 Precision diagnostic 39

J Multiple-choice evaluation and generation examples 40
J.1 Answer-selection protocol and traceability 40
J.2 Selected paired continuations 40

A Verification-Risk Analysis and Causality

A.1 Proof of Theorem 1

Let $\mathbf{C}$ denote the observed context token vector and Y the true next token, drawn jointly from a fixed evaluation distribution. Hold the first-pass and verification predictors $\mathbf{P}(\mathbf{C}), \widetilde{\mathbf{P}}(\mathbf{C}) \in \Delta^{V-1}$ fixed. The gate receives the measurable first-pass hidden vector $\mathbf{H} = \mathcal{F}_\theta(\mathbf{C}) \in \mathbb{R}^d$ at the last observed position. The scalar entries $P_Y(\mathbf{C})$ and $\widetilde{P}_Y(\mathbf{C})$ are the probabilities assigned to Y. Both predictors assign strictly positive probability to every token almost surely. All expectations below use this evaluation distribution, with

$$\mathbb{E}[-\log P_Y(\mathbf{C})] + \mathbb{E}[-\log \widetilde{P}_Y(\mathbf{C})] < \infty, \qquad \mathbb{E}[U^2] < \infty, \qquad U = \frac{\widetilde{P}_Y(\mathbf{C})}{P_Y(\mathbf{C})} - 1 > -1. \tag{9}$$

For $b \in [0,1]$, let $\mathcal{A}_b$ denote measurable maps $a : \mathbb{R}^d \to [0,b]$, identified up to equality almost surely under the law of $\mathbf{H}$. Thus $a(\mathbf{H})$ is $\sigma(\mathbf{H})$-measurable and cannot depend on the unobserved target. Define

$$\mathcal{R}(a) = \mathbb{E}\Big[-\log\big((1-a(\mathbf{H}))P_Y(\mathbf{C}) + a(\mathbf{H})\widetilde{P}_Y(\mathbf{C})\big)\Big]. \tag{10}$$

This risk is finite for every $a \in \mathcal{A}_1$, since the mixture probability is at least $\min\{P_Y(\mathbf{C}), \widetilde{P}_Y(\mathbf{C})\}$. The conditional mean $D(\mathbf{H}) = \mathbb{E}[U \mid \mathbf{H}]$ and second moment $S(\mathbf{H}) = \mathbb{E}[U^2 \mid \mathbf{H}]$ have measurable versions. In expectations and bounds, we suppress the argument $\mathbf{H}$ when writing a, D, and S. Then $D^2 \le S < \infty$ almost surely, $\mathbb{E}|D| < \infty$, and $S = 0$ implies $D = 0$.

Proof of Theorem 1. Factoring out the positive first-pass probability gives the exact identity

$$\mathcal{R}(a) - \mathcal{R}(0) = -\mathbb{E}[\log(1 + a(\mathbf{H})U)]. \tag{11}$$

For fixed $u > -1$, put $f_u(s) = -\log(1+su)$. If $0 \le s \le 1/2$, then $1 + su \ge 1/2$ and $f_u''(s) = u^2/(1+su)^2 \le 4u^2$. Taylor's formula with an integral remainder yields, for $0 \le a \le 1/2$,

$$f_u(a) = -au + \int_0^a \frac{(a-s)u^2}{(1+su)^2}\, ds \le -au + 4u^2 \int_0^a (a-s)\, ds = -au + 2a^2u^2. \tag{12}$$

Because $a(\mathbf{H})$ is $\sigma(\mathbf{H})$-measurable, conditioning on $\mathbf{H}$ and then applying the tower property gives

$$\begin{aligned} \mathbb{E}[f_U(a(\mathbf{H})) \mid \mathbf{H}] &\le -a(\mathbf{H})D(\mathbf{H}) + 2a(\mathbf{H})^2 S(\mathbf{H}), \\ \mathcal{R}(a) - \mathcal{R}(0) &\le -\mathbb{E}[aD] + 2\mathbb{E}[a^2 S]. \end{aligned} \tag{13}$$

All terms are integrable by Eq. 9 and the boundedness of a. This proves Eq. 6.

Define the measurable comparison gate

$$a^\star(\mathbf{H}) = \begin{cases} \min\{1/2, D_+(\mathbf{H})/(4S(\mathbf{H}))\}, & S(\mathbf{H}) > 0, \\ 0, & S(\mathbf{H}) = 0, \end{cases} \qquad D_+(\mathbf{H}) = \max\{D(\mathbf{H}), 0\}. \tag{14}$$

On $S = 0$, both D and $a^\star$ vanish. On $S > 0$, the definition gives $a^\star D = a^\star D_+$ and $2(a^\star)^2 S \le a^\star D_+/2$. Hence

$$\mathcal{R}(a^\star) - \mathcal{R}(0) \le -\tfrac{1}{2}\mathbb{E}[a^\star D_+] = -G, \qquad G := \tfrac{1}{2}\mathbb{E}[a^\star D_+] \ge 0. \tag{15}$$

The gain margin G is finite because $a^\star \le 1/2$ and $\mathbb{E}|D| < \infty$. Moreover, $G > 0$ if and only if $\Pr(D(\mathbf{H}) > 0) > 0$, since $D > 0$ entails $0 < S < \infty$ almost surely and therefore $a^\star D_+ > 0$.

Finally, for a learned gate $\widehat{a} \in \mathcal{A}_1$ satisfying the theorem's comparison assumption,

$$\mathcal{R}(\widehat{a}) \le \inf_{a \in \mathcal{A}_{1/2}} \mathcal{R}(a) + \varepsilon \le \mathcal{R}(a^\star) + \varepsilon \le \mathcal{R}(0) - G + \varepsilon. \tag{16}$$

Here ε is the gate-comparison error, whereas G is the available gain margin. No minimizing gate need exist: the explicit feasible $a^\star$ is sufficient. Only the comparison class is capped at $1/2$; the learned gate can take values throughout $[0,1]$. □

Lemma 1 (A useful verification direction)**.** *Under Theorem 1's assumptions, use a regular conditional law of U given $\mathbf{H}$ to define $g_{\mathbf{H}}(a) = \mathbb{E}[-\log(1 + aU) \mid \mathbf{H}]$ for scalar $a \in [0, 1]$. For almost every $\mathbf{H}$, some $a > 0$ has $g_{\mathbf{H}}(a) < 0$ if and only if $D(\mathbf{H}) > 0$. If $D(\mathbf{H}) \le 0$, zero is a minimizer of the conditional risk over $[0, 1]$.*

Proof. For $0 < a \le 1/2$, the mean value theorem gives $|\log(1 + aU)|/a \le 2|U|$. Conditional dominated convergence therefore gives the right derivative at zero. Convexity of $-\log$ also supplies the supporting-line inequality:

$$g'_{\mathbf{H}}(0+) = -\mathbb{E}[U \mid \mathbf{H}] = -D(\mathbf{H}), \qquad g_{\mathbf{H}}(a) \ge -aD(\mathbf{H}) \quad (0 \le a \le 1). \tag{17}$$

The conditional first moment is finite almost surely. If $D(\mathbf{H}) > 0$, the negative derivative implies $g_{\mathbf{H}}(a) < 0$ for sufficiently small positive a. If $D(\mathbf{H}) \le 0$, the second inequality gives $g_{\mathbf{H}}(a) \ge 0 = g_{\mathbf{H}}(0)$ for every $a \in [0, 1]$. □

This criterion concerns expected log loss conditional on the gate's information, not correctness of an individual token. Identical predictors give $U = D = S = G = 0$ and no strict gain. If verification reduces the probability of the realized target almost surely, every strictly positive gate instead increases loss.

Corollary 1 (Successive prediction updates)**.** *Let $\mathbf{P}^{[0]}$ be a fixed initial predictor. At each stage $k = 1, \dots, m$, let $\mathcal{F}_k$ be a fixed measurable map from the observed context to the gate's hidden representation, and define*

$$\mathbf{P}^{[k]}(\mathbf{C}) = (1 - \widehat{a}_k(\mathbf{H}_k))\mathbf{P}^{[k-1]}(\mathbf{C}) + \widehat{a}_k(\mathbf{H}_k)\widetilde{\mathbf{P}}^{[k]}(\mathbf{C}), \qquad \mathbf{H}_k = \mathcal{F}_k(\mathbf{C}). \tag{18}$$

At each stage, suppose the fixed predictor pair $(\mathbf{P}^{[k-1]}, \widetilde{\mathbf{P}}^{[k]})$ satisfies Theorem 1's assumptions on the same evaluation distribution, with comparison error ε_k for its gate. Define $U_k = \widetilde{P}_Y^{[k]}(\mathbf{C})/P_Y^{[k-1]}(\mathbf{C}) - 1$, $D_k = \mathbb{E}[U_k \mid \mathbf{H}_k]$, $S_k = \mathbb{E}[U_k^2 \mid \mathbf{H}_k]$, and $G_k = \frac{1}{2}\mathbb{E}[a_k^\star(D_k)_+]$, where $a_k^\star$ is Eq. 14 for (D_k, S_k). Writing $\mathcal{R}(\mathbf{Q}) = \mathbb{E}[-\log Q_Y(\mathbf{C})]$, we have

$$\mathcal{R}(\mathbf{P}^{[m]}) \le \mathcal{R}(\mathbf{P}^{[0]}) - \sum_{k=1}^{m} G_k + \sum_{k=1}^{m} \varepsilon_k. \tag{19}$$

Proof. The theorem applies to the current predictor and its verification candidate separately at each stage. Subtract consecutive risks and sum:

$$\begin{aligned} \mathcal{R}(\mathbf{P}^{[m]}) - \mathcal{R}(\mathbf{P}^{[0]}) &= \sum_{k=1}^{m} \left[\mathcal{R}(\mathbf{P}^{[k]}) - \mathcal{R}(\mathbf{P}^{[k-1]})\right] \\ &\le \sum_{k=1}^{m} (-G_k + \varepsilon_k). \end{aligned} \tag{20}$$

□

Square brackets index complete prediction updates; the implemented CLB corresponds to one such update. Each stage improves strictly when $G_k > \varepsilon_k$; nonincreasing risk follows when $G_k \ge \varepsilon_k$. These statements concern expected log loss and, by exponentiation, population PPL on the same evaluation distribution. Token-level accuracy requires a separate analysis.

Connection to the implemented gate. CLB takes $\mathbf{H} = \mathbf{H}_i$, $\mathbf{P} = \mathbf{P}_i$, $\widetilde{\mathbf{P}} = \widetilde{\mathbf{P}}_i$, and $\widehat{a}(\mathbf{H}_i) = \Pi_{i,2}$. The two-way softmax gate uses the first-pass hidden state; its affine parameters are trained jointly with the backbone through the mixture likelihood, without a separate target for the mixing weight. The comparison error accounts for this restricted gate class and its optimization and generalization relative to all measurable gates in the reference class. Applying the bound requires the moment assumptions and the gate-risk comparison in Theorem 1. The conditional moments describe the evaluation distribution; they are not inputs to the forward gate, and $a^\star$ is a theoretical comparison gate rather than the implemented rule. Neither these moments nor the comparison error is estimated in the reported experiments. The analysis fixes both predictors after training, not during optimization. Because the argument uses only the predictive distributions and their mixture weights, it also applies to other feedback predictors.

A.2 Causal likelihood and endpoint approximation

The clean causal mask makes $\mathbf{H}_i$ and $\mathbf{P}_i$ functions of $x_{\le i}$. Their top-K endpoint and bridge are prefix-derived. At every layer, clean cache entries at $j < i$ depend only on $x_{<i}$, while the verification state combines its own prefix-derived input with those entries. Induction makes $\widetilde{\mathbf{P}}_i$ prefix-derived. The gate shares that dependence, and its nonnegative weights sum to one; hence $\widehat{\mathbf{P}}_i$ is a normalized conditional and its product gives Eq. 5. The argument holds at each fixed evaluation coordinate and draft size.

We next bound the embedding-space approximation error introduced by top-K truncation. Here and in the supplementary experiments, T denotes the interpolation coordinate written t in Eq. 2. Fix the prefix $x_{<i}$ and suppress this argument in the bridge evaluations below. Let $\mathbf{D}_i^\star = \sum_v P_{i,v}\mathbf{E}_v$, $\delta_{i,K} = 1 - \sum_{v\in S_{i,K}} P_{i,v}$, and $\mathcal{B}_{\theta,t}^\star = t\mathbf{E}_{x_i} + (1-t)\mathbf{D}_i^\star$. If all embedding norms are at most R, decomposing the full mean into retained and omitted mass gives

$$\|\mathbf{D}_i^{(K)} - \mathbf{D}_i^\star\|_2 \le 2R\delta_{i,K}, \qquad \|\mathcal{B}_{\theta,t}^{(K)} - \mathcal{B}_{\theta,t}^\star\|_2 \le 2R(1-t)\delta_{i,K}.$$

For nonzero omitted mass, the first difference is $\delta_{i,K}(\mathbf{D}_i^{(K)} - \mathbf{D}_{\mathrm{tail}})$, and both normalized means have norm at most R; zero omitted mass is immediate. The bound is stated in the bridge's embedding space.

B Implementation and Model Variants

B.1 Coordinate encoding and clean-KV inputs

The implementation uses a two-layer SiLU map $\mathcal{T}(\cdot) : \mathbb{R}^{256} \to \mathbb{R}^d$ over Fourier features, with widths $256, d, d$ and an explicit zero-conditioning region:

$$\begin{aligned} \mathcal{C}(t) &= 1_{\{|t|\ge 10^{-2}\}}\,\mathcal{T}([\cos(t\boldsymbol{\Omega}); \sin(t\boldsymbol{\Omega})]) \in \mathbb{R}^d, \\ \omega_r &= \exp[-\log(10^4)(r-1)/128]. \end{aligned} \tag{21}$$

Here $\boldsymbol{\Omega} = (\omega_1, \ldots, \omega_{128})^\top \in \mathbb{R}^{128}$ collects the frequencies, with the cosine block preceding the sine block. Each Transformer block maps $\mathcal{C}(t)$ through an AdaLN MLP of widths $d, 128, 4d$ to four vectors in $\mathbb{R}^d$: a shift/scale pair for each of its two RMSNorm layers. A separate map of widths $d, 128, 2d$ conditions the final normalization. The first pass uses the zero conditioning vector with observed token embeddings. Setting the bridge coordinate to zero instead gives the soft predicted endpoint as its input; it does not recover the first-pass computation. Zero input to the conditioning MLPs also need not annul their biases.

For the OWT high-dimensional predictive endpoint, a top-50 buffer is formed and one K is sampled uniformly from $\{5, \ldots, 50\}$ per training forward call. Evaluation uses $K = 10$. The entire predictive endpoint is detached before continuous verification, including its probability weights and embedding lookup; the observed anchor, clean cache, and first-pass-conditioned gate retain their gradient paths. The clean-KV mask allows cached first-pass states at $j < i$ and the position's own second-pass state, excluding the current position's clean entry and every other position's verification state.

For $y \in \mathcal{V}$, the normalized mixture, it has the equivalent log-domain expression

$$\log \widehat{P}_{i,y} = \mathrm{logsumexp}\Big(\log \Pi_{i,1} + \log P_{i,y},\ \log \Pi_{i,2} + \log \widetilde{P}_{i,y}\Big). \tag{22}$$

For numerical protection, the evaluated mixture readouts floor the target probability at 10^{-12}, equivalently clipping its log probability at $\log(10^{-12})$. The frequency of activation and the resulting difference from the unclipped log loss were not measured. If clipping occurs, the reported mixture log loss is smaller than the unclipped value. This safeguard affects numerical scoring, not the normalization or autoregressive factorization of the underlying model.

B.2 Parameter accounting and the low-dimensional variant

Tied-embedding bridge. The default construction, called CLB-high in the experiments, uses the tied token space for both endpoints, with $d = 768$ or 1,280 at the two dense scales. No separate

bridge embedding or lift is required. The 130M OWT AR/Loop backbone has 133,757,313 unique parameters; CLB-high has 133,758,852. At width 768, the gate adds 2×768 weights and two biases. One legacy scalar, inactive in the clean-KV forward path, accounts for the remaining parameter. The difference is 1,539 parameters, approximately 0.0012%. The nominal 700M-scale tied-embedding model has approximately 812.3M parameters. Parameter sharing does not eliminate the second pass's operations; this accounting concerns unique parameters, not FLOPs or readout structure.

Low-dimensional bridge. CLB-low uses an untied embedding matrix $\mathbf{E}_{\mathrm{low}} \in \mathbb{R}^{128\times V}$, with column $\mathbf{E}_{\mathrm{low},v}$ for token v, eight-group normalization $\mathcal{N}(\cdot) : \mathbb{R}^{128} \to \mathbb{R}^{128}$, and a three-layer lift $\mathcal{U}(\cdot) : \mathbb{R}^{128} \to \mathbb{R}^{d}$. Its predictive endpoint and observed anchor are

$$\mathbf{D}_{i,\mathrm{low}} = \mathcal{N}\left(\sum_{v\in\mathcal{V}} \bar{P}_{i,v}\mathbf{E}_{\mathrm{low},v}\right), \qquad \mathbf{X}_{i,\mathrm{low}} = \mathcal{N}(\mathbf{E}_{\mathrm{low},x_i}),$$

both in $\mathbb{R}^{128}$. The verification input is $\mathcal{U}((1-T)\mathbf{D}_{i,\mathrm{low}} + T\mathbf{X}_{i,\mathrm{low}}) \in \mathbb{R}^{d}$. Each normalization group contains 16 coordinates; the lift has widths $128, d, d, d$ with GELU after each of its first two linear layers. Normalization follows the probability-weighted sum, rather than preceding it for each candidate. The tied output head and probability mixture are unchanged. The expectation identity and truncation bound for the unnormalized tied-embedding endpoint do not transfer automatically through this nonlinear normalization. The smaller and larger low-dimensional models have approximately 141.5M and 822.2M parameters, respectively, about 7.7M and 9.9M above their tied-embedding counterparts. This variant changes capacity and feedback geometry, not embedding dimension alone.

B.3 MoE IMPLEMENTATION AND ROUTING POLICIES

Backbone and expert dimensions. The audited fixed-ID model has 36 layers, width $d = 1{,}280$, 20 attention heads, and 2,110,776,744 registered unique parameters. Its embedding/head, clean-KV attention, coordinate conditioning, and categorical mixture follow the dense construction. Only the feed-forward sublayers are replaced by MoE modules. Each expert $\mathcal{S}_s(\cdot)$ or $\mathcal{G}_j(\cdot)$ maps $\mathbb{R}^d \to \mathbb{R}^d$, implemented as two bias-free linear maps of shape $d \times d$ with GELU between them. Its intermediate width is d, whereas the corresponding dense FFN has intermediate width $4d$. The reported 2.1B count is total parameters, not measured active parameters per token.

Packed storage and effective topology. The implementation stores fifteen experts in one array. In zero-based code indices, experts 11 to 14 supply the four always-active shared contributions, and experts 0 to 10 are the eleven routed candidates in the clean and fixed-ID verification passes. Thus stored configuration values of zero shared experts and top-eight routing describe the array format, not the executed fixed-ID topology. Each pass evaluates the same four shared experts and four selected routed experts. The shared outputs are summed without division by four; only the selected routed weights are softmax-normalized. The Transformer residual addition is outside the MoE expression in Eq. 8.

Coordinate-aware router. Suppressing layer and token indices, let $\mathbf{Z}, \widetilde{\mathbf{Z}} \in \mathbb{R}^d$ be the conditioned, normalized clean and verification FFN inputs. The eleven routed-candidate logits are $\mathbf{A} = \mathbf{W}_{\mathrm{r}}[\mathbf{Z}; \mathcal{Q}(0)]$ and $\widetilde{\mathbf{A}} = \mathbf{W}_{\mathrm{r}}[\widetilde{\mathbf{Z}}; \mathcal{Q}(T)]$, with $\mathbf{W}_{\mathrm{r}} \in \mathbb{R}^{11\times(d+64)}$. For the first eleven selection-bias entries $\mathbf{B}$, the fixed-ID policy uses

$$\mathcal{R} = \mathrm{Top4}_{j\in\{1,\dots,11\}}(A_j + B_j), \qquad w_j = \frac{\exp A_j}{\sum_{k\in\mathcal{R}} \exp A_k}, \qquad \widetilde{w}_j = \frac{\exp \widetilde{A}_j}{\sum_{k\in\mathcal{R}} \exp \widetilde{A}_k}. \tag{23}$$

For the fixed identities $j \in \mathcal{R}$, each pass normalizes its own current router logits to obtain the combination weights. The selection bias is used in choosing the identities.

For each layer, the packed router has $\mathbf{W}_{\mathrm{r}}^{\mathrm{pack}} \in \mathbb{R}^{15\times(d+d_T)}$. The first eleven rows form $\mathbf{W}_{\mathrm{r}}$ defined above. With $d_T = 64$, its separate coordinate projection $\mathcal{Q}(\cdot)$ is

$$\begin{aligned} \mathcal{Q}(T) &= \mathbf{W}_T[\sin(1000T\boldsymbol{\nu}); \cos(1000T\boldsymbol{\nu})] + \mathbf{B}_T \in \mathbb{R}^{64}, \\ \nu_r &= \exp[-\log(10^4)(r-1)/32], \qquad r = 1, \dots, 32, \end{aligned} \tag{24}$$

where $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_{32})^\top \in \mathbb{R}^{32}$, $\mathbf{W}_T \in \mathbb{R}^{64\times 64}$, and $\mathbf{B}_T \in \mathbb{R}^{64}$. The bias-free packed router takes $[\mathbf{Z}; \mathcal{Q}(0)]$ in the clean pass and $[\widetilde{\mathbf{Z}}; \mathcal{Q}(T)]$ in verification. The router uses sine/cosine features with a coordinate scale of 1,000 and an unmasked affine projection. At $T = 0$, its input includes the cosine features, and the projection includes a learned bias. Setting $d_T = 0$ removes this router coordinate branch while preserving the bridge and the backbone's coordinate conditioning. The routing policy is selected by pass identity, including for verification at coordinate zero.

Three routing policies. All three bridge policies begin with four shared experts plus four of eleven routed candidates. In shared/reselected verification, the second pass retains the shared sum but makes a new top-four selection from the eleven candidates using its own router logits and the selection bias. In no-shared verification, the second pass instead selects eight of all fifteen packed experts, normalizes their current logits over those eight, and applies the routed-output scale γ without an additional shared sum. In fixed-ID verification, the selected clean-pass integer identities are retained for the second pass. Their combination weights and expert outputs are recomputed, not cached or frozen. Removing the shared sum also changes the output's weighting structure; it is not solely a change in expert diversity. Router dimensions and initialization differ across some recorded controls, as documented in Supplementary Sec. E.1.

Load balancing and three distinct gates. The balancing bias affects top-k selection only; it never enters the routed combination softmax. After each optimizer update, globally aggregated selection counts $\mathbf{C}_{\text{load}} \in \mathbb{R}^{15}$ update the packed bias according to

$$\begin{aligned} \mathbf{B}^{\text{pack}} &\leftarrow \mathbf{B}^{\text{pack}} + \eta_{\text{b}} \operatorname{sign}\big(\bar{C}_{\text{load}}\mathbf{1} - \mathbf{C}_{\text{load}}\big)\,, \\ \bar{C}_{\text{load}} &= \frac{1}{15}\sum_{j=1}^{15} C_{\text{load},j}, \qquad \eta_{\text{b}} = 10^{-3}. \end{aligned} \tag{25}$$

Here the mean covers the packed array, and the selection bias $\mathbf{B}$ consists of its first eleven entries. The bias is updated directly from counts of selected routed identities. In the fixed-ID branch, these counts exclude the always-active shared evaluations. The normalized w_j and $\widetilde{w}_j$ combine selected expert outputs, and the unconstrained per-layer scalar $\gamma \in \mathbb{R}$, shared across passes, scales their sum. At the vocabulary level, $\mathbf{\Pi}_i \in \Delta^1$ mixes the two final token distributions using the first-pass hidden state. Retained expert IDs are detached integers; router weights, expert parameters, and current expert outputs retain their gradient paths.

Initialization and interpretation. The fixed-ID module initializes γ to one and the selection-bias buffer to zero; a resumed checkpoint restores their trained values. The bridge MoE is upcycled from its trained dense tied-embedding bridge after 70K effective dense updates, while independent MoE-AR is upcycled from its own dense AR initialization. MoE-stage update counts start after dense pretraining, and the two runs inherit different initializations and computational costs. The historical shared/reselected and no-shared runs evaluate the alternative routing policies described above. Supplementary Sec. E.1 gives their training and routing configurations. Across these policies, the prefix-derived bridge and normalized likelihood are preserved.

C Learning Curves and OpenWebText Results

This appendix reports LM1B test learning curves and OWT validation results, including dense-model comparisons and bridge ablations. Each plotted series follows one training run. The subsections specify the evaluation protocol and update counting for each comparison.

C.1 Dense 130M Models

Table S1 compares the dense 130M endpoints. The nearly parameter-matched CLB-high model reduces PPL from 21.64 to 21.28; CLB-low reaches 21.16 but increases the parameter count to 141.5M. Both CLB variants use two passes, compared with one for AR. Because the AR optimizer budget is not independently verified and CLB-low changes model capacity, these endpoints do not isolate the bridge mechanism at matched training and inference compute.

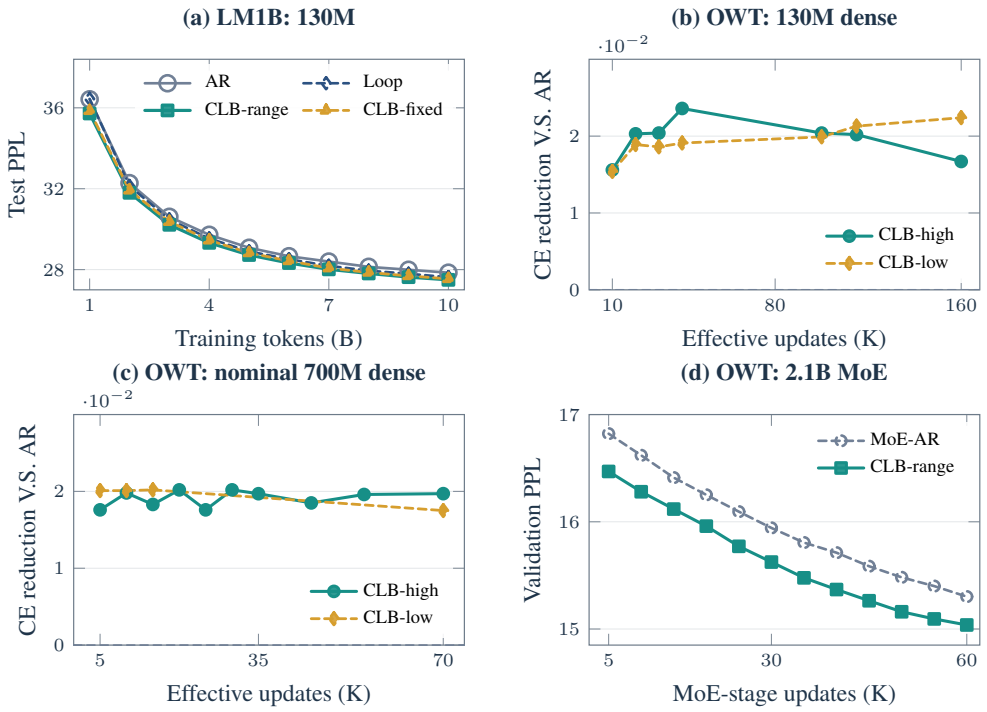


Figure S1: Learning curves from recorded evaluations. **(a)** All four categorical LM1B arms have test measurements at each 1B-token milestone through 10B; both CLB curves consistently use the prespecified inference coordinate $T = 0.6$. **(b,c)** Positive historical same-update CE reductions favor CLB; the recovered 130M AR resume counter does not independently certify its cumulative budget. Nominal 700M denotes 812.3M total parameters for AR/high. **(d)** MoE comparisons stop at 60K stage updates, with validation-selected bridge coordinates and different dense initializations. Table 2 separately reports the 60K result at $T = 0.6$. Lines guide the eye between observations, not independent replications.

Table S1: Historical 130M OpenWebText endpoint comparison, labelled 160K in the original ledger. The AR cumulative optimizer budget is not independently verified from its recovered resume metadata. AR cites the causal-LM family (Radford et al., 2019); CLB variants are ours. Seed, effective batch, and 2K-chunk validation size are shared; identical resumed data order is not established.

Model	Parameters	Passes	CE ↓	PPL ↓	ΔCE V.S. AR
AR	133.757M	1	3.0746	21.64	–
CLB-high (near matched)	133.759M	2	3.0579	21.28	−0.0167
CLB-low (capacity changing)	141.5M	2	3.0522	21.16	−0.0224

C.2 Nominal 700M dense models and resume accounting

The historical runs named *700M* use 36 layers, hidden width 1,280, and 20 attention heads. Their logged parameter totals are approximately 812.3M for AR and CLB-high, and 822.2M for CLB-low. Thus 700M is a nominal run-scale label, not an exact total or an audited non-embedding count. The low-dimensional variant adds approximately 9.9M parameters at this scale. The recorded protocol uses OpenWebText, GPT-2 tokenization, length-1,024 evaluation sequences, effective batch 128, and seed 1337. Fig. S2 plots the recovered CE records at their actual cumulative update counts; Table S2 gives the common 70K endpoint.

Cumulative steps count only accepted checkpoint histories. CLB-high resumes at 35K and completes another 35K; its final internal `ckpt_35000.pt` therefore represents 70K total updates. CLB-low's

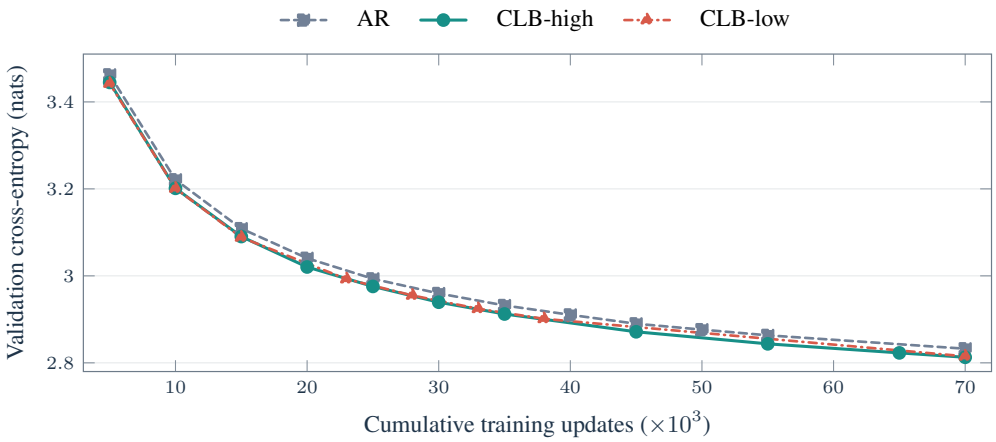


Figure S2: Recorded likelihood curves at the nominal 700M dense scale. Markers denote available n=2,000 evaluations; connecting lines are visual guides, not measurements at missing checkpoints. CLB-low retains its off-grid cumulative steps after resumption. The series combine archived experiment records with recovered evaluation outputs; the 70K entries are supported by original evaluation logs. These are historical training trajectories, with the hardware and resumption differences described in the text.

Table S2: Common cumulative-70K dense endpoint on 2,000 OWT validation sequences. AR cites its model family; CLB variants are ours. The bridge records evaluate $T = 0.6$; they do not document a new full coordinate scan. Parameter totals and bridge metrics retain their logged precision.

Model	Parameters	CE	PPL	PPL reduction vs. AR
AR (Radford et al., 2019)	812.3M	2.832506	16.9880	—
CLB-high	812.3M	2.8128	16.657	1.95%
CLB-low	822.2M	2.8150	16.694	1.73%

accepted contributions are $17K + 1K + 30K + 3K + 19K = 70K$, ending at internal step 19K. Discarded tails are excluded.

The intermediate low-dimensional segment has an 18K offset, placing its internal 5/10/15/20K evaluations at cumulative 23/28/33/38K. These points are not relabelled as matched AR checkpoints.

The high-dimensional run and AR used different training machines and software stacks: the initial high-dimensional run used Pro 6000 Blackwell GPUs with PyTorch 2.7.1, whereas AR used H100 GPUs with PyTorch 2.6.0. CLB-low used the H100 system, but changed GPU count and resumed multiple times. Its nominal effective batch was preserved by adjusting gradient accumulation; this does not establish identical minibatch order throughout the resumed histories. Restarts also introduced renewed warmup. These records therefore support a scale-specific likelihood comparison, not a fully isolated architectural or equal-compute effect.

The older AR-70K value of 2.8030 nats belongs to a training-time summary and is excluded from the common-harness series. The recovered n=2,000 evaluation reports 2.8325, and a same-code reevaluation reports 2.832506 with PPL 16.9880. Original bridge outputs report CLB-high 2.8128/16.657 and CLB-low 2.8150/16.694 at their cumulative 70K checkpoints. Earlier points have mixed provenance: some are recovered evaluation outputs, while others are preserved in contemporaneous experiment summaries or the archived report. No missing coordinate or checkpoint is represented as an additional observation.

C.3 Historical multiple-choice readout comparisons

The archived experiment report also records zero-shot multiple-choice scoring on five tasks. Each comparison uses the first-pass head and the probability-mixture readout of the *same* bridge checkpoint.

CLB-high is evaluated at cumulative 55K on 200 examples per task; CLB-low is evaluated at 38K on collections described in the report as the full validation sets. The two columns therefore have different training and evaluation budgets and should not be compared as a controlled high/low ablation. The original MC result files and per-example predictions have not been recovered here, so Table S3 preserves the report's rounded values as provenance-qualified historical evidence.

Table S3: Historical within-checkpoint readout comparisons for our CLB variants, in percent. Each cell is first-pass head → bridge mixture. Values are character-length-normalized accuracy except WinoGrande, which uses ordinary accuracy. These are neither comparisons against independently trained AR models nor tests of statistical significance.

Task	CLB-high, 55K (200/task)	CLB-low, 38K (full sets)
HellaSwag	$40.0 \rightarrow 44.0$	$38.9 \rightarrow 39.0$
ARC-Easy	$28.5 \rightarrow 29.0$	$36.0 \rightarrow 37.9$
ARC-Challenge	$23.5 \rightarrow 24.0$	$22.4 \rightarrow 23.1$
OpenBookQA	$26.0 \rightarrow 28.0$	$28.4 \rightarrow 28.6$
WinoGrande	$48.5 \rightarrow 51.0$	$50.2 \rightarrow 50.6$

These comparisons describe changes in answer-option scoring, not token-level error correction. The small high-dimensional sample, rounded reporting, unequal checkpoint budgets, and unrecovered per-example outputs preclude claims of significance or general reasoning improvement. They complement the likelihood record without establishing that continuous bridge states are faithful explanations of a model's reasoning.

C.4 Repeated-computation controls

The 70K Loop and bridge-range checkpoints were evaluated on 2,000 contiguous OpenWebText validation chunks, sequence length 1,024, with BF16. The bridge inference time is frozen at $t = 0.6$ for the extended evaluation after the original validation scan. The extended 4,330-chunk evaluation comprises all nonoverlapping chunks available in the validation binary. Its additional 2,330-chunk suffix yields PPL 22.70 for Loop and 22.92 for bridge-range, preserving the full-set ordering. At 65K, the additional suffix yields 22.89 (Loop), 23.03 (fixed), and 23.09 (range), again preserving the ordering.

Table S4: Exact dense-control ledger for our CLB and tied-layer Loop variants (Appendix C.4). These are historical bridge checkpoints paired with the newly trained Loop control, not independent bridge retrains.

Bridge setting	Checkpoint	Chunks	Loop	Historical CLB	Δ Loop
Range, selected on 2K	70K	2,000	22.3825	22.6076	−0.996%
Range, frozen from 2K	70K	4,330	22.5529	22.7734	−0.968%
Fixed $t = 0.6$, frozen from 2K	65K	4,330	22.7367	22.8625	−0.550%
Range, frozen from 2K	65K	4,330	22.7367	22.9382	−0.878%

C.5 Coordinate and Brownian-noise ablations

For the Brownian study, all three arms use the same 130M architecture, seed, effective batch, OpenWebText validation chunks, and 11-point inference scan. Noise is disabled at inference. The Brownian RNG is independent of the model, time, and data RNGs and is checkpointed separately. Each arm is nevertheless a single training realization, so the seven checkpoints are not seven independent replications.

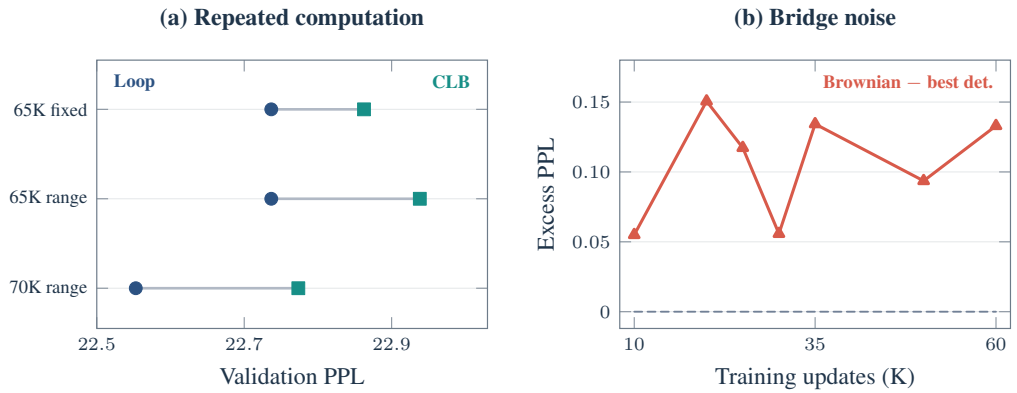


Figure S3: Controls for the full bridge architecture. **(a)** Each row pairs Loop with a historical CLB checkpoint on the same 4,330 validation chunks; lower PPL is better. **(b)** Brownian excess PPL relative to the better of range and fixed training at each recorded update. All seven margins are positive. Connected checkpoints come from single training runs and do not constitute independent replications or confidence intervals.

Table S5: Exact 130M ledger for our coordinate and Brownian-noise ablations (Appendix C.5). Each entry is the best validation PPL over an 11-point inference-T scan; positive margins favor the best deterministic comparator.

Checkpoint	Range	Fixed T=0.6	Brownian	Brownian margin
10K	32.4080	32.4270	32.4630	+0.170%
20K	27.1018	27.0072	27.1578	+0.558%
25K	25.9466	25.8751	25.9924	+0.453%
30K	25.2031	25.1319	25.1877	+0.222%
35K	24.5609	24.4832	24.6175	+0.549%
50K	23.4064	23.3399	23.4335	+0.401%
60K	22.9732	22.8658	22.9988	+0.582%

D Historical Bridge Ablation Audits

These studies concern earlier causal bridge implementations. Their role is to record attempted design choices and exclusions; they are not matched ablation estimates for the final clean-KV model.

D.1 Historical pass budget

An earlier causal bridge checkpoint was evaluated at pass-count settings of one, two, three, four, and six. Its available values are approximately 22.59, 19.98, 20.06, 20.03, and 20.05 PPL, respectively. Fig. S4 keeps this schedule ablation distinct from the later clean-KV experiments: it has different checkpoint provenance and is a single historical evaluation rather than a final matched control.

D.2 Path shape and explicit time injection

The historical path audit records variants labelled k2, k4, and k6. The k2 and k4 entries were invalidated by training–evaluation mismatch and target dependence, and are excluded from improvement claims. The recovered summary describes k6 as approximately neutral, but does not provide sufficient original evidence to reconstruct a matched final-implementation effect or a full numerical trajectory.

The explicit time-injection audit considered adaptive normalization, query/output low-rank adapters, and a learned time feature. No stable benefit was recorded in those historical runs. This is not a controlled removal of $\mathcal{C}(T)$ from the final clean-KV architecture, which still uses the encoding in Appendix B.1. Neither audit establishes that time conditioning or path shape is irrelevant in general.

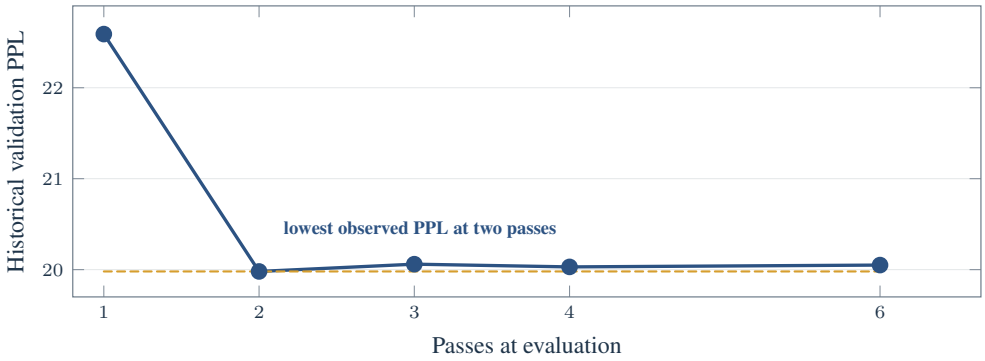


Figure S4: Historical pass-budget diagnostic. Lines connect the recorded evaluation schedules of one earlier causal-bridge checkpoint as visual guides. Its implementation differs from the final clean-KV model, so absolute PPL is not directly comparable to Table S1.

D.3 Perturbing the bridge and conditioning on the target

Noise changes the state, not the training coordinate alone. The noise control preserves the predictive mean $\mathbf{M}_i(T) = T\mathbf{E}_{x_i} + (1-T)\mathbf{D}_i^{(K)} \in \mathbb{R}^d$ and perturbs the verification input during training:

$$\mathbf{X}_i(t) = \mathbf{M}_i(t) + 2\sigma_{\mathcal{B}}\sqrt{t(1-t)}\,\boldsymbol{\varepsilon}_i, \qquad \boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d), \qquad \sigma_{\mathcal{B}} = 0.5\ \mathrm{RMS}(\mathbf{M}). \tag{26}$$

Here $\mathrm{RMS}(\mathbf{M})$ is one detached scalar over the interpolated batch. The peak noise standard deviation is half the signal RMS. Independent Gaussian draws implement a marginal variance envelope, not a correlated Brownian trajectory or an SDE solver. Scoring in Sec. C.5 disables the noise. At 60K, range/noise/fixed training gives 22.9732/22.9988/22.8658 PPL. At 30K, noise slightly improves on range training (25.1877 vs. 25.2031), but not fixed training (25.1319). Thus these seven single-run checkpoints show no consistent benefit; they are not independent replications, and training hardware/software histories differ. A separate truncated-Gaussian distribution over t changes coordinate sampling, not embedding noise. Its completed 68,936-update checkpoint yields 22.5612 OWT PPL at prespecified $t = 0.6$; Sec. H reports this final readout and all seven external collections separately from the historical learning curves.

Ground-truth injection: evaluate after removing the answer. Four historical 130M runs inject the true next token during training. Let $\mathbf{R}_i \in \Delta^9$ be the first prediction restricted to its top-10 candidates. The target occupies slot j_y; if absent, it replaces the weakest candidate while inheriting that slot's original mass. The training input uses

$$\widetilde{\mathbf{R}}_i = (1-\gamma t)\mathbf{R}_i + \gamma t\,\mathbf{e}_{j_y}, \qquad \mathbf{X}_i^{\mathrm{GT}}(t) = t\mathbf{E}_{x_i} + (1-t)\sum_{j=1}^{10}\widetilde{R}_{i,j}\mathbf{E}_{v_j}, \tag{27}$$

where $\mathbf{e}_{j_y}$ is a slot indicator and $v_{j_y} = y_i$. The explicit added target coefficient in the input is $\gamma t(1-t)$, not a constant γ; replacing a candidate also changes the input. At 10K, original no-GT evaluation logs give gated PPL 56.446, 112.743, 380.755, and 1160.366 for $\gamma = .05, .10, .25, .50$, vs. 32.411 without injection (Fig. 3). All use the first 2,000 SL1024 OWT validation chunks, BF16 and batch eight. Injection is explicitly disabled; each GT run selects its best $t \in \{.4, .6, .7\}$ on these same chunks. The baseline additionally scans .8, without changing its minimum at .6. These are historical single-run controls: loading was non-strict, some load diagnostics were filtered from the saved logs, and the unmodified training draft samples K whereas the injection branch fixes $K = 10$. Consequently the comparison does not isolate target information alone. All four gated outputs are also worse than their own first pass (Table S6). This is consistent with a harmful training–evaluation conditioning mismatch, not a proof that every target-informed training scheme fails. Internal GT-on scores are excluded because they reveal the answer.

A distinct mean-blend variant and excluded histories. An earlier control sets $\widetilde{\mathbf{D}}_i = (1-a)\mathbf{D}_i + a(\mathbf{E}_{y_i} + \sum_{j=1}^{7}\mathbf{E}_{v_j})/8$, $a = 0.1t$, before interpolation with $\mathbf{E}_{x_i}$. The sum is not deduplicated when

Table S6: No-GT readouts of the four target-injected 10K checkpoints. All values are PPL on 2,000 OWT chunks. Head diagnostics and the best gated scan can use different coordinates, shown separately; the first pass is independent of the verification coordinate.

Training γ	Head t	First pass	Second pass	Best gated	Gate t
0.05	0.6	36.266	339.776	56.446	0.7
0.10	0.6	37.809	490.251	112.743	0.4
0.25	0.4	40.169	871.435	380.755	0.4
0.50	0.4	39.001	2089.736	1160.366	0.4

the target is already among the top seven. Its historical 10K summary gives no-GT CE 3.4802 (PPL approximately 32.4662), essentially unchanged from the baseline CE 3.4785; the original scoring output has not been recovered. We therefore do not mix it into the original-log curve. Direct target blends have no recovered formal no-GT result. Separate curved-target and bidirectional token-prefix trials were invalidated or mismatched. These records are implementation cautions, not additional valid performance claims.

Historical mixture-readout control. A recovered 10K log evaluates a plain two-head mixture-of-softmax adaptation (Yang et al., 2018) on 2,000 OWT chunks and reports CE 3.5856 / PPL 36.076, vs. independent AR's 3.4938 / 32.911. This larger evaluation does not reproduce the earlier small-batch improvement claim. It concerns a historical readout design, not a gate-only ablation of the final two-pass CLB, so it does not resolve the latter's component contributions.

E Supplementary Campaigns

The following sections detail the 2.1B MoE routing experiments and the LM1B test results, keeping their initialization, evaluation, and training-budget differences explicit. MDLM likelihood bounds are separated from categorical perplexity throughout.

E.1 The 2.1B MoE scale and routing ablations

The 2.1B configuration is a mixture-of-experts model with 2,110,776,744 registered unique parameters, not a dense model or a count of parameters activated per token. Its backbone has 36 layers, width 1,280, and 20 attention heads. The fixed-expert variant uses four always-active shared experts and selects four of eleven routed experts per token in each layer. Verification reuses the first pass's selected expert identities while recomputing their combination weights. The checkpoint packs all fifteen experts into a routed array; its stored zero-shared/top-eight configuration therefore does not describe the effective forward topology. The router uses 64-dimensional coordinate features.

Table S7: OWT expert-combination ablations at 5K MoE-stage updates (2,000 validation chunks). AR follows Radford et al. (2019); CLB variants are ours. Topology describes continuous verification: shared experts are always active, routed counts are selected / available, and coordinate dimension counts the router's added time features.

Variant	Shared	Routed	Selection	Coord. dim.	PPL ↓
Independent AR	–	–	–	–	16.8216
Shared	4	4 of 11	Reselect	0	16.4624
Shared	4	4 of 11	Reselect	64	16.4720
Shared	4	4 of 11	Reselect	256	16.4770
No shared sum	0	8 of 15	Reselect	64	16.6293
Fixed expert IDs	4	4 of 11	Reuse	64	16.4693

Initialization and comparison protocol. The bridge is upcycled from the dense high-dimensional bridge after 70K effective dense updates; the independent MoE-AR baseline is upcycled from its

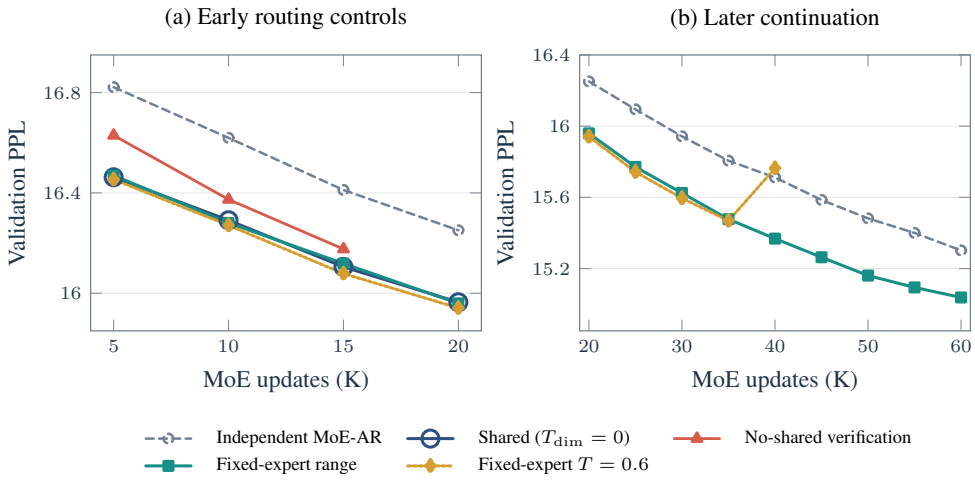


Figure S5: Recorded MoE comparisons. **(a)** Early routing controls retain the shared and no-shared variants; the 5K configurations are detailed in Table S7. **(b)** AR and range-trained fixed-expert CLB are shown through 60K, matching Table S8. The separate fixed-training-$T = 0.6$ curve ends at its recorded 40K checkpoint. Panels use different vertical scales. Missing observations are not filled, and connected checkpoints are not independent replications.

own dense AR initialization. The update labels below count the subsequent MoE stage, not training from scratch. Hardware changes and historical data-order differences further prevent a strict single-variable comparison. Bridge predictions use two shared-backbone passes and a probability-space gate, whereas the independent AR baseline uses one pass. Equal MoE update labels thus do not establish equal training or inference FLOPs. The formal evaluations use 2,000 OWT validation chunks of length 1,024; bridge coordinates are scanned on this collection. These are single-run comparisons, with no claim of statistical significance.

Table S8: OWT validation PPL through 60K MoE-stage updates. AR follows Radford et al. (2019); CLB uses fixed expert IDs and range training. These historical CLB readouts select the inference coordinate on validation: the 60K entry uses $T = 0.5$, whereas Table 2 uses the prespecified $T = 0.6$. Early routing variants are shown separately in Fig. S5(a) and Table S7.

MoE update	AR PPL	CLB PPL
5K	16.8216	16.4693
10K	16.6195	16.2802
15K	16.4117	16.1185
20K	16.2510	15.9595
25K	16.0943	15.7711
30K	15.9426	15.6244
35K	15.8061	15.4772
40K	15.7116	15.3682
45K	15.5855	15.2636
50K	15.4818	15.1608
55K	15.4002	15.0941
60K	15.3021	15.0376

Routing controls. The shared variant retains four shared experts and reselects four routed experts at verification. Its early trajectory in Fig. S5(a) uses $T_{\text{dim}} = 0$ and ends at 20K; this disables router coordinate features rather than interpolation. The no-shared trajectory ends at 15K. These shorter trajectories are retained in the figure rather than padded with missing entries in Table S8. Separate shared runs with $T_{\text{dim}} = 64$ and 256 give 16.4720 and 16.4770 PPL at 5K, compared with

16.4624 for the zero-dimensional router. The no-shared-verification variant preserves the first pass's shared experts but selects eight of all fifteen experts at verification, without a separate always-active shared sum. It uses $T_{\mathrm{dim}} = 64$, as does fixed-expert range training. These routing and conditioning differences must not be collapsed into a single causal effect of expert reuse.

Range continuation and checkpoint provenance. The range continuation restores its original 35K checkpoint, including optimizer and random-number-generator states, at update 35001 without renewed warmup. It does not continue the separate fixed-T run or add another 35K to the labels. A retest gives 15.4771 PPL at $T = 0.5$, vs. the historical 15.4772 retained in the ledger. The displayed 40K–60K range results select inference $T = 0.5$. At the 60K reporting endpoint, AR reaches 15.3021 PPL and CLB reaches 15.0392 at the prespecified $T = 0.6$, a 1.72% reduction. The validation-selected $T = 0.5$ readout is 15.0376, corresponding to a 1.73% reduction, and is kept distinct from the headline result. The original 65K/70K ledger entries remain in the accompanying data; the separate 70K cross-corpus campaign in Sec. H must not be relabelled as a 60K evaluation.

Fixed-coordinate training and observed deterioration. A separate fixed-expert run trains at $T = 0.6$. Its recovered $T = 0.6$ evaluations at 5K, 10K, 15K, 20K, 25K, 30K, 35K, and 40K give PPL 16.4535, 16.2707, 16.0785, 15.9411, 15.7430, 15.5954, 15.4688, and 15.7634. The last value is worse than AR's 15.7116 at 40K; a full coordinate scan gives a nearly identical best value of 15.7633 at $T = 1$. In a diagnostic using the same four samples at $T = 0.6$, second-pass CE increases from 4.7935 to 9.4722 between 35K and 40K, second-pass accuracy falls from 23.729% to 7.111%, and mean gate weight on that pass falls from 8.718% to 0.0683%. These diagnostic samples are not the full validation collection. The run was stopped, and the older range checkpoint was restored for the separate continuation above. The triggering cause remains unresolved; this observation does not establish that fixed-coordinate training is intrinsically unstable.

E.1.1 Verification schedules and routing diagnostics

We also evaluate additional verification passes without retraining the shared/reselected-expert checkpoints. A schedule of m coordinates executes $1 + m$ backbone passes: the initial prediction followed by m verification passes. Each new pass rebuilds its predictive endpoint from the preceding pass's distribution and reuses the clean context. Last-gate evaluation combines the first and final predictions with the trained two-way gate. Multi-gate evaluation preserves the first-pass weight and distributes the existing verification weight uniformly across all m verification distributions; it adds no learned parameters. These are different readouts, not interchangeable names for the same inference procedure.

Table S9: Post-training schedules on 2,000 OWT validation chunks of length 1,024. Panels use different shared/reselected-expert checkpoints and are not a matched comparison across router dimensions. Total passes include the initial prediction. Multi-gate results were recorded only for panel (a); panel (b) therefore reports last-gate PPL alone.

(a) Router dimension 0, checkpoint 10K

Verification coordinates	Total passes	Last-gate PPL	Multi-gate PPL
[0.6]	2	16.2910	16.2910
[0.8, 0.6]	3	16.3703	16.2943
[1.0, 0.8, 0.6]	4	16.3269	16.2952
[0.9, 0.7, 0.5]	4	16.3307	16.2894
[1.0, 0.9, 0.8, 0.6]	5	16.3355	16.2973

(b) Router dimension 64, checkpoint 5K

Verification coordinates	Total passes	Last-gate PPL
[0.6]	2	16.4720
[0.8, 0.7, 0.6]	4	16.4913
[0.9, 0.75, 0.6]	4	16.5030
[1.0, 0.95, 0.9, 0.8]	5	16.5026
[1.0, 0.9, 0.8, 0.7]	5	16.5041

At 10K, the first-pass PPL is 16.7287. The two-pass mixture reaches 16.2910, while longer last-gate schedules worsen it. Multi-gate retains intermediate predictions and is closer to the two-pass result; its lowest recorded value, 16.2894, is only 0.0016 lower. The separate 5K checkpoint has first-pass PPL 16.9144, and none of its recorded longer schedules improves on 16.4720. These finite schedule tests show neither monotonic improvement with extra passes nor a statistically established benefit from the best schedule. They are single-checkpoint validation comparisons, not independent training repetitions; exploratory 512-chunk results are excluded from the table.

Coordinate-dependent expert selection. A separate 256-chunk diagnostic at the router-dimension-64, 5K checkpoint compares expert selections at $T = 0.6$ with those at $T = 0$. Top-one agreement is 0.3449, exact top-four agreement is 0.1082, and set Jaccard similarity is 0.4399. Top-one agreement is higher in the first layer (0.7155) than in the remaining layers (0.3343). These statistics describe changes in routed identities; they do not measure prediction accuracy or establish that more route variation improves likelihood. The smaller diagnostic collection is not pooled with the 2,000-chunk performance evaluations.

E.2 Masked diffusion likelihood-bound record

The 130.6M OWT MDLM run was evaluated every 5K steps through 70K on the same 2,000 chunks. Its final conditional $\exp(\mathrm{NELBO})$ is 43.5631 for raw weights and 38.5484 for EMA, over 1,023 targets after an observed first token. The complete 5K–70K metric ledger is retained separately because it is not an AR-PPL trajectory or a matched bridge ablation.

E.3 LM1B protocol, learning curves, and final evaluation

Data and matched training. The LM1B campaign uses the original benchmark archive with GPT-2 tokenization and wrapped 128-token blocks, following the GPT-2 alternative in the MDLM codebase rather than its primary BERT-tokenized setup. The 99 training shards contain 30,301,028 sentences; heldout shards 00–09 provide validation and 10–49 provide test data. Official detokenization, sentence EOS markers, and shard-local packing produce 6,984,055 training, 14,180 validation, and 56,567 test blocks. Each block has BOS/EOS and 126 payload tokens. Incomplete packing tails are discarded consistently. This is a matched-family experiment, not a reproduction of published BERT-tokenized LM1B scores.

All five arms are trained from fresh initialization, with identical shared named backbone tensors under seed 1337. Global batch 512 comprises two GPUs, microbatch 64, and four accumulation steps, giving 65,536 input tokens per optimizer update. An update-index-derived sampler supplies exactly the same global batch to every arm at each completed-update index. Repeated model passes do not count as additional input tokens. Each arm completed 152,588 updates, or 10,000,007,168 input tokens; the excess over 10B results from rounding to full batches. Training samples blocks with replacement, so 10B denotes cumulative exposure rather than unique corpus tokens: the packed training file contains 893,959,040 tokens. Stages are scheduled in approximately 1B-token increments, with full-state checkpoints and permanent stage snapshots.

Architecture and optimization. The backbone has 12 layers, width 768, 12 heads, learned positional embeddings, tied input/output embeddings, and zero dropout. AR and adjacent tied-layer Loop each have 130,708,352 parameters; CLB-range and CLB-fixed each have 130,709,890, an increase of 1,538. MDLM has 129,920,384 parameters and uses bidirectional attention with a mask-only SUBS objective. Loop applies each layer twice consecutively. The bridge arms use the clean-KV, high-dimensional construction and a first-pass auxiliary loss of weight $\lambda = 1$. This campaign disables logit-z regularization ($\beta = 0$). All arms use AdamW with learning rate 3×10^{-4}, betas $(0.9, 0.999)$, zero weight decay, gradient clipping at 1, and 2,500 linear-warmup updates followed by constant learning rate. Forward/backward arithmetic is BF16 with FP32 parameters and optimizer state; MDLM additionally maintains EMA weights with decay 0.9999.

The draft helper samples one K uniformly from $\{5, \ldots, 50\}$ per training forward call, shared across that call's batch and token positions; evaluation fixes $K = 10$. Softmax renormalizes the retained top-K logits before the weighted embedding is constructed. This draft branch is computed without gradients, while the observed-token endpoint and shared backbone remain trainable. Range training

draws $T \sim U(0,1)$ separately at each token position; fixed training uses $T = 0.6$ everywhere. These settings match the high-dimensional draft and coordinate conventions used in the OWT construction.

Disjoint validation and test evaluation. AR, Loop, and CLB report teacher-forced final-output CE/PPL on the same first 16,000 test blocks, comprising 2,032,000 scored next-token targets. These categorical readouts are distinct from the Monte Carlo MDLM metric below; the mixture's numerical safeguard is described in Supplementary Sec. B.1. At each bridge checkpoint, the 11 coordinates $\{0, 0.1, \ldots, 1\}$ are evaluated on the first 2,000 validation blocks; the lowest validation CE determines the coordinate used for the separate test evaluation. The prespecified $T = 0.6$ test result is also retained when selection chooses another coordinate. Test blocks are disjoint from the blocks used for coordinate selection. The range selections at 1B, 2B, and 3B are 0.5, 0.5, and 0.6; the fixed-training model selects 0.6 at all three stages. The final 10B readouts use the prespecified $T = 0.6$, which also minimizes validation CE for both CLB arms. Table S10 preserves validation-selected historical values; the complete 1B–10B categorical learning curves instead hold inference $T = 0.6$ fixed, as indicated in their caption.

Table S10: LM1B exact budget ledger and test perplexity at early milestones and completion. AR follows causal language modeling (Radford et al., 2019); Loop is our tied-layer control and CLB variants are ours. Each row shares a token budget and test blocks. Bridge coordinates are selected on validation; the final selected coordinate is also the prespecified $T = 0.6$.

Updates	Input tokens	AR	Loop	CLB-range	CLB-fixed
15,259	1,000,013,824	36.4210	36.4420	35.7209	35.8545
30,518	2,000,027,648	32.2757	32.1328	31.7985	31.9224
45,777	3,000,041,472	30.6110	30.4780	30.2161	30.3689
152,588	10,000,007,168	27.8477	27.6247	27.4940	27.5433

The historical 9B coordinate-selection comparison. At 137,330 updates (9,000,058,880 input tokens), the AR and CLB-range checkpoints provide the same-budget comparison in Table S11. CLB-range reduces test PPL by approximately 1.30% relative to AR. Its inference coordinate is selected as $T = 0.5$ on validation only. The separately recorded prespecified-$T = 0.6$ test evaluation gives CE 3.318804 and PPL 27.6273; it does not replace the validation-selected result merely because its test value is slightly lower. This comparison documents coordinate selection at an intermediate checkpoint. The fixed-$T = 0.6$ learning curves also include Loop and CLB-fixed at 9B; final 10B readouts are available for all five arms.

Table S11: Intermediate LM1B comparison at 137,330 updates / 9,000,058,880 input tokens. Both models use the same 16,000 test blocks; CLB's coordinate is selected on disjoint validation data. The final endpoint is 10B.

Model	Test CE ↓	Test PPL ↓	Selected T
AR (Radford et al., 2019)	3.331929	27.9923	–
CLB-range (ours)	3.318829	27.6280	0.5

Coordinate sensitivity across checkpoints. Fig. 3(a) compares complete validation scans at 3B and 9B. At the common 45,777-update checkpoint, range training produces a comparatively flat curve over the nonzero scanned coordinates, while fixed training has a more pronounced optimum near its training coordinate. Both select $T = 0.6$, but range achieves lower test PPL at this stage (Table S10). The range model's 137,330-update scan selects $T = 0.5$, with validation PPL 27.4861 vs. 27.4863 at $T = 0.6$. Thus both checkpoints favor interior coordinates, without establishing a unique universal choice. The special time-feature mask at $T = 0$ means that these samples do not establish continuity of the full network across the masking threshold. The scans are coordinate-sensitivity diagnostics, not evidence that inference traverses the bridge or that prediction quality improves monotonically along it. Fixed-training $T = 0.1$ and $T = 0.9$ arms are not part of this LM1B campaign; those inference coordinates in a scan must not be counted as additional training ablations.

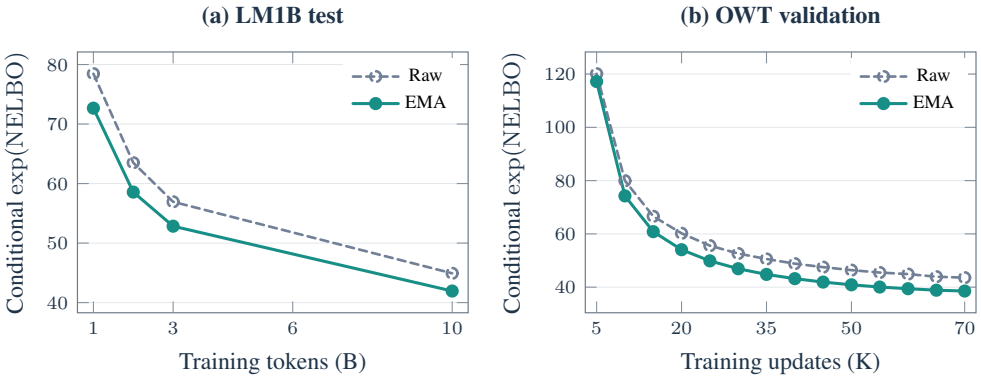


Figure S6: Held-out masked-diffusion likelihood-bound trajectories, separate from exact categorical PPL and training loss. Both panels use the conditional-prefix-one, eight-draw estimator with their corpus-specific protocols. LM1B shows recovered 1B–3B and final 10B values; no intermediate bound measurements are inferred. Raw and EMA weights remain separate; lines connect recorded checkpoints as visual guides.

Separate masked-diffusion metric. MDLM is evaluated on the same 16,000 test blocks using eight stratified Monte Carlo draws per block. Table S12 reports the conditional-prefix-one variant, which leaves the first token observed and scores 127 targets per block. Its $\exp(\text{NELBO})$ is a Monte Carlo likelihood-bound estimate, not exact autoregressive PPL. Raw and EMA weights are kept separate. At 3B, raw/EMA NELBO values are 4.041935/3.967243, with rough chunk-level standard errors 0.003896/0.003866; these errors are not document-aware. The final 10B conditional values are 44.9205/41.9462. The separately retained joint 128-target variant gives raw/EMA $\exp(\text{NELBO})$ of 55.1676/51.2275 at 3B and 43.6049/40.7401 at 10B; neither is substituted for the conditional metric.

Table S12: LM1B conditional likelihood-bound estimates for our MDLM adaptation (Sahoo et al., 2024). Budgets and updates match Table S10, but this metric must not be read as exact categorical-model PPL.

Training tokens (approx.)	Raw $\exp(\text{NELBO})$	EMA $\exp(\text{NELBO})$
1B	78.4596	72.6690
2B	63.5028	58.5706
3B	56.9364	52.8386
10B	44.9205	41.9462

Final 10B test results. All five arms completed 152,588 updates and have final 10B test measurements. Re-evaluating the four categorical models on the original 16,000 test blocks reproduced their recorded CE and PPL values. CLB-range's final 27.4940 PPL is 1.27% below AR and 0.47% below Loop. These are observations from single-seed runs, not independent replications or direct measurements of error correction.

Sampled training-set evaluation. The final checkpoints were additionally evaluated on 16,000 training blocks sampled uniformly without replacement using seed 20260920. Identical sample-index and token hashes across all five arms verify that they use the same sampled training collection, distinct from the heldout test collection; this is not an evaluation of the full training corpus. AR, Loop, CLB-range, and CLB-fixed obtain training PPL 24.4386, 24.3273, 24.2745, and 24.3139, respectively, with 2,032,000 scored targets per model. Both CLB models use $T = 0.6$ and the final probability mixture, not the auxiliary-loss sum or an average over random training coordinates. Conditional-prefix-one MDLM raw/EMA training $\exp(\text{NELBO})$ is 44.4792/41.5538, again a separate metric. In this sample, the test-minus-training CE gap is 0.130587 for AR and 0.124542 for CLB-range; the smaller gap does not establish a causal generalization mechanism or statistical significance. Fig. S8 separately

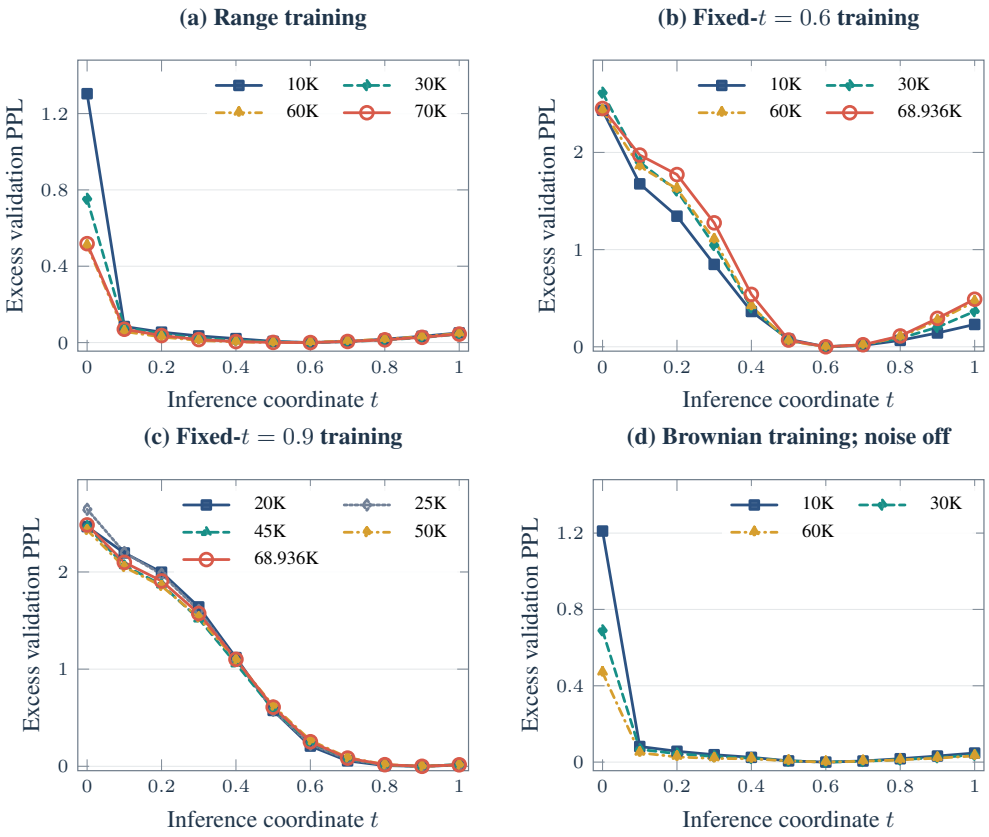


Figure S7: OWT coordinate sensitivity at multiple training stages, using existing n=2,000 evaluations rather than new experiments. Each curve subtracts that checkpoint's own minimum PPL over the 11 scanned coordinates; this shows sensitivity, not absolute model quality. Panels use different vertical ranges. The 10K/30K/60K records come from the matched range, fixed-0.6, and Brownian evaluation files; range-70K comes from the Loop-control evaluation, and fixed-0.6/0.9 final records and fixed-0.9 intermediate stages come from their original completed scans. Lines connect discrete observations; $t = 0$ additionally activates the time-feature mask. Checkpoints from one training run are not independent seeds, and the figure neither demonstrates a continuous network nor a trajectory followed at inference.

displays sparse online training snapshots with their objective-specific labels, not these post-training sampled evaluations or a complete per-update trace.

F Additional Coordinate Scans

Table S13: Training-coordinate ablation of our CLB variants (Sec. 3). OWT fixed-coordinate arms share 68,936 updates; range has 70,000 and is not strictly budget-matched. OWT selects and reports on validation. LM1B reports held-out test PPL after 3B training tokens, with coordinates selected on validation; no audited fixed-training $0.1/0.9$ arms are available there.

Training T	OWT updates	OWT selected T	OWT val. PPL	LM1B test PPL
Fixed 0.1	68,936	0.1	22.6101	–
Fixed 0.6	68,936	0.6	22.4985	30.3689
Fixed 0.9	68,936	0.9	22.5822	–
Range $U(0, 1)$	70,000	0.6	22.6076	30.2161

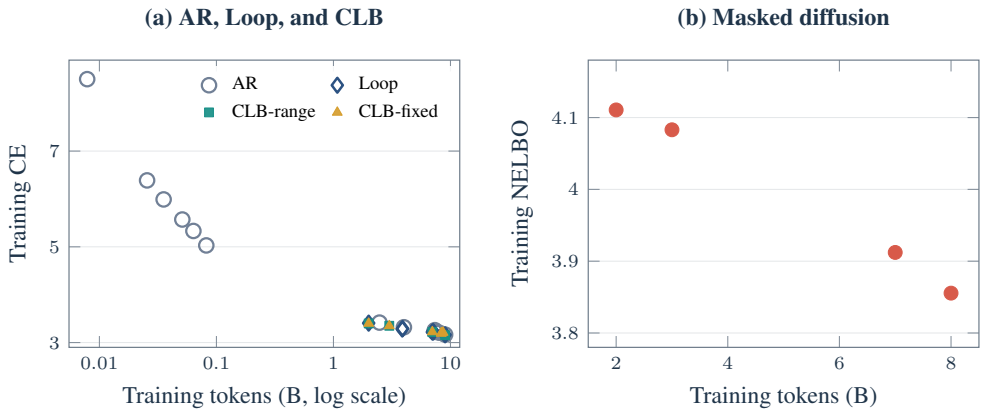


Figure S8: Recovered LM1B training snapshots, not complete training curves. **(a)** Final-output training CE for AR, Loop, CLB-range, and CLB-fixed; the logarithmic token axis includes the early AR observations. **(b)** MDLM's stochastic training NELBO uses a separate metric and axis. Each marker is one observed update's global-batch mean, not a moving average or heldout evaluation. Sparse points are not connected or interpolated. CLB's plotted CE excludes its first-pass auxiliary loss; its total training objective is the sum of both losses.

Fig. 3(a,b) contrasts LM1B range/fixed training at 3B and 9B with the OWT endpoint training-coordinate comparison. The OWT/MoE records contain 32 complete 11-point scans: 31 at the 130M OWT scale and one for the 2.1B MoE at 55K, retaining all 352 observations. These scans evaluate saved checkpoints rather than independently trained models. The training coordinate distribution and inference coordinate are distinct: evaluating a range-trained model at $T = 0.9$ does not constitute fixed-$T = 0.9$ training. Supplementary Sec. H additionally reports a 21-point scan at the final 130M OWT range checkpoint.

Fig. S7 selects 16 scans to show early, intermediate, and late sensitivity within each of four training arms. All score the same-sized 2,000-chunk OWT validation collection. For checkpoint s and scanned grid $\mathcal{T} = \{0, 0.1, \ldots, 1\}$, the plotted quantity is $\Delta\mathrm{PPL}_s(T) = \mathrm{PPL}_s(T) - \min_{u\in\mathcal{T}} \mathrm{PPL}_s(u)$. Each minimum is therefore zero by construction, not evidence of equal performance across checkpoints. The final fixed-coordinate records use 68,936 updates, while the nearby range record uses 70,000; subtracting their minima does not remove this budget difference. Fixed-$T = 0.1$ has a complete final scan but no available multi-stage series, so it appears only in the endpoint training-coordinate comparison.

Range and Brownian mean-path sensitivity is concentrated near the masked $T = 0$ setting in these selected records, while the fixed-coordinate runs retain distinct preferences near their training coordinate. Brownian noise is disabled during the plotted inference scans. These descriptions concern the observed validation grid only: the mask prevents an inference of whole-network continuity, coordinate selection on the reporting collection is not a heldout test, and multiple checkpoints do not provide independent-seed replication. The recovered MoE-55K scan is retained in the numerical ledger, but is not mixed into the dense panels because its architecture, initialization, and training stage differ.

G Training Dynamics and Trace Coverage

G.1 LM1B training measurements

Training objectives and held-out metrics are kept separate throughout this appendix. Fig. S8 shows recorded training measurements for all five LM1B arms. For categorical models the plotted primary metric is final-output next-token CE; the bridge's optimized total additionally includes first-pass CE. MDLM instead has a stochastic NELBO objective and occupies a separate panel. Each marker averages the current global update, not a time window. The measurements are too sparse to establish smooth convergence, so they are not connected.

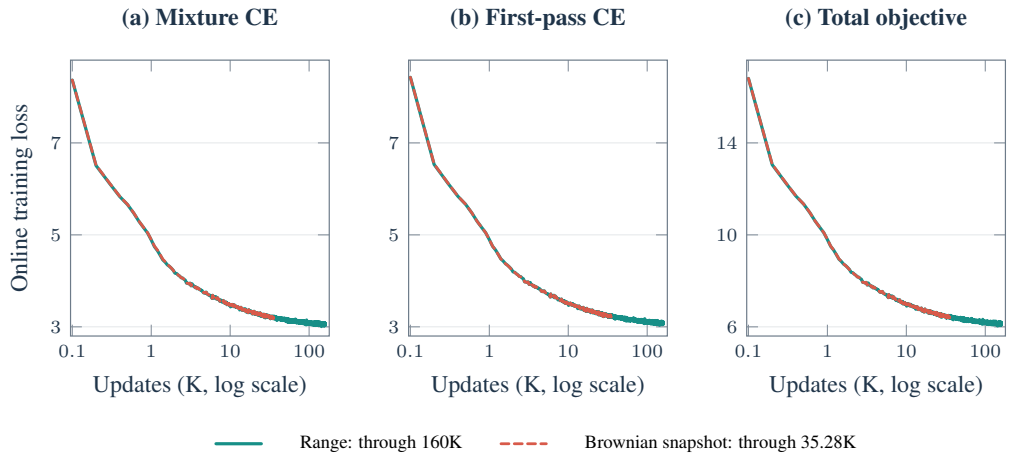


Figure S9: Recovered 130M OWT production training traces. Curves average the logged rank-0 observations in nonoverlapping 100-update windows; the last Brownian window contains only updates 35,210–35,280. Update zero is excluded from the logarithmic axis. Mixture CE, first-pass CE, and their regularized total objective are different metrics, not validation measurements. The Brownian trace ends because the local log snapshot is partial, not because training stopped. Its overlap with range training does not establish equivalent final performance. These are two single-seed traces; missing histories for other arms are not filled by evaluation scores.

G.2 OWT TRAINING TRACES AND COVERAGE

Fig. S9 uses original 130M training logs: range training through 160K and the available Brownian segment through 35,280 updates. Nonoverlapping 100-update windows average only the recorded observations within each window. The first-pass CE, mixture CE, and total objective are distinct quantities. These traces concern the runs represented by the logs, not every fixed-coordinate or larger-scale comparison.

G.3 TARGET-CONDITIONED TRAINING TRACES

Fig. S10 records the six recovered target-blending training traces from Sec. D.3. The four top-10 injection runs, the weak top-8 mean blend, and the short direct blend remain separate designs. Their logged mixture CE uses target information during training; it is not a valid target-free generation score. The first-pass CE is shown separately, and no missing training interval is filled from an evaluation. In particular, a falling target-conditioned objective does not contradict the degraded no-GT readouts in Sec. D.3.

Trace coverage. Complete training traces are unavailable for the remaining OWT AR, Loop, fixed-coordinate, nominal 700M, MoE, and MDLM runs. Their checkpoint evaluations are reported separately and are not treated as training-loss measurements. The LM1B panel likewise contains sparse measurements rather than complete per-update traces. These coverage limits constrain conclusions about training dynamics, independently of the completed evaluations.

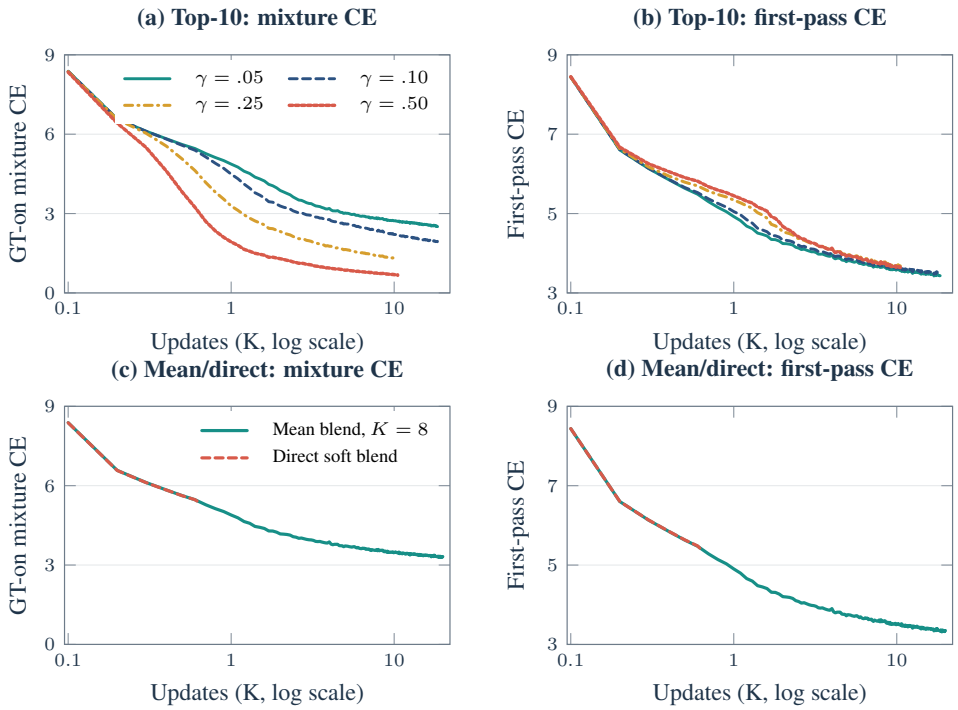


Figure S10: Online training traces for six historical GT-conditioned 130M runs. Curves average printed rank-0 observations in nonoverlapping 100-update windows, retaining each final partial window; update zero is omitted from the log axis. Mixture CE uses a target-conditioned intermediate during training and is not a valid held-out language-model likelihood. The first pass receives no injected target, but shares jointly trained parameters. Lower GT-on mixture loss therefore does not establish better no-GT generation. Colors identify runs within each row; these are different constructions and single training realizations, not matched training-seed replications. Traces stop at the last recovered training record, not necessarily a saved checkpoint.

H Final-Checkpoint Cross-Corpus Evaluation

This archived campaign evaluates selected checkpoints on seven external text collections and an OWT control. All 26 configurations have complete readouts: 21 OWT-trained configurations and five LM1B-trained arms. The 316 measurements comprise 252 categorical readouts, including three 21-point coordinate scans, and 64 masked-diffusion bound estimates. The default MoE pair in this campaign is at 70K stage updates, distinct from the 60K OWT endpoint now reported in Table 2; 60K MoE cross-corpus readouts are not available in this archive. A final checkpoint is the selected endpoint of its configuration, not a common training budget.

H.1 Data construction and scoring protocol

PTB and WikiText-103-raw use validation splits; LM1B, the GPT-2 LAMBADA release, and AG News use test splits; Scientific Papers PubMed and Arxiv use validation splits. We follow the detokenization and separator-free concatenation convention described for MDLM evaluation (Sahoo et al., 2024), using a pinned implementation and GPT-2 BPE. Each external stream contains at most its first 2,048,000 tokens in deterministic source order, with no added BOS, EOS, or record separators. Shorter collections use all available tokens; an incomplete final block is discarded. This capped evaluation is not a reproduction of a full-split literature table. Wiki and AG below abbreviate WikiText-103 and AG News.

The OWT-trained cohort uses context 1,024; the LM1B-trained cohort uses its native context 128. Their scores are not context-matched. All categorical models observe the first token of each block and score the remaining positions. Bridge inference is fixed in advance at $T = 0.6$ on external

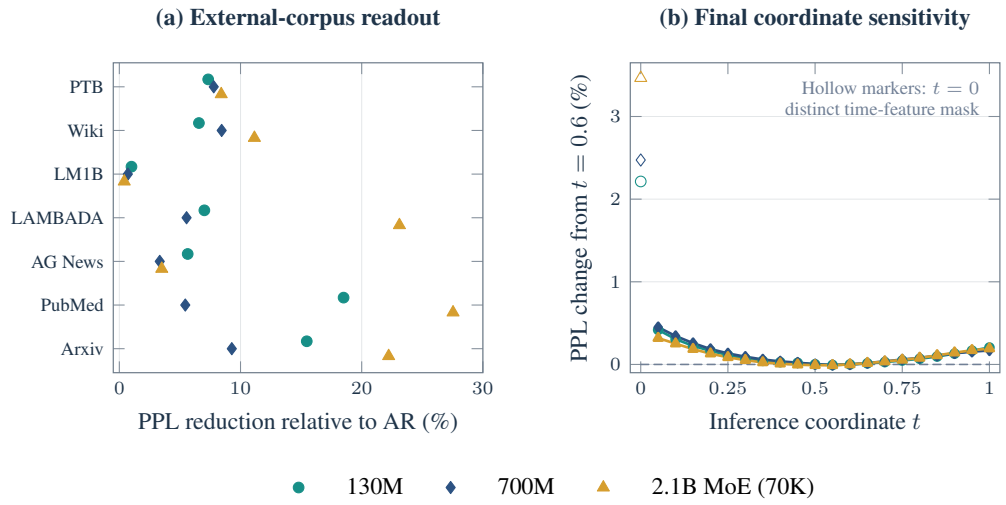


Figure S11: Final OWT-trained checkpoint diagnostics across scales. **(a)** Each marker compares the default range CLB with independent AR of the same scale on one external collection, at the prespecified $t = 0.6$ and context 1,024; categorical datasets are not connected by curves. These are single-run checkpoint comparisons, with the budget and initialization caveats stated in the text. **(b)** The three 21-point OWT scans are normalized by each checkpoint's own $t = 0.6$ PPL. Lines connect recorded positive coordinates, not sequential inference steps. Hollow $t = 0$ markers are disconnected because that branch uses a distinct time-feature mask.

collections, without dataset-specific tuning. PPL is the exponential of total token NLL divided by scored tokens, not an average of block PPLs. Dense and LM1B models use FP32 parameters, while all MoE models use BF16 parameters; every model uses BF16 autocast. The native 10^{-12} bridge probability safeguard is retained. MDLM uses stratified Monte Carlo estimates; raw/EMA weights and prefix-zero/prefix-one protocols remain separate.

Within a cohort and prefix convention, dataset hashes and scored-token counts agree across models. At context 1,024, the counts for PTB, Wiki, AG, and LAMBADA are 77,748, 235,290, 391,809, and 422,499; LM1B, PubMed, Arxiv, and OWT each have 2,046,000 targets. At context 128 the corresponding short-collection counts are 77,724, 234,188, 389,382, and 420,243, with 2,032,000 targets for each capped stream. Prefix-zero MDLM additionally scores the first token of every block.

H.2 Categorical readouts and transfer limitations

The 130M CLB-range and CLB-low checkpoints have documented 160K endpoints. All three fixed-coordinate arms, truncated-normal coordinate training, and Brownian training end at 68,936 updates; Loop ends at 70K. The recovered 130M AR checkpoint stores step 80K and a data-step offset of 80K; that offset alone is not independent evidence of 160K optimizer updates. We preserve the earlier ledger's historical label elsewhere but do not use this survey to certify budget matching. The nominal 700M checkpoints have documented 70K cumulative histories. CLB-low changes representation dimension and parameter count as described in Supplementary Sec. B.2.

Truncated-normal training samples the coordinate from $\mathcal{N}(0.6, 0.1^2)$ conditioned on $[0, 1]$; it does not inject Gaussian noise into the embedding. The Brownian-trained model is evaluated on its deterministic mean path with noise disabled, not by marginalizing a stochastic likelihood. Their OWT PPLs at the common $T = 0.6$ are 22.56 and 22.64, compared with 22.50 for fixed-0.6 training. Rankings differ on external collections: Brownian training gives lower LAMBADA PPL than fixed-0.6 training (56.72 vs. 58.23), but higher Wiki PPL (33.91 vs. 33.23). These results do not establish a uniform benefit from either alternative.

The 70K range-trained fixed-expert bridge and MoE-AR have paired final-checkpoint readouts. The bridge is upcycled from the dense high-dimensional bridge and AR from its own dense initialization; their compute, initialization, and historical data order are not matched. The range bridge yields

Table S14: OWT-trained dense categorical checkpoints, context 1,024. Values are PPL; every CLB row is evaluated at $T = 0.6$, including models trained with a different coordinate distribution. OWT is in-domain; the other columns are external collections. AR follows Radford et al. (2019); Loop and CLB are our implementations. This is a checkpoint survey with unequal training budgets, not a controlled ranking of all variants.

Model / readout	OWT	PTB	Wiki	LM1B	LAMBADA	AG	PubMed	Arxiv
AR, 130M	21.64	109.14	32.72	64.14	59.78	77.31	58.01	55.77
CLB-range, 130M	21.28	101.13	30.58	63.50	55.58	72.94	47.27	47.14
CLB-low, 130M	21.16	102.81	29.97	63.32	61.56	73.27	47.50	46.22
CLB-fixed 0.1, 130M	22.96	113.69	32.92	66.55	56.31	79.56	53.19	51.90
CLB-fixed 0.6, 130M	22.50	113.50	33.23	65.39	58.23	77.09	54.13	53.47
CLB-fixed 0.9, 130M	22.84	116.98	33.70	66.34	58.21	80.82	53.47	52.07
CLB-trunc. normal, 130M	22.56	112.74	33.91	65.58	58.18	77.20	54.58	53.89
CLB-Brownian, 130M	22.64	111.96	33.91	66.24	56.72	79.96	53.57	52.41
Loop, 130M	22.38	113.67	33.82	65.05	59.11	77.60	54.89	52.89
AR, 700M	16.99	79.77	25.13	52.01	47.23	50.77	40.32	42.72
CLB-range, 700M	16.66	73.55	23.01	51.65	44.61	49.08	38.12	38.76
CLB-low, 700M	16.69	74.09	23.46	51.41	48.03	48.48	39.43	41.68

Table S15: OWT-trained 2.1B MoE categorical checkpoints, context 1,024, evaluated at $T = 0.6$ for every bridge. Values are PPL. Row labels state MoE-stage updates after initialization; initialization, routing, and budgets differ. Fixed $T = 0.6$ denotes the training coordinate, not an independently tuned evaluation. The stopped 40K run is retained rather than replaced by a better earlier checkpoint.

Model / readout	OWT	PTB	Wiki	LM1B	LAMBADA	AG	PubMed	Arxiv
AR, 70K	15.15	70.16	23.47	48.31	60.01	42.61	49.41	48.09
Fixed experts, range, 70K	14.98	64.27	20.85	48.11	46.14	41.12	35.80	37.40
Fixed experts, $T = 0.6$, 40K	15.76	72.18	22.91	49.19	53.69	45.19	43.37	44.05
Reselected, $d_T = 0$, 20K	15.96	70.39	22.36	49.65	45.55	45.92	36.95	37.80
Reselected, $d_T = 64$, 5K	16.47	72.89	23.28	51.35	47.79	46.77	38.48	39.37
Reselected, $d_T = 256$, 5K	16.48	72.40	22.83	51.08	45.84	46.26	37.41	39.16
No-shared verification, 16K	16.16	70.86	23.03	51.05	50.75	46.49	39.75	40.91
Legacy low-dim., 5K	16.62	73.66	24.05	51.88	47.00	47.63	38.25	37.57

14.9782 OWT PPL at the prespecified $T = 0.6$, vs. 15.1541 for MoE-AR. These fixed-coordinate readouts are reported separately from the validation-selected results in Supplementary Sec. E.1. The remaining MoE rows are endpoint diagnostics, not matched routing ablations: the no-shared verification checkpoint reaches 16K, the coordinate-free router reaches 20K, and the 64- and 256-dimensional reselected routers reach 5K. The legacy low-dimensional row has an AR-derived initialization. The fixed-coordinate 40K run retains its documented deterioration; its OWT PPL of 15.76 cannot be compared with the 70K range result as a coordinate-only ablation.

The default OWT-trained range bridges have lower PPL than their respective independent AR checkpoints on all seven external collections at each of the three scales, as shown in Fig. S11(a). At 700M the reductions range from 0.70% on LM1B to 9.28% on Arxiv; the 2.1B MoE reductions range from 0.40% on LM1B to 27.54% on PubMed. These are single-run comparisons without uncertainty estimates, not evidence that improvements grow monotonically with model size. The favorable ranking does not extend to every variant: 700M CLB-low is worse than AR on LAMBADA (48.03 vs. 47.23), and the shorter-budget fixed-coordinate and routing variants have mixed rankings.

Native LM1B gains do not transfer unchanged to these streams. Range CLB is worse than AR on all eight collections in this format; fixed-coordinate CLB improves over AR but not Loop. Even the LM1B column changes sharply under different preprocessing and sample construction, including removal of training-time separators. This does not isolate the causal contribution of separators alone.

Table S16: LM1B-trained 130M categorical checkpoints after 10B training tokens, context 128. The seven external streams have no added separators; OWT retains its original preprocessing. The LM1B column differs from the native BOS/EOS-packed benchmark in Table 1: it is an in-domain format-change diagnostic, not zero-shot transfer. All CLB readouts use $T = 0.6$.

Model / readout	OWT	PTB	Wiki	LM1B	LAMBADA	AG	PubMed	Arxiv
AR	763.87	639.96	304.01	130.41	1202.59	430.42	1097.86	2398.02
Loop	413.89	439.67	158.55	61.49	667.82	222.11	667.27	1204.54
CLB-range	954.59	811.99	403.60	148.17	1314.77	535.55	1450.21	2902.93
CLB-fixed 0.6	627.73	576.36	274.74	102.10	920.81	373.47	1093.70	1971.76

The survey is therefore not a second estimate of native LM1B PPL and does not support a general claim of cross-domain superiority.

H.3 Masked-diffusion bounds

Table S17: Our MDLM adaptation (Sahoo et al., 2024): OWT-trained, 70K updates, context 1,024. Every entry is $\exp(\widehat{\mathrm{NELBO}})$, not exact categorical PPL. Prefix one conditions on the first token; prefix zero scores the whole block. These distinct target sets must not be pooled.

Model / readout	OWT	PTB	Wiki	LM1B	LAMBADA	AG	PubMed	Arxiv
Raw, prefix 1	43.53	175.16	64.86	115.19	76.64	157.43	79.56	69.49
Raw, prefix 0	43.69	175.51	65.10	115.52	76.86	157.95	79.87	69.80
EMA, prefix 1	38.51	155.86	57.24	99.83	69.41	133.20	72.03	62.29
EMA, prefix 0	38.65	156.20	57.44	100.12	69.62	133.68	72.31	62.56

Table S18: Our MDLM adaptation (Sahoo et al., 2024): LM1B-trained, 10B tokens, context 128. Every entry is $\exp(\widehat{\mathrm{NELBO}})$, not exact categorical PPL. Prefix one conditions on the first token; prefix zero scores the whole block. These distinct target sets must not be pooled.

Model / readout	OWT	PTB	Wiki	LM1B	LAMBADA	AG	PubMed	Arxiv
Raw, prefix 1	412.24	413.89	205.18	82.80	526.61	225.13	668.85	1155.42
Raw, prefix 0	471.20	469.34	233.85	94.80	599.77	256.85	762.79	1316.02
EMA, prefix 1	382.41	402.20	190.63	77.06	497.47	212.55	617.77	1089.58
EMA, prefix 0	435.97	454.95	216.72	88.04	565.07	241.92	702.76	1237.73

These seeded Monte Carlo readouts are new evaluations, not replacements for the historical MDLM checkpoint ledger. Small differences from its OWT endpoint are retained rather than forced to match; for example, the present 130M raw prefix-one OWT estimate is 43.53, whereas the earlier draw produced 43.56.

H.4 Final coordinate scans and within-checkpoint readout

The three default OWT-trained bridge endpoints have 21-point coordinate scans with increments of 0.05. Their observed validation minima all occur at $T = 0.55$: 21.2807 PPL for 130M, 16.6556 for 700M, and 14.9767 for 2.1B MoE. The corresponding prespecified $T = 0.6$ readouts are 21.2821, 16.6568, and 14.9782. Fig. S11(b) compares the full scans after normalization by each checkpoint's own preset readout. The small interior differences do not establish a unique optimum or a continuous improvement trajectory. The $T = 0$ cases use a distinct time-feature mask and are not joined to the positive-coordinate curves.

The campaign also retains first-pass CE from each bridge checkpoint. All 144 model–collection readouts at $T = 0.6$ (18 bridge configurations and eight collections) have lower mixture CE than

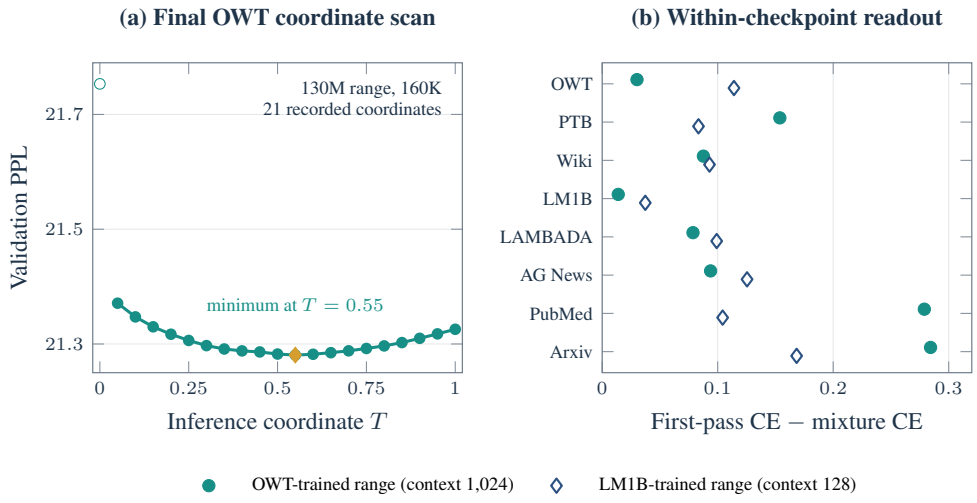


Figure S12: Final-checkpoint diagnostics. **(a)** OWT range coordinate scan; the diamond marks the observed validation minimum, not an independently established optimum. Hollow $T = 0$ denotes the distinct time-feature branch. **(b)** First-pass minus mixture CE at $T = 0.6$ within each range checkpoint; positive values mean lower average loss. Cohorts have different contexts. Neither comparison uses independent AR or measures token-level correction.

their own first pass; reductions range from 0.000163 to 0.284294 nats. Of the 60 additional OWT scan readouts, 59 also improve their first-pass CE. The exception is the 2.1B range bridge at $T = 0$: mixture CE is 2.740670 vs. first-pass CE 2.733033, a deterioration of 0.007636 nats. Thus, the favorable preset readouts do not imply that every bridge state or second pass must help.

These are within-checkpoint average-loss diagnostics, not comparisons with independently trained AR models, token-correction counts, or guarantees for unseen inputs. In particular, LM1B-trained range CLB improves its own first pass while its final output remains worse than independent AR on all eight campaign collections. These measurements do not estimate the gain margin or gate error in Theorem 1 and therefore do not verify its sufficient conditions.

H.5 Separating within-checkpoint and independent-baseline gains

Let L_{AR}, L_{first}, and L_{mix} be token-averaged CE for independent AR, CLB's first pass, and its mixture. The first-pass deficit $D_{\mathrm{first}} = L_{\mathrm{first}} - L_{\mathrm{AR}}$, within-checkpoint gain $G_{\mathrm{within}} = L_{\mathrm{first}} - L_{\mathrm{mix}}$, and net gain $G_{\mathrm{net}} = L_{\mathrm{AR}} - L_{\mathrm{mix}}$ satisfy the accounting identity

$$G_{\mathrm{net}} = G_{\mathrm{within}} - D_{\mathrm{first}}.$$

A positive deficit means that CLB's first pass is worse than independent AR. These quantities are descriptive differences on the same scored targets, not causal effects of training, verification, or the gate.

Table S19: OWT CE and readout-gap accounting in nats. AR is independently trained; first and mixture are from the same CLB checkpoint. Deficit is first minus AR; within is first minus mixture; net is AR minus mixture. Every bridge uses the prespecified $T = 0.6$. Budget, initialization, and compute differences remain.

Scale	AR CE	First CE	Mixture CE	Deficit	Within	Net
130M	3.074614	3.087994	3.057866	0.013380	0.030128	0.016748
700M	2.832506	2.838106	2.812818	0.005600	0.025288	0.019688
2.1B MoE, 60K	2.727992	2.737213	2.710659	0.009221	0.026554	0.017333

Table S19 summarizes OWT, using the 60K MoE-stage endpoint from Supplementary Sec. E.1. Its MoE differences are computed from the six-decimal CE values in the original logs. The separate archived cross-corpus campaign contains 24 comparisons across three scales and eight collections, but its MoE pair is at 70K, not 60K. Each campaign pair matches context (1,024), token hash, observed prefix, samples, and scored targets, using raw categorical readouts and CLB $T = 0.6$. This matches evaluation inputs, not training budgets or initialization (Supplementary Sec. H).

In that archived campaign, the final mixture improves over AR in all 24 comparisons, but the first pass is worse in 17: eight at 130M, five at 700M, and four at 2.1B MoE. These counts do not describe 60K MoE transfer and do not show that training deliberately sacrifices the first pass; they distinguish the two comparison baselines.

The checkpoints recur across collections, so these counts are not independent training replications or significance tests. The decomposition neither measures token-level correction nor separates feedback from extra computation and mixing. It also does not verify the population and gate-learning assumptions of Theorem 1.

I LM1B Sentence Boundaries and Position Interpolation

These diagnostics use the completed 10B-token checkpoints and the same first 64,000 raw LM1B test sentences. They complement the native packed benchmark rather than replace it. All four categorical readouts use the final distribution; CLB has two passes and inference $T = 0.6$. PPL is computed from total NLL and valid target count, not by averaging sentence PPLs.

I.1 Sentence-isolated vs. packed evaluation

Each native sentence input contains BOS, at most 126 content tokens, and EOS. Context is reset between sentences, and exact-length buckets avoid attention to padding. Of 64,000 sentences, 105 are truncated, omitting 4,823 content tokens from evaluation; the original data remain unchanged. The sentence readout scores 1,860,526 targets. A same-source packed control uses 14,764 length-128 blocks and 1,875,028 targets. Its token stream matches the corresponding prefix of the original test binary, but sentence truncation, packing tails, and block-end EOS change the scored target set. Therefore the difference is not a pure single-variable estimate of cross-sentence context.

Table S20: LM1B categorical PPL under sentence-isolated and same-source packed protocols. Native length-128 checkpoints are compared with an *untrained* position-table interpolation to length 256. AR follows Radford et al. (2019); Loop and CLB are ours. Scored targets differ between columns, as detailed in the text.

	Sentence-isolated		Packed, same source	
Model	Native 128	Interpolated 256	Native 128	Interpolated 256
AR	29.6109	40.7034	27.9376	35.0412
Loop	29.2710	39.2970	27.7050	34.3836
CLB-range	29.1623	40.8839	27.5805	36.9597
CLB-fixed 0.6	29.2101	40.9891	27.6226	36.0282

The native sentence protocol preserves the categorical ordering: range CLB reaches 29.1623 PPL, compared with AR's 29.6109 and Loop's 29.2710. Range CLB's relative reduction from AR is 1.51%, while the same-source packed reduction is 1.28%. These single-seed observations do not establish a statistically significant difference between protocols.

I.2 Untrained extension of learned position embeddings

For evaluation only, the learned 128×768 position table is linearly interpolated to 256×768, with new position j mapped to old coordinate $127j/255$ and both endpoints preserved. The attention mask is extended, but no optimizer step, retraining, norm rescaling, or checkpoint overwrite occurs. Thus this is not a model trained natively at context 256.

All four categorical models deteriorate after interpolation in both protocols (Table S20). The transformed sentence inputs contain at most 254 content tokens; six sentences are truncated, omitting 725 content tokens, and 1,864,624 targets are scored. The 7,311 packed blocks score 1,864,305 targets. Identical raw sentences do not imply identical scored tokens across lengths. Because most sentences are already shorter than 128, deterioration also reflects changed position vectors within short inputs, not simply exposure to longer contexts.

Table S21: Separate MDLM conditional $\exp(\widehat{\mathrm{NELBO}})$ diagnostics (Sahoo et al., 2024), with BOS observed and eight stratified mask draws per sample. These are Monte Carlo bounds, not the categorical PPL of Table S20.

	Sentence-isolated		Packed, same source	
Weights	Native 128	Interpolated 256	Native 128	Interpolated 256
Raw	53.5345	895.3219	45.0806	1040.3159
EMA	50.2660	885.2025	42.0902	1117.4528

MDLM estimates also deteriorate sharply. Restoring the original position table exactly recovers FP32 scores on two fixed 45-token inputs: AR CE follows $3.721315 \to 3.998912 \to 3.721315$ and MDLM fixed-mask CE $2.383431 \to 5.814936 \to 2.383431$. The latter is not NELBO; this recovery check covers only those two inputs, not the full collection.

I.3 Precision diagnostic

The full tables use FP32 weights with BF16 autocast and exact-length batches of at most 64 (native) or 32 (interpolated). Table S22 checks numerical sensitivity against FP32 on the same 2,048 native sentences and 58,545 targets, not all 64,000 sentences.

Table S22: Native sentence-level precision check on a shared 2,048-sentence subset. CE differences are BF16 minus FP32; both columns use the same scored tokens.

Model	BF16 PPL	FP32 PPL	CE difference
AR	29.3642	29.3473	0.00057675
Loop	29.1625	29.1729	-0.00035450
CLB-range	28.8659	28.8642	0.00005817
CLB-fixed 0.6	29.1737	29.1682	0.00019004

J MULTIPLE-CHOICE EVALUATION AND GENERATION EXAMPLES

J.1 ANSWER-SELECTION PROTOCOL AND TRACEABILITY

Table 4 evaluates four OWT-trained checkpoints on PIQA (Bisk et al., 2020), ARC-Easy and ARC-Challenge (Clark et al., 2018), WinoGrande (Sakaguchi et al., 2020), and HellaSwag (Zellers et al., 2019). The respective evaluation sizes are 1,838, 2,376, 1,172, 1,267, and 10,042 examples. PIQA, WinoGrande, and HellaSwag use their validation splits; both ARC tasks use their test splits. All four models use zero-shot prompts and identical example IDs, labels, and dataset hashes. The 130M CLB checkpoint is the final 160K range-trained model; its AR counterpart retains the recorded 80K internal step and 80K resume offset. At nominal 700M, AR and CLB use their cumulative-70K checkpoints, with CLB's final internal step of 35K following a 35K resume. This comparison shares evaluation inputs, not strictly matched training budgets, initialization, or computation (Supplementary Secs. C.1 and C.2).

For each candidate answer, we score the full text after inserting that answer into its context and select the candidate with the lowest token-averaged negative log likelihood, equivalently the lowest PPL. An EOS token is prepended as an unscored prefix; all subsequent text tokens, including context tokens, enter the score. Thus these are full-text-PPL answer-selection accuracies, not token-prediction accuracy or conventional continuation-only accuracy. Evaluation uses GPT-2 tokenization, FP32 weights with BF16 forward passes, FP32 probabilities, and FP64 accumulation. CLB uses its two-pass probability mixture at $t = 0.6$. The reported mean gives each of the five tasks equal weight, and the improvement is the difference between CLB and AR means in percentage points, computed before rounding. These are single-checkpoint comparisons, without a claim of statistical significance.

The secondary continuation-only readout without a BOS prefix gives macro-averages of 40.63% and 40.55% for AR and CLB at 130M, and 43.28% and 43.76% at nominal 700M. The corresponding CLB-minus-AR differences are -0.08 and $+0.49$ percentage points. The gains in Table 4 therefore depend on the stated scoring and prefix convention; this comparison does not isolate either factor alone.

The September 26 campaign contains 20 model–task records and 66,780 model–example evaluations. Each reported accuracy was recomputed from the per-example candidate scores and gold labels, checked against the saved correct counts, and matched to the four-model summary. The accompanying `data/owt_mc_accuracy_20260926.json` records exact counts, unrounded accuracies, and checkpoint, dataset, and result-file hashes. This campaign compares independent AR and CLB checkpoints and is distinct from the historical within-checkpoint readout comparisons in Supplementary Sec. C.3.

J.2 SELECTED PAIRED CONTINUATIONS

Figs. S13 and S14 show two selected prompt pairs from the nominal 700M models, giving two examples per model. We chose the story and public-library prompts for readable paired illustration from four archived prompts; the other two concern ocean science and travel. Each displayed continuation contains its first three complete paragraphs, without rewriting or internal omissions. These are curated opening excerpts, not random samples or a ranking of overall generation quality. All eight full continuations are retained in `data/owt_generation_20260926.json`. Both models' longer outputs contain repetition and topic drift, and generated names, quotations, and factual assertions have not been verified.

Both models use the same prompt and corresponding random seed, temperature 0.8, top-p 0.95, and no repetition penalty. Each full continuation contains 1,024 new tokens, with EOS disabled to enforce that length and no initial BOS token. Generation uses FP32 with TF32 disabled, unlike the BF16 forward passes used for multiple-choice evaluation. CLB uses two passes, $t = 0.6$, top-10 soft embeddings, and its learned probability gate. Once the context exceeds 1,023 real tokens, generation retains the most recent 1,023 and rebuilds the cache with reset positions. The same settings apply to both models; the examples illustrate sampled behavior rather than establish a general quality or factuality advantage.

Prompt. When the last train left the station, Emily noticed a small leather suitcase beneath the bench. Inside, she found

AR (700M)	**CLB (700M)**
a bunch of papers. “I was a little nervous,” she says. “But I wasn’t scared at all.” That’s not to say that Emily’s life hasn’t changed dramatically. She’s out of jail, back to work and has started a new job as an immigration researcher at the Cornell Institute of Social Policy. But not much has changed for Emily, a 36-year-old who grew up in western New York. “In my head, I’m still a student,” she says. “I’m still struggling with depression. I can’t remember how it felt to come home and be homeless.” *[Continuation omitted.]*	a bottle with a note on it. “Somehow, my little suitcase of precious cargo ended up in my bunk,” she said. Emily jumped into the arms of her bunkmate and made her way downstairs. The boy and his mother had come down from the bus station to see their daughter. The mother hugged Emily and then she walked to the first-floor master bedroom where the man slept. The boy found a slice of pizza hidden underneath a bed and some bagels. He reached for Emily’s suitcase and told her to look for it. The father opened the bag to find a note: “Watch out for my friend, I’m coming to you.” *[Continuation omitted.]*

Figure S13: Selected story continuations, seed 20260926. Each column reproduces the first three paragraphs of a 1,024-token output; omitted tails are retained in the accompanying data.

Prompt. Public libraries are changing as more people read and work online. Yet their role in local communities remains important because

AR (700M)	**CLB (700M)**
many libraries struggle to recruit and retain new readers. The addition of online resources such as ebooks and services, however, is changing that. “People now are experiencing a whole new world of digital reading,” says Ms. Roberts. “It’s not that we’ve got an audience for everything that’s online — everyone’s kind of a heavy reader.” Mr. Novak, too, has seen the benefits of e-books: “There’s a huge difference between reading through a newspaper and actually being able to get to the Web site and the page,” he says. *[Continuation omitted.]*	the majority of the money generated is not directed to the local community. The digital library is providing new resources that help communities re-create their own spaces, cultures and landscapes, said Heather Griffin, director of the National Public Library Association. “We’re seeing communities come together around local resources, that enable local citizens to make their voices heard and support local efforts to make library resources more accessible, accessible and accessible,” she said. *[Continuation omitted.]*

Figure S14: Selected public-library continuations, seed 20260928. Each column reproduces the first three paragraphs of a 1,024-token output; omitted tails are retained in the accompanying data.