BOLTZMANN ROUTING FOR ENERGY-COMPATIBLE MIXTURE OF EXPERTS

Bharat Runwal
IBM Research

Bishwajit Saha
IBM Research

Benjamin Hoover
IBM Research

Mauro Martino
MIT-IBM Watson AI Lab

Nima Dehmamy
IBM Research

ABSTRACT

The Energy Transformer (ET) recasts the forward pass as gradient descent on a scalar energy, connecting attention to Modern Hopfield Networks and associative memory. Scaling ETs via Mixture-of-Experts (MoE) breaks this variational structure: standard router weights depend on the token state, producing a router gradient residual that prevents the MoE output from being any energy's gradient. We propose **Boltzmann Routing**, which eliminates the external router and derives expert selection from a free-energy functional $\mathcal{F} = -\beta_r^{-1} \log \sum_e \exp(-\beta_r E_e)$. We prove that the negative gradient of $\mathcal{F}$ exactly recovers the weighted expert output with zero residual, that the combined system admits a Lyapunov function, and that attention and routing are *dual instances of the same associative retrieval mechanism*. Experiments across three scales (8 to 32 experts) show that Boltzmann routing achieves accuracy comparable to standard MoE (0.440 avg at 8 experts) *without any auxiliary balancing loss*, while a distillation variant maintains near-perfect load balance at 32 experts. A cross-scale analysis reveals a fundamental tension: exact energy compatibility comes at the cost of expert collapse at scale, and collapse count alone does not determine performance, with implications for energy-based routing more broadly.

1 INTRODUCTION

Modern Hopfield Networks (Hopfield, 1982; Krotov & Hopfield, 2016; Ramsauer et al., 2021) store patterns as attractors of an energy landscape and retrieve them via gradient descent. The Energy Transformer (ET) (Hoover et al., 2023) builds on this principle: attention, feed-forward layers, and the iterative forward pass are all cast as minimizing a single energy functional $E_{\text{total}} = E^{\text{ATT}} + E^{\text{FF}}$. This variational structure guarantees monotonic energy decrease, convergence to fixed points, and a transparent connection to associative memory. More recently, Dehmamy et al. (2025) introduced Energy-GPT, a variant of ET suited for auto-regressive language modeling, which we will focus on in this paper.

Mixture-of-Experts (MoE) (Shazeer et al., 2017; Fedus et al., 2022; Dai et al., 2024) has emerged as the dominant paradigm for efficient scaling. A natural question arises: *can MoE be integrated into Energy Transformers while preserving their energy-based structure?* The answer is not straightforward. In standard MoE, the output is $\text{MoE}(h) = \sum_e w_e(h)\, f_e(h)$ with input-dependent router weights $w_e(h)$. If each expert has energy E_e with $f_e = -\nabla_h E_e$, the chain rule produces a *router gradient residual*:

$$-\nabla_h \Big[\sum_e w_e(h)\, E_e(h)\Big] \;=\; \underbrace{\textstyle\sum_e w_e\, f_e(h)}_{\text{desired}} \;-\; \underbrace{\textstyle\sum_e \nabla_h w_e \cdot E_e(h)}_{\text{residual} \neq 0}\,. \tag{1}$$

No energy function exists whose gradient yields the standard MoE output. The stop-gradient fix ($\nabla_h w_e := 0$) restores compatibility but freezes routing, limiting expert specialization. Recent energy-based language models (Dehmamy et al., 2025; Gladstone et al., 2025) demonstrate viability at scale but do not address this routing incompatibility.

Contribution. We propose *Boltzmann Routing*, which eliminates the external router and derives expert selection from a free-energy functional over expert energies. Our contribution include:

- We prove *exact* energy compatibility with zero residual (Theorem 1), Lyapunov stability (Theorem 2), and a structural duality revealing that attention and routing are instances of the *same* associative memory retrieval mechanism (Remark 2).
- We evaluate three routing strategies across three ET configurations with 8–32 experts, showing that Boltzmann routing achieves comparable downstream accuracy to standard MoE *without a separate router loss* (Section 6).
- We identify a fundamental tension between energy compatibility and expert collapse at scale, along with cross-scale findings (including that collapse count does not determine performance and that temperature initialization is critical), offering practical guidance for energy-based MoE design.

2 Background

Energy Transformer. The Energy-GPT variant of ET defines a total energy of token representations $\boldsymbol{g} = \{g_A\}_{A=1}^{N}$ (layer-normed) as $E_A = E_A^{\text{ATT}}(\boldsymbol{g}) + E^{\text{FF}}(g_A)$. The attention energy $E_A^{\text{ATT}}(\boldsymbol{g}) = -\log \sum_{B<A} \exp(S_{AB})$ is a negative log-partition function, with $S_{AB} = g_A^T W_Q^T W_K g_B$. For the feed-forward energy, Energy-GPT showed good performance using $E^{\text{FF}}(h) = -\operatorname{GELU}(W_1 h)^\top (W_2 h)$ with $W_1, W_2 \in \mathbb{R}^{m\times d}$, however, Hopfield network variants such as $E^{\text{FF}}(h) = -\|\operatorname{GELU}(Wh)\|^2$ also work. The forward pass performs T iterations of $h \leftarrow h - \eta \nabla_h E_A$ for each token A. Although each E_A does not monotonically decrease, due to it being coupled to tokens $B < A$, Dehmamy et al. (2025) showed in very networks (i.e. large T) E_A will eventually converge after previous tokens converge.

MoE with Energy Experts. Each expert $e \in \{1, \dots, N_e\}$ is an Energy MLP with energy $E_e(h) = -\operatorname{GELU}(W_1^{(e)} h)^\top (W_2^{(e)} h)$. The expert output $f_e(h) = -\nabla_h E_e(h)$ follows a dual-path architecture reusing the same weight matrices. A standard router computes $w_e(h) = \operatorname{softmax}(W_g h)_e$ via a learned gate $W_g \in \mathbb{R}^{N_e \times d}$, producing the incompatibility in equation 1.

3 Boltzmann Routing

Our key idea is to eliminate the external router and let experts "vote for themselves" through their own energy values.

Definition 1 (Boltzmann MoE Energy)**.** Given N_e experts with energies $\{E_e(h)\}_{e=1}^{N_e}$ and a learnable inverse temperature $\beta_r > 0$, the *Boltzmann MoE energy* is the negative free energy of a canonical ensemble over experts:

$$\mathcal{F}^{\text{MoE}}(h) \;=\; -\frac{1}{\beta_r} \log \sum_{e=1}^{N_e} \exp\big(-\beta_r\, E_e(h)\big)\,. \tag{2}$$

The associated *Boltzmann routing weights* $w_e^{\mathcal{B}}(h) = \exp(-\beta_r E_e)/\sum_{e'} \exp(-\beta_r E_{e'})$ assign higher probability to experts with lower energy i.e. those that fit the token best in the energy landscape.

Theorem 1 (Exact Energy Compatibility)**.** *The negative gradient of $\mathcal{F}^{\text{MoE}}$ equals the Boltzmann-weighted expert output with zero residual:*

$$-\nabla_h \mathcal{F}^{\text{MoE}}(h) \;=\; \sum_{e=1}^{N_e} w_e^{\mathcal{B}}(h)\, f_e(h)\,. \tag{3}$$

Proof. Differentiating equation 2 directly: $\nabla_h \mathcal{F}^{\text{MoE}} = \sum_e \frac{\exp(-\beta_r E_e)}{Z}(-\nabla_h E_e) = -\sum_e w_e^{\mathcal{B}} f_e$. An alternative derivation via the Helmholtz free-energy decomposition $\mathcal{F} = \langle E \rangle_{w^{\mathcal{B}}} - \beta_r^{-1} H(w^{\mathcal{B}})$ is given in Section A. □

Why does this work? In standard MoE, the router W_g and experts are independently parameterized, so $\nabla_h w_e$ introduces directions unrelated to the energy landscape. In Boltzmann routing, the weights

Table 1: Structural parallel between attention, Hopfield retrieval, and Boltzmann routing. Note, in Boltzmann routing the softmax is over the expert indices e.

	Attention	**Hopfield retrieval**	**Boltzmann routing**
Score S	$q^\top k$	$\beta\, \xi^\top x^\mu$	$-\beta_r\, E_e(h)$
Weights	$\mathrm{softmax}_A(S_{AB})$	$\mathrm{softmax}_\mu(\beta\, \xi^\top x^\mu)$	$\mathrm{softmax}_e(-\beta_r E_e)$
Retrieves	Context	Stored tokens	Computation

are *derived from* the expert energies, creating a log-sum-exp structure where weight gradients cancel by construction. The router is not an external component; it *emerges from the energy landscape itself*.

Remark 1 (Top-k approximation)**.** *Theorem 1 holds for the full Boltzmann distribution over all N_e experts. In practice, we apply top-k selection and renormalize: $\tilde{w}_e = w_e^{\mathcal{B}} / \sum_{e' \in \text{top-}k} w_{e'}^{\mathcal{B}}$ for $e \in$ top-k. This introduces a small approximation error bounded by the total probability mass of the discarded experts. When β_r is large (sharp routing), the top-k experts carry nearly all the mass, making the approximation tight.*

4 Expert Selection as Associative Retrieval

The Boltzmann MoE energy equation 2 and the Energy-GPT E^{ATT} share a striking structural identity.

Remark 2 (Attention-Routing Duality)**.** *The ET attention energy and the Boltzmann MoE energy are both negative log-partition functions over softmax distributions. Defining expert scores $S_e(h) = -\beta_r E_e(h)$:*

$$\underbrace{E_A^{\mathrm{ATT}} = -\log \sum_B \exp(S_{AB})}_{\textit{attention: sum over tokens}} \quad \longleftrightarrow \quad \underbrace{\mathcal{F}^{\mathrm{MoE}} = -\tfrac{1}{\beta_r} \log \sum_e \exp(S_e)}_{\textit{routing: sum over experts}} . \tag{4}$$

Table 1 makes the parallel precise. In Modern Hopfield Networks (Ramsauer et al., 2021), retrieval produces a softmax-weighted sum over stored patterns. Boltzmann routing produces a softmax-weighted sum over expert computations, where the "similarity" is the negative expert energy. The ET and Energy-GPT with Boltzmann MoE thus performs two complementary forms of associative retrieval: attention retrieves relevant *context* from the sequence, while routing retrieves relevant *computations* from the expert repertoire.

5 Convergence and Implementation

Theorem 2 (Per-Token Lyapunov Stability)**.** *Fix a token A and hold all other token representations $\{g_B\}_{B \neq A}$ constant. Define the per-token energy $E_A(h_A) = E_A^{\mathrm{ATT}}(\boldsymbol{g}) + \alpha\, \mathcal{F}^{\mathrm{MoE}}(h_A)$ with $\alpha > 0$. Assume each expert energy E_e has L_e-Lipschitz continuous gradient and that E_A is bounded below. Under the dynamics $h_A^{(t+1)} = h_A^{(t)} - \eta\, \nabla_{h_A} E_A(h_A^{(t)})$ with $\eta < 2/L$ (where L is the Lipschitz constant of $\nabla_{h_A} E_A$), E_A is a strict Lyapunov function: $E_A(h_A^{(t+1)}) \leq E_A(h_A^{(t)}) - \frac{\eta}{2} \|\nabla_{h_A} E_A(h_A^{(t)})\|^2$.*

The proof follows from the standard descent lemma: both E_A^{ATT} (a log-sum-exp in h_A with fixed context) and $\mathcal{F}^{\mathrm{MoE}}$ (a proper energy by Theorem 1) are L-smooth in h_A. The smoothness constant of the MoE term satisfies $L_{\mathcal{F}} \leq \max_e L_e + \beta_r \mathrm{Var}_{w^{\mathcal{B}}}[\nabla_h E_e]$ (see Section B); full derivation in Section A.

Note that global monotonic decrease across all tokens does *not* hold: in Energy-GPT, updating token B shifts the attention energy E_A^{ATT} for all $A > B$ via the scores S_{AB}. Crucially, $\mathcal{F}^{\mathrm{MoE}}(h_A)$ depends only on token A's own representation and introduces no additional inter-token coupling. Global convergence follows from the cascading argument of Dehmamy et al. (2025): for large T, tokens converge sequentially—token 1 first (no attention dependencies), then token 2 (coupled only to the now-converged token 1), and so on—with the per-token Lyapunov guarantee applying at each stage.

Table 2: Cross-scale results. WikiText PPL (↓), average accuracy over 11 benchmarks (↑), and expert collapse count (Dead = <1% tokens). Best per column in **bold**.

	8×1024 (E2)			32×2048 (E6)			32×1024 (E9)		
Routing	PPL↓	Avg↑	Dead	PPL↓	Avg↑	Dead	PPL↓	Avg↑	Dead
Standard MoE	40.36	**0.440**	**0**	**35.27**	**0.453**	3	42.30	**0.433**	6
Stop-Gradient	**40.17**	0.437	**0**	36.84	0.445	**0**	43.48	0.429	**0**
Boltzmann Pure	41.17	**0.440**	4	46.18	0.420	26	68.72	0.404	22
Std (no aux)	40.23	**0.440**	3	35.51	0.451	24	**42.23**	0.431	23
Boltzmann F5	42.82	0.433	**0**	42.58	0.442	2	57.53	0.419	27

In practice, the Energy Block update (Equation (8)) includes a learned projection Π and layer norm, which deviate from the idealized gradient descent assumed here. These components improve training stability but mean the strict Lyapunov guarantee holds only approximately.

Temperature interpolation. The inverse temperature β_r interpolates between uniform averaging ($\beta_r \to 0^+$: all experts contribute equally) and winner-take-all routing ($\beta_r \to \infty$: lowest-energy expert only). Making β_r learnable lets the model discover the optimal exploration-exploitation trade-off, mirroring the role of temperature in Hopfield retrieval (Demircigil et al., 2017).

Implementation. Boltzmann routing evaluates all N_e expert energies for $\mathcal{F}^{\text{MoE}}$ ($\sim \mathcal{O}(N_e dm)$), then executes only the top-k lowest-energy experts with renormalized weights. No auxiliary loss is needed, the β_r provides implicit entropy regularization. The full algorithm is in Algorithm 1.

6 Experiments

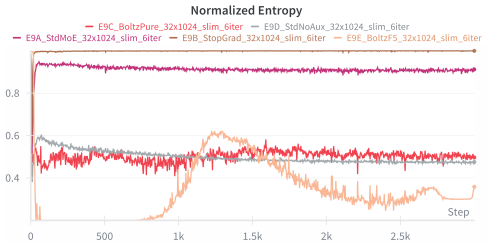
Setup. All experiments use a three-layer architecture: a standard GPT block (softmax attention + SwiGLU MLP), an Energy Block (energy attention + Energy MoE, iterated T=6 times), and a standard GPT post-block. Hidden size 1024, top-2 routing, 30k training steps on NemoTron-CC (Su et al., 2025). We report `acc_norm` for ARC-C, ARC-E, HellaSwag, OBQA, PIQA, SciQ and `acc` for BoolQ, COPA, LAMBADA, RACE, Winogrande. **Avg** is the mean of these 11 scores; WikiText word perplexity (PPL, ↓) is reported separately. More details can be found in Section D. Formal variant definitions with energy compatibility analysis are in Section C; full per-benchmark results and routing dynamics figures for each configuration are in Sections D.1, D.3 and D.4.

Routing variants. We compare five strategies, all using **Energy MLP experts** (Table 3): *Standard MoE*: learned linear gate with aux loss (λ=0.001), not energy-compatible (Section C.1); *Std (no aux)*: same gate without balancing, a collapse stress test; *Stop-Gradient*: detached gate, per-step energy compatible via $\text{sg}(w_e)$ (Section C.2); *Boltzmann Pure*: free-energy routing with exact global compatibility, no external router (Section C.3); *Boltzmann F5*: Boltzmann training with KL-distilled gate for inference (Section C.4).

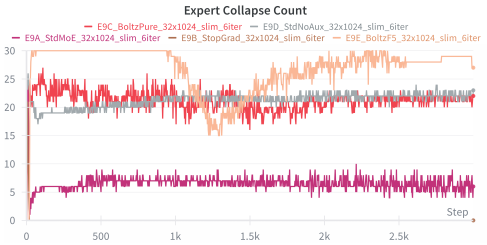
Scale configurations. E2 (Section D.1): 8 experts × 1024 intermediate, dense 4096 (156M total, 144M active), all variants comparable at this scale. **E6** (Section D.3): 32 × 2048, dense 4096 (274M total, 148M active, dense-dominated), collapse differences emerge but dense layers mask them. **E9** (Section D.4): 32 × 1024, dense 512 (185M total, 122M active, MoE-focused), routing quality differences fully exposed. We also performed a temperature sweep (Section D.2) showing that $\beta_r = 2.0$ achieves the best accuracy while the learned β_r converges to ≈ 0.17, near the soft end for E2 configuration.

Finding 1: Energy compatibility costs performance at scale. Standard MoE (non-compatible) achieves the best perplexity at 32 experts (35.27 in E6, 42.30 in E9), while the energy-compatible Boltzmann Pure degrades from 41.17 to 68.72 as expert count and MoE scale grow (Table 2). The router gradient residual equation 1, while theoretically undesirable, appears to provide useful optimization signal that helps maintain expert diversity.

Finding 2: Stop-Gradient degrades at scale. Stop-Gradient achieves the lowest perplexity at 8 experts (40.17, Table 2) through near-perfect load balance (Figure 1a), but drops to third at 32 experts (36.84 in E6, 43.48 in E9). Its detached routing prevents expert co-adaptation, a cost that grows



(a) Normalized Entropy ($H/H_{\max}$)



(b) Expert Collapse Count

Figure 1: E9 routing dynamics over 30k steps (32 experts, MoE-focused). Stop-Gradient maintains perfect entropy but lacks specialization. Boltzmann Pure and F5 collapse catastrophically ($\geq$22/32 dead). Standard MoE loses 6 experts but achieves the best perplexity. Full figures for all configurations in Sections D.1, D.3 and D.4.

with expert count, visible in the flat entropy curve that masks a lack of content-aware specialization (Table 10).

Finding 3: Collapse count does not determine performance. In E9, Std (no aux) with 23 collapsed experts achieves 42.23 PPL, while Boltzmann Pure with *fewer* collapsed (22) achieves 68.72 PPL, that is 63% worse (Table 2 and fig. 1b). The quality of surviving experts matters more than their count: a learned gate produces better-specialized active experts than energy-based routing, which concentrates on suboptimal low-energy attractors.

Finding 4: Dense layers mask routing quality. Moving from E6 to E9 widens the PPL gap between Standard MoE and Boltzmann Pure from 11 to 26 points (Table 2), revealing that large dense layers absorb most modeling burden and mask MoE routing differences. The MoE-focused architecture (E9) is a more discriminative testbed for routing efficacy. Temperature initialization is also critical: in E6, F5 with β_r=0.1 collapses 28/32 experts (62.14 PPL), while β_r=1.0 collapses only 2/32 (42.58 PPL; Tables 7 and 8).

7 Discussion

Boltzmann routing provides the first exact solution to the MoE energy incompatibility problem, with provable Lyapunov stability and a structural duality connecting expert selection to associative memory retrieval. However, our experiments reveal a gap between theory and practice: the energy-compatible Boltzmann Pure variant suffers from catastrophic expert collapse at scale (22 to 26 of 32 experts dead), because the energy landscape funnels tokens to a few low-energy attractors in a self-reinforcing cycle. Standard MoE, which breaks energy compatibility, avoids collapse through aux loss and independent router parameterization. This suggests that the router gradient residual may act as an implicit regularizer promoting expert diversity.

The attention-routing duality (Remark 2) positions the ET as an associative memory with two retrieval mechanisms: attention retrieves the relevant *context* from the sequence, while routing retrieves the relevant *expert dynamics*. Both mechanisms emerge from descent on log-sum-exp free energies, differing only in whether the stored objects are vectors or vector fields. This perspective inherits the log-sum-exp structure underlying the exponential storage capacity results for Modern Hopfield Networks (Ramsauer et al., 2021; Demircigil et al., 2017), though whether analogous capacity bounds apply to expert routing remains an open question.

Limitations and future work. All results are reported without error bars due to computational constraints; differences within $\sim$1% accuracy should not be over-interpreted. Direct empirical verification of the monotonic energy descent predicted by Theorem 2 (e.g., tracking E_{total} across inner-loop iterations) remains an important direction. Energy evaluation cost grows linearly with N_e; pre-filtering strategies could help for $N_e > 128$. Key open directions include: entropy regularization within the Boltzmann framework to prevent collapse without external routers; per-layer or curriculum temperature schedules (our sweep in Section D.2 shows β_r is critical); and scaling within the Energy-GPT framework (Dehmamy et al., 2025) to determine whether the compatibility/collapse trade-off persists at larger scales.

REFERENCES

Damai Dai, Chengqi Deng, Chenggang Zhao, R X Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y Wu, et al. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. *arXiv preprint arXiv:2401.06066*, 2024.

Nima Dehmamy, Benjamin Hoover, Bhaswar Saha, Leo Kozachkov, Jean-Jacques Slotine, and Dmitry Krotov. NRGPT: An energy-based alternative for GPT. *arXiv preprint arXiv:2512.16762*, 2025.

Mete Demircigil, Judith Heusel, Matthias Löwe, Sven Upgang, and Kostas Vlachos. On a model of associative memory with huge storage capacity. *Journal of Statistical Physics*, 168(2):288–299, 2017.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B*, 39(1):1–38, 1977.

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.

Ryan J Gladstone, Barak Durumeric, and Govind Bhatt. Energy-based transformers are scalable learners and thinkers. *arXiv preprint arXiv:2507.01026*, 2025.

Benjamin Hoover, Yuchen Liang, Bao Pham, Rameswar Panda, Hendrik Strobelt, Duen Horng Chau, Mohammed J Zaki, and Dmitry Krotov. Energy Transformer. In *Advances in Neural Information Processing Systems*, 2023.

John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982.

Dmitry Krotov and John J Hopfield. Dense associative memory for pattern recognition. In *Advances in Neural Information Processing Systems*, pp. 1172–1180, 2016.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Milena Pavlović, Geir Kjetil Sandber, Victor Greber, David Kreil, Michael Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Dan Su, Kezhi Kong, Ying Lin, Joseph Jennings, Brandon Norick, Markus Kliegl, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. Nemotron-cc: Transforming common crawl into a refined long-horizon pretraining dataset, 2025. URL https://arxiv.org/abs/2412.02595.

A Full Gradient Derivation via Free-Energy Decomposition

The Boltzmann MoE energy decomposes as:

$$\mathcal{F}^{\mathrm{MoE}} = \sum_e w_e^{\mathcal{B}} E_e + \frac{1}{\beta_r} \sum_e w_e^{\mathcal{B}} \log w_e^{\mathcal{B}} = \langle E \rangle_{w^{\mathcal{B}}} - \frac{1}{\beta_r} H(w^{\mathcal{B}}), \tag{5}$$

where $\langle E \rangle$ is the expected energy and H is the Shannon entropy. Differentiating by the product rule:

$$\nabla_h \mathcal{F}^{\mathrm{MoE}} = \sum_e w_e^{\mathcal{B}} \nabla_h E_e + \sum_e (\nabla_h w_e^{\mathcal{B}}) \underbrace{\left(E_e + \tfrac{1}{\beta_r}(1 + \log w_e^{\mathcal{B}})\right)}_{= \mathcal{F}^{\mathrm{MoE}} + 1/\beta_r \text{ for all } e}. \tag{6}$$

The parenthetical is constant across e despite appearances: from $w_e^{\mathcal{B}} = \exp(-\beta_r E_e)/Z$ we get $\log w_e^{\mathcal{B}} = -\beta_r E_e - \log Z$, so $E_e + \beta_r^{-1} \log w_e^{\mathcal{B}} = E_e - E_e - \beta_r^{-1} \log Z = \mathcal{F}^{\mathrm{MoE}}$, where the E_e terms cancel. Adding the remaining β_r^{-1} gives $\mathcal{F}^{\mathrm{MoE}} + \beta_r^{-1}$, independent of e. Since $\sum_e \nabla_h w_e^{\mathcal{B}} = \nabla_h (\sum_e w_e^{\mathcal{B}}) = \nabla_h 1 = 0$, the second sum vanishes entirely. Therefore $\nabla_h \mathcal{F}^{\mathrm{MoE}} = \sum_e w_e^{\mathcal{B}} \nabla_h E_e = -\sum_e w_e^{\mathcal{B}} f_e$. □

B Smoothness Bound

Proposition 3. *If each E_e is L_e-smooth (i.e., $\|\nabla^2 E_e\| \le L_e$), then $\mathcal{F}^{\mathrm{MoE}}$ is $L_{\mathcal{B}}$-smooth with:*

$$L_{\mathcal{B}} \le \max_e L_e + \beta_r \sum_e w_e^{\mathcal{B}} \|\nabla_h E_e - \overline{\nabla E}\|^2, \tag{7}$$

where $\overline{\nabla E} = \sum_e w_e^{\mathcal{B}} \nabla_h E_e$ is the Boltzmann-weighted mean gradient.

The second term is the Boltzmann-weighted variance of expert gradients. When experts are well-specialized (gradients point in different directions), this variance is large, requiring a smaller step size η. In practice, gradient clipping and per-layer β_r scheduling control this.

C Evaluated Routing Strategies

All experiments share a three-layer architecture. The first and last layers are standard GPT blocks (softmax attention + SwiGLU MLP), each executed once. The middle layer is an **Energy Block** consisting of energy attention and an Energy MoE layer, iterated $T = 6$ times via gradient-based state refinement. Within this inner loop, each step applies a shared layer norm and updates the state via:

$$\bar{h}^{(t)} = \mathrm{LN}(h^{(t)}), \qquad h^{(t+1)} = h^{(t)} - \eta\, \Pi\big(\nabla_h E^{\mathrm{ATT}}(\bar{h}^{(t)}) + \alpha_{\mathrm{ff}}\, \mathrm{MoE}(\bar{h}^{(t)})\big), \tag{8}$$

where E^{ATT} is the energy attention functional, $\mathrm{MoE}(\cdot)$ is the expert output, Π is a learned projection, and α_{ff} is a **learnable scalar** (initialized to 4.0) that controls the relative contribution of the MoE layer to the energy descent step. This scale parameter allows the model to learn the optimal balance between attention-driven and expert-driven updates during training.

Crucially, **every expert across all variants is an Energy MLP**, a dual-path architecture with energy $E_e(h) = -\,\mathrm{GELU}(W_1^{(e)} h)^\top (W_2^{(e)} h)$ and output $f_e(h) = -\nabla_h E_e(h)$. The variants differ *only* in how tokens are routed to experts within the Energy Block and whether this routing preserves the energy interpretation.

Table 3 summarizes the five evaluated strategies and their theoretical properties.

C.1 Baseline: Standard Router with Energy Experts

The standard MoE baseline uses energy experts paired with a conventional learned linear router. This is the configuration that motivates the entire formulation space: it achieves strong empirical performance but breaks energy compatibility.

Table 3: Evaluated routing strategies. All use Energy MLP experts; they differ in routing mechanism and energy compatibility.

Variant	Energy Compat.	Aux Loss	External Router	Key Mechanism
Standard MoE	No ($\nabla_h w_e \neq 0$)	Yes	Yes	Learned gate $W_g h$
Std (no aux)	No ($\nabla_h w_e \neq 0$)	No	Yes	Same gate, no balancing
Stop-Gradient	Per-step exact	Yes	Yes	$\mathrm{sg}(w_e)$ each iteration
Boltzmann Pure	Exact (global)	No	No	Free-energy routing
Boltzmann F5	Exact (train)	KL	Train: No / Infer: Yes	Boltzmann $\rightarrow$ distilled gate

C.1.1 Formulation

Definition 2 (Standard Router with Energy Experts)**.** A learned gate $W_g \in \mathbb{R}^{N_e \times d}$ produces routing logits. After top-k selection, the weights are renormalized via softmax:

$$w_e(h) = \frac{\exp([W_g h]_e)}{\sum_{e' \in \text{top-}k} \exp([W_g h]_{e'})}, \qquad y = \sum_{e \in \text{top-}k} w_e(h)\, f_e(h). \tag{9}$$

Each expert output is $f_e(h) = -\nabla_h E_e(h)$, the gradient of its scalar energy. An optional auxiliary load-balancing loss $\mathcal{L}_{\text{aux}} = \lambda \sum_e \hat{p}_e \cdot \bar{p}_e$ penalizes uneven expert utilization, where $\hat{p}_e$ is the fraction of tokens routed to expert e and $\bar{p}_e$ is its mean routing probability.

C.1.2 Energy Incompatibility

Despite using energy experts, this formulation is *not* energy-compatible. Differentiating the naïve energy $\sum_e w_e(h)\, E_e(h)$ yields:

$$-\nabla_h \Big[\sum_e w_e(h)\, E_e(h)\Big] = \underbrace{\sum_e w_e(h)\, f_e(h)}_{\text{MoE output (computed)}} \; - \; \underbrace{\sum_e (\nabla_h w_e)\, E_e(h)}_{\text{router gradient residual}}. \tag{10}$$

Since $w_e(h) = \mathrm{softmax}(W_g h)_e$ depends on h through the learned gate, $\nabla_h w_e \neq 0$. The residual term introduces directions in the update step that do not correspond to any energy descent, breaking the Lyapunov guarantee.

Within the T-iteration inner loop of the Energy Block, the energy attention contribution performs valid energy descent; the MoE contribution does not. Nevertheless, backpropagation through the full computation graph remains well-defined, and the model trains normally via standard gradient-based optimization.

C.1.3 Two Ablations

- **Standard MoE** ($\lambda_{\text{aux}} = 0.001$): The router gradient residual is present but the aux loss maintains load balance, preventing expert collapse even at 32 experts.
- **Std (no aux)** ($\lambda_{\text{aux}} = 0$): Removes all explicit routing regularization. This is a stress test: can the learned gate self-organize without any balancing signal? At 8 experts (E2) this works reasonably (3/8 collapsed); at 32 experts (E9) it collapses severely (23/32) yet still achieves competitive perplexity (Tables 11 and 12), revealing that surviving expert quality matters more than quantity.

C.2 Stop-Gradient Routing (Bilevel)

The simplest fix for energy compatibility: freeze the routing weights at each inner-loop iteration via stop-gradient, treating routing and token-state descent as two levels of a bilevel optimization problem.

C.2.1 FORMULATION

Definition 3 (Stop-Gradient Routing)**.** At each inner-loop iteration t, routing weights are computed from the current state and immediately detached:

$$w_e^{(t)} = \mathrm{sg}\big(\mathrm{softmax}(\text{top-k}(W_g\, h^{(t)}))\big), \qquad y^{(t)} = \sum_{e \in \text{top-}k} w_e^{(t)}\, f_e(h^{(t)}). \tag{11}$$

Since $\nabla_h w_e^{(t)} = 0$ by construction, the router gradient residual vanishes identically within each descent step.

C.2.2 ENERGY COMPATIBILITY

At each iteration, the update $h^{(t+1)} = h^{(t)} - \eta \nabla_h E_{\text{total}}(h^{(t)}, w^{(t)})$ descends a well-defined energy landscape with $w^{(t)}$ treated as a constant.

Proposition 4 (Per-iteration energy descent)**.** *For step size $\eta \le 1/L_t$ (where L_t is the smoothness constant of $E_{total}(\cdot, w^{(t)})$):*

$$E_{total}(h^{(t+1)}, w^{(t)}) \le E_{total}(h^{(t)}, w^{(t)}) - \frac{\eta}{2} \|\nabla_h E_{total}(h^{(t)}, w^{(t)})\|^2. \tag{12}$$

*However, re-routing at iteration $t{+}1$ may change the energy landscape, so **global monotonicity is not guaranteed**; only per-step descent holds.*

In our experiments, the router is re-evaluated at each of the $T = 6$ inner-loop iterations (bilevel alternating), with $\lambda_{\text{aux}} = 0.001$.

C.2.3 KEY PROPERTIES

- Zero overhead: requires only a single `.detach()` call per iteration.
- The detached router cannot co-adapt with expert gradients during the inner loop, which limits expert specialization, a cost that becomes visible at 32 experts where uniform but content-agnostic routing degrades perplexity.

C.3 BOLTZMANN PURE ROUTING

The central contribution of this paper (Section 3): routing weights derived from expert energies via a free-energy functional, eliminating the external router entirely.

C.3.1 FORMULATION

Definition 4 (Boltzmann Pure Routing)**.** The MoE energy is the Boltzmann free energy over expert energies:

$$\mathcal{F}^{\text{MoE}}(h) = -\frac{1}{\beta_r} \log \sum_{e=1}^{N_e} \exp\big(-\beta_r\, E_e(h)\big), \tag{13}$$

with routing weights $w_e^{\mathcal{B}} = \exp(-\beta_r E_e)/Z$ and output $-\nabla_h \mathcal{F}^{\text{MoE}} = \sum_e w_e^{\mathcal{B}} f_e(h)$. No external router W_g exists; no auxiliary loss is used.

C.3.2 ENERGY COMPATIBILITY

Exact and global: the router gradient residual is *absent* by construction (Theorem 1). Within the Energy Block, the combined energy $E_{\text{total}} = E^{\text{ATT}} + \alpha\, \bar{\mathcal{F}}^{\text{MoE}}$ admits a strict Lyapunov function (Theorem 2).

In our experiments, $\beta_r = 1.0$, top-$k = 2$ with renormalized Boltzmann weights, and $\lambda_{\text{aux}} = 0$.

C.3.3 KEY PROPERTIES

- Strongest theoretical guarantees: exact compatibility, global Lyapunov stability, no external parameters.

- The energy landscape can trap routing in collapsed states: low-energy experts attract more tokens, receive more gradient signal, and further lower their energy, a self-reinforcing cycle.
- Without any balancing mechanism, this collapse is tolerable at 8 experts (4/8 collapsed in E2) but catastrophic at 32 experts (22/32 collapsed in E9, 68.72 vs. 42.30 PPL for Standard MoE; Tables 9 and 12).
- The gap between theory and practice motivates the inference-efficient hybrid variant (Boltzmann F5) described next.

C.4 Inference-Efficient Boltzmann Routing

A dual-mode architecture: Boltzmann routing during training for energy compatibility, distilled to a linear router for efficient inference.

C.4.1 Formulation

Definition 5 (Inference-Efficient Boltzmann)**.** This variant maintains two routing mechanisms:

1. **Training**: Boltzmann weights $w_e^{\mathrm{B}} = \exp(-\beta_r E_e)/Z$ with learnable $\beta_r = \exp(\log \beta_r)$. All expert energies are evaluated for the partition function.
2. **Inference**: Linear router $w_e^{\mathrm{L}} = \mathrm{softmax}(W_g h)_e$, trained via KL distillation:

$$\mathcal{L}_{\text{distill}} = \lambda_{\mathrm{KL}} \cdot \mathrm{KL}\big(\mathrm{sg}(w^{\mathrm{B}}) \,\big\|\, w^{\mathrm{L}}\big). \tag{14}$$

During training, the MoE output uses Boltzmann weights (energy-compatible); at inference, the distilled linear router is used (standard MoE cost).

C.4.2 Energy Compatibility

During training: exact, inheriting all Boltzmann Pure guarantees including Lyapunov stability (Theorem 2).

At inference: approximate. The quality of the approximation is bounded by Pinsker's inequality:

Proposition 5 (Inference bound)**.** *If* $\mathrm{KL}(w^{\mathrm{B}} \| w^{\mathrm{L}}) \leq \epsilon$ *uniformly, then:*

$$\big\|\mathrm{MoE}^{\mathrm{B}}(h) - \mathrm{MoE}^{\mathrm{L}}(h)\big\| \leq \sqrt{2\epsilon} \cdot \max_e \|f_e(h)\|. \tag{15}$$

In our experiments: $\lambda_{\mathrm{KL}} = 0.01$, β_r initialized to 0.1 (E9) or 1.0 (E6), learnable via $\log \beta_r$ parameterization excluded from weight decay.

C.4.3 Key Properties

- Training overhead: $\sim 2.5\times$ standard MoE (all N_e expert energies evaluated per token). Inference cost: identical to standard MoE.
- The KL distillation loss competes with the language modeling objective, which may explain Boltzmann F5's consistently higher perplexity relative to Standard MoE.
- Temperature initialization is critical: in E6, $\beta_r = 0.1$ initialization collapses 28/32 experts (62.14 PPL), while $\beta_r = 1.0$ collapses only 2/32 (42.58 PPL; Tables 7 and 8). This connects to the smoothness bound (Section B): low β_r reduces the gradient variance term $\beta_r \mathrm{Var}_{w^{\mathcal{B}}}[\nabla_h E_e]$, smoothing the landscape but delaying expert differentiation.
- The learned β_r (initialized at 1.0) converges to ≈ 0.17 in E2, suggesting optimizer dynamics or weight decay bias it toward uniformity.

C.5 F6: Gaussian Mixture / Hopfield Routing

Formulation F6 routes tokens based on their squared Euclidean distance to expert prototype means $\boldsymbol{\mu}_e \in \mathbb{R}^d$, establishing connections to Gaussian mixture models Dempster et al. (1977) and modern continuous Hopfield networks Ramsauer et al. (2021).

C.5.1 Energy Functional

Definition 6 (Gaussian Mixture MoE Energy)**.** Given expert means $\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_E\}$ and inverse temperature $\beta_{\mathrm{G}} = 1/T_{\mathrm{G}}$:

$$E_{\mathrm{gauss}}(\mathbf{h}) = -\frac{1}{\beta_{\mathrm{G}}} \log \sum_{e=1}^{N_e} \exp\Big(-\frac{\beta_{\mathrm{G}}}{2} \|\mathbf{h} - \boldsymbol{\mu}_e\|^2\Big). \tag{16}$$

The Gaussian routing weights are $w_e^{\mathrm{G}} = \mathrm{softmax}(-\|\mathbf{h} - \boldsymbol{\mu}_e\|^2/(2T_{\mathrm{G}}))$.

The negative gradient of E_{gauss} pulls $\mathbf{h}$ toward the weighted centroid: $-\nabla_{\mathbf{h}} E_{\mathrm{gauss}} = \sum_e w_e^{\mathrm{G}}(\boldsymbol{\mu}_e - \mathbf{h})$, a soft nearest-neighbor retrieval step analogous to the Hopfield network update rule.

C.5.2 Hybrid Routing and Expert Mean Dynamics

F6 supports a hybrid mode combining distance and linear gate logits:

$$r_e^{\mathrm{hybrid}} = \alpha_{\mathrm{H}} \cdot \Big(-\frac{\|\mathbf{h} - \boldsymbol{\mu}_e\|^2}{2T_{\mathrm{G}}}\Big) + (1 - \alpha_{\mathrm{H}}) \cdot [\mathbf{h}^\top W_g]_e, \tag{17}$$

where $\alpha_{\mathrm{H}} \in [0, 1]$ interpolates between geometric and learned routing. Expert means are maintained via EMA: $\boldsymbol{\mu}_e \leftarrow \gamma\, \boldsymbol{\mu}_e + (1 - \gamma)\, \bar{\mathbf{h}}_e$, mirroring the M-step of the EM algorithm. Router weights are detached ($\bar{w}_e = \mathrm{sg}(w_e^{\mathrm{G}})$) for energy compatibility, following the bilevel (F1) strategy.

C.5.3 Key Properties

- **Geometric interpretability**: routing corresponds to Voronoi-like partitions of representation space.
- **Hopfield connection**: expert means act as stored patterns; routing performs associative retrieval.
- **Natural load balancing**: EMA-maintained means track the token distribution, preventing expert collapse.
- **Negligible overhead**: routing operates in $\mathbb{R}^d$ ($\sim 0.5\%$ additional FLOPs).
- **Combined F6+F3**: Gaussian pre-filter for two-stage Boltzmann routing combines F6's efficiency with F3's theoretical guarantees.

C.5.4 GaussianBoltzmannMoE: Learned Precision Matrices

The isotropic F6 formulation assumes all experts have equal, spherical influence regions. A natural generalization replaces Euclidean distances with Mahalanobis distances, giving each expert a learned precision structure via low-rank factors $W_e \in \mathbb{R}^{k \times d}$, where k is the expert intermediate dimension.

Definition 7 (Mahalanobis Gaussian MoE)**.** Given expert centroids $\{\boldsymbol{\mu}_e\}_{e=1}^{N_e}$, low-rank precision factors $\{W_e\}$, and optional mixing coefficients π_e, the expert energy is:

$$E_e(\mathbf{h}) = \|W_e(\mathbf{h} - \boldsymbol{\mu}_e)\|^2, \tag{18}$$

with full GMM routing logits:

$$\ell_e = \log \pi_e + \tfrac{1}{2} \log\det(W_e^\top W_e) - \|W_e(\mathbf{h} - \boldsymbol{\mu}_e)\|^2 + b_e, \tag{19}$$

where b_e are optional per-expert bias terms for auxiliary-loss-free load balancing (cf. DeepSeek-V3).

The expert output is the exact negative energy gradient:

$$f_e(\mathbf{h}) = -\nabla_{\mathbf{h}} E_e = -2\, W_e^\top W_e\, (\mathbf{h} - \boldsymbol{\mu}_e), \tag{20}$$

preserving energy compatibility: defining the free energy $F(\mathbf{h}) = -\log \sum_e \exp(\ell_e)$, we have $-\nabla_{\mathbf{h}} F = \sum_e w_e(\mathbf{h})\, f_e(\mathbf{h})$ exactly, where $w_e = \mathrm{softmax}(\ell_e)$.

KL distillation router. Following the F5 strategy (§C.4), we add an auxiliary linear gate $W_g \in \mathbb{R}^{N_e \times d}$ trained via KL divergence to mimic the full GMM routing distribution:

$$\mathcal{L}_{\mathrm{KL}} = \lambda_{\mathrm{KL}} \cdot D_{\mathrm{KL}}\big(\log \mathrm{softmax}(W_g\,\mathbf{h}) \;\big\|\; \mathrm{sg}\big(\mathrm{softmax}(\boldsymbol{\ell})\big)\big), \tag{21}$$

where sg denotes stop-gradient. At inference time, only the linear gate is evaluated for top-k expert selection, followed by sparse computation of f_e for the selected experts only. This avoids computing all N_e expert energies, reducing per-token cost from $O(N_e k d)$ to $O(kd + N_e d)$.

Anti-collapse regularization. Expert collapse—where a subset of experts receives all tokens and the rest become inactive—is the primary practical challenge for Gaussian MoE routing. We employ five complementary anti-collapse mechanisms:

1. **Routing entropy bonus** (γ_{ent}): Maximizes per-token routing entropy $H(w_1, \ldots, w_E)$ to prevent winner-take-all dynamics.
2. **Mixing entropy bonus** (γ_{mix}): Maximizes entropy of the mixing coefficients $H(\pi_1, \ldots, \pi_E)$ to keep all experts a priori equally likely.
3. **Centroid repulsion** (λ_{rep}): Penalizes $-\frac{1}{N_e(N_e-1)} \sum_{i \neq j} \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|^2$ to spread expert centroids in representation space.
4. **W-diversity** (λ_{div}): Penalizes pairwise cosine similarity between flattened W_e matrices to encourage distinct precision structures.
5. **Bias-based balancing**: Maintains running per-expert token fractions $\bar{f}_e$ via EMA and updates biases $b_e \mathrel{+}= \alpha\,(1/N_e - \bar{f}_e)$ to steer routing toward underutilized experts, without backpropagation through the balancing signal.

C.5.5 Preliminary Results: Expert Collapse Analysis

We test GaussianBoltzmannMoE on WikiText-2 (GPT-2 tokenizer, sequence length 1024) with a small model: hidden size 512, 3 layers with 4 energy iterations, 8 experts, intermediate size 256, top-2 routing. Training uses AdamW ($\mathrm{lr} = 3 \times 10^{-4}$, cosine decay) for 1000 steps on a single RTX 4090.

Baseline: log-determinant normalization enabled. With the full GMM logits of Eq. equation 19, the $\frac{1}{2} \log\det(W_e^\top W_e)$ term grows unboundedly during training, reaching auxiliary loss values of $\sim 1.5 \times 10^6$. This drowns the distance-based routing signal, causing **6 out of 8 experts to collapse** (normalized routing entropy ≈ 0.33, language modeling loss stuck at ~ 7.5).

Exp A: disabling log-determinant normalization. Dropping the $\frac{1}{2} \log\det$ term from the routing logits yields a large improvement: LM loss drops from 10.9 to 5.4, and auxiliary loss falls to ~ 18. However, 4 out of 8 experts remain collapsed (normalized entropy ≈ 0.65).

Exp B: strong anti-collapse regularization. Starting from Exp A and increasing penalties by $10\times$ (γ_{ent}=1.0, γ_{mix}=1.0, λ_{rep}=0.1, λ_{div}=0.1) yields the best results: collapsed experts drop from 4 to 2 over the course of training, with normalized entropy rising to 0.80 and two previously dead experts recovering (reaching 6.6% and 7.7% token share respectively). LM loss remains comparable at 5.44, confirming that anti-collapse regularization does not hurt language modeling quality.

Key findings.

1. The log-determinant normalization in Eq. equation 19 is the *primary cause* of expert collapse at this scale and should be disabled or carefully regularized.
2. Strong anti-collapse penalties substantially improve expert utilization ($0.33 \rightarrow 0.80$ normalized entropy) without degrading LM loss.
3. Two experts remain persistently dead even with $10\times$ penalties, suggesting that bias-based balancing alone cannot rescue completely collapsed experts. Periodic re-initialization of dead expert centroids or warm-starting with uniform routing may be needed.

Algorithm 1: Boltzmann Routing with Sparse Execution

Input: $h \in \mathbb{R}^d$; expert weights $\{W_1^{(e)}, W_2^{(e)}\}_{e=1}^{N_e}$; inverse temperature β_r; sparsity k
Output: MoE output $y \in \mathbb{R}^d$
`/* Stage 1: Compute all expert energies */`
for $e = 1, \ldots, N_e$ **do**
  $E_e \leftarrow -\,\mathrm{GELU}(W_1^{(e)} h)^\top (W_2^{(e)} h)$;
end
$\boldsymbol{w}^{\mathcal{B}} \leftarrow \mathrm{softmax}\big(-\beta_r \cdot [E_1, \ldots, E_{N_e}]\big)$;
`/* Stage 2: Sparse top-k execution */`
$\mathcal{S} \leftarrow$ top-k indices of $\boldsymbol{w}^{\mathcal{B}}$;
$\tilde{w}_e \leftarrow w_e^{\mathcal{B}} \big/ \sum_{e' \in \mathcal{S}} w_{e'}^{\mathcal{B}}$ for $e \in \mathcal{S}$;
$y \leftarrow \sum_{e \in \mathcal{S}} \tilde{w}_e \cdot f_e(h)$; // $f_e = -\nabla_h E_e$
return y

C.6 Boltzmann Routing Algorithm

Stage 1 Algorithm 1 requires N_e forward projections ($\mathcal{O}(N_e dm)$), higher than a linear router's $\mathcal{O}(N_e d)$ but reusing expert weight matrices. Stage 2 executes only k expert's full dual-path forward pass. For large expert counts ($N_e \geq 64$), a linear pre-filter can reduce candidates to $2k$ before full energy evaluation.

D Experimental Results

We evaluate five MoE routing strategies within a three-layer architecture: a standard GPT pre-layer (softmax attention + SwiGLU MLP), an Energy Block (energy attention + MoE, iterated 6 times), and a standard GPT post-layer. Hidden size 1024, top-2 routing, trained for 30k steps ($\approx$ 63B tokens) on NemoTron-CC data.

Training details. All models are trained with sequence length 4096. We use AdamW with learning rate 3×10^{-4}, $\beta = (0.9, 0.95)$, $\epsilon = 10^{-10}$, weight decay 0.1, and gradient clipping at 1.0. The learning rate follows a cosine schedule with 2,000 warmup steps, 28,000 decay steps, and a minimum LR factor of 0.1.

Training uses BF16 mixed precision with FSDP (stage 0, algorithm 2). Standard MoE, Stop-Gradient, and Std (no aux) use micro-batch size 8 with 1 gradient accumulation step; Boltzmann Pure and F5 use micro-batch size 2 with 4 gradient accumulation steps (same effective batch size) due to the higher memory cost of evaluating all expert energies. Attention uses 16 heads with multi-head attention (no GQA), RoPE positional embeddings (dimension 64), and RMSNorm normalization ($\epsilon = 10^{-5}$). Word embeddings are tied between input and output.

Routing variants:

- **Standard MoE** : Learned linear gate with aux load-balancing loss ($\lambda_{\text{aux}} = 0.001$).
- **Boltzmann F5** : Full Boltzmann routing (all experts evaluated) with KL distillation to a linear gate; learnable β_r.
- **Boltzmann Pure** : Energy-based routing $w_e = \mathrm{softmax}(-\beta E_e(h))$, no aux loss.
- **Stop-Gradient** : Linear gate with `detach()` on router input; energy-compatible.
- **Standard (no aux)** : Same as Standard MoE with $\lambda_{\text{aux}} = 0$.

Routing metrics. We track four routing diagnostics throughout training:

- **Normalized Entropy** ($H/H_{\max}$, range $[0, 1]$) : Shannon entropy of the routing distribution divided by $\ln E$. A value of 1.0 indicates perfectly uniform routing; 0.0 means only one expert is ever selected. This measures how spread out routing decisions are, independent of expert count.
- **Routing Entropy** ($H = -\sum_e p_e \ln p_e$, range $[0, \ln E]$) : Unnormalized Shannon entropy. The maximum scales with expert count ($\ln 8 \approx 2.08$, $\ln 32 \approx 3.47$), making it useful for comparing absolute information content across configurations.

- **Load Balance** ($E \cdot \min_e(p_e)/\max_e(p_e)$, range $[0, E]$) : Ratio between least- and most-loaded experts, scaled by E. A perfect score of E means equal token counts across all experts; 0 means at least one expert receives no tokens. This captures worst-case imbalance relevant for hardware utilization.
- **Expert Collapse Count** (range $[0, E]$) : Number of experts receiving $<1\%$ of tokens. Collapsed experts are effectively dead, they receive too little gradient signal to recover, creating a self-reinforcing cycle.

D.1 Experiment E2: 8-Expert Baseline

Configuration: 8 experts $\times$ 1024 intermediate, dense SwiGLU $= 4096$ intermediate.

D.1.1 Downstream Evaluation

Table 4: E2: 8-expert downstream results. Lower PPL is better (↓); higher Avg is better (↑). Best in **bold**.

Routing	ARC-C	ARC-E	BoolQ	COPA	HSwag	LAMB.	OBQA	PIQA	RACE	SciQ	Wino.	PPL↓	Avg↑
Standard MoE	0.242	0.469	**0.615**	0.620	0.325	0.192	0.282	0.637	0.266	0.656	**0.534**	40.36	0.440
Boltzmann F5	0.249	0.462	0.541	**0.630**	0.320	0.176	**0.300**	0.637	0.264	0.673	0.511	42.82	0.433
Boltzmann Pure	0.246	0.470	0.608	0.640	**0.327**	0.182	0.290	0.639	0.269	0.667	0.504	41.17	0.440
Stop-Gradient	0.239	0.470	0.560	0.640	0.322	**0.194**	0.274	**0.647**	**0.277**	**0.676**	0.508	**40.17**	0.437
Std (no aux)	**0.249**	**0.475**	0.599	**0.630**	0.326	0.193	0.284	0.642	0.272	0.655	0.518	40.23	**0.440**

At 8 experts, all routing variants achieve comparable accuracy (0.433 to 0.440, Table 4). Stop-Gradient achieves the lowest perplexity (40.17) through near-perfect load balance, while Boltzmann F5 has the highest perplexity (42.82), likely due to the KL distillation objective competing with the language modeling loss. Even Boltzmann Pure, which collapses 4 of 8 experts (Table 5), matches the best accuracy, at this scale the remaining active experts compensate for dead ones.

D.1.2 Routing Metrics

Table 5: E2: Routing metrics at convergence (step 30k). $H_{\max} = \ln 8 \approx 2.08$. Load balance $= E \cdot \min(p_e)/\max(p_e)$; perfect $= 8.0$. Collapsed = experts receiving $<1\%$ of tokens.

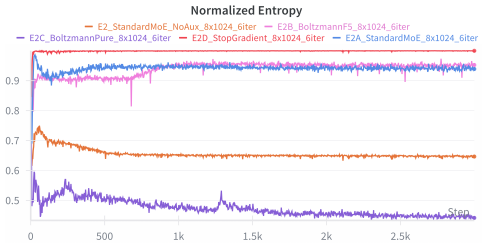
Routing	**Norm. Entropy** (max = 1.0)	**Routing Entropy** (max = 2.08)	**Load Balance** (max = 8.0)	**Collapsed** (out of 8)
Standard MoE	0.939	1.953	2.109	**0**
Boltzmann F5	0.952	1.980	2.207	**0**
Boltzmann Pure	0.442	0.919	0.0	4
Stop-Gradient	**0.999**	**2.079**	**7.581**	**0**
Std (no aux)	0.647	1.345	0.0	3

Key observations (Table 5 and fig. 2). Stop-Gradient achieves near-perfect uniformity (entropy 0.999, load balance $7.58/8.0$) by preventing expert gradients from biasing the router. Boltzmann Pure collapses rapidly, 4 of 8 experts die by step 50 and never recover, as the energy landscape concentrates on a few low-energy attractors in a self-reinforcing cycle. Standard MoE and Boltzmann F5 maintain all experts alive with moderate balance via aux loss. Without aux loss, 3 experts collapse, yet downstream performance is unaffected at this scale.

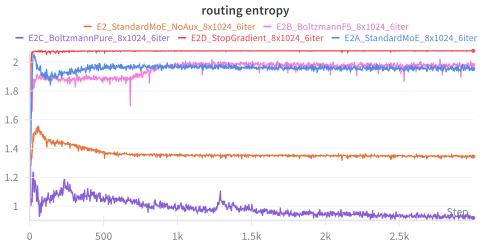
D.2 Experiment E5: Temperature Sweep

We fix the Boltzmann F5 architecture at 8 experts and sweep the inverse temperature $\beta_r \in \{0.1, 1.0, 2.0, 5.0\}$ (fixed, not learned). Lower β_r corresponds to softer (more uniform) routing; higher β_r to sharper (more peaked) routing.

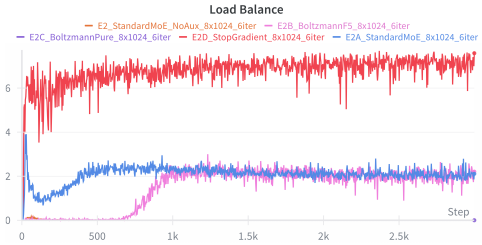
Moderate sharpness ($\beta_r = 2.0$) achieves the best average accuracy (0.437), while sharper routing ($\beta_r = 5.0$) gives the lowest perplexity (42.06) at some cost to accuracy (Table 6). The softest setting ($\beta_r = 0.1$) produces the worst perplexity (45.05), confirming that near-uniform routing



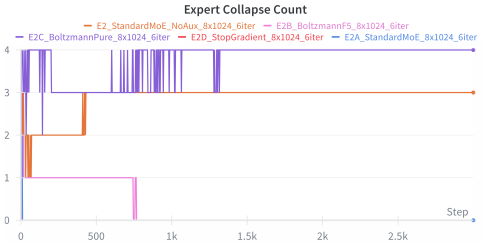
(a) Normalized Entropy



(b) Routing Entropy



(c) Load Balance



(d) Expert Collapse Count

Figure 2: E2: Routing dynamics over 30k training steps (8 experts). Stop-Gradient (teal) converges to near-perfect uniformity. Boltzmann Pure (orange) collapses irreversibly by step 50. Standard MoE and F5 stabilize at moderate entropy with aux loss.

Table 6: E5: Temperature sweep. 8 experts $\times$ 1024, Boltzmann F5, fixed β_r, 30k steps.

β_r	$T = 1/\beta_r$	ARC-C	ARC-E	HSwag	BoolQ	PPL↓	Avg↑
0.1 (soft)	10.0	**0.253**	0.447	0.319	**0.584**	45.05	0.435
1.0	1.0	0.241	0.451	0.319	0.564	43.45	0.424
2.0	0.5	0.247	**0.473**	0.319	0.571	42.23	**0.437**
5.0 (sharp)	0.2	0.237	0.464	0.318	0.486	**42.06**	0.428
Learned	$\rightarrow$ 0.17	*0.249*	*0.462*	*0.320*	*0.541*	*42.82*	*0.433*

wastes expert capacity. The learned β_r (E2-B) converges to ≈ 0.17, near the soft end of our sweep, suggesting that optimizer dynamics or weight decay bias the learned temperature toward uniformity rather than the optimal operating point.

D.3 Experiment E6: 32-Expert Scaling

Configuration: 32 experts $\times$ 2048 intermediate, dense SwiGLU $=$ 4096 intermediate. The dense layers still carry most of the per-token compute, which may mask routing quality differences. We include two Boltzmann F5 variants: one with $\beta_r = 0.1$ initialization and one with default $\beta_r = 1.0$ (compiled).

D.3.1 Downstream Evaluation

Table 7: E6: 32-expert results. Dense $=$ 4096, MoE $= 32 \times 2048$, top-2, 30k steps.

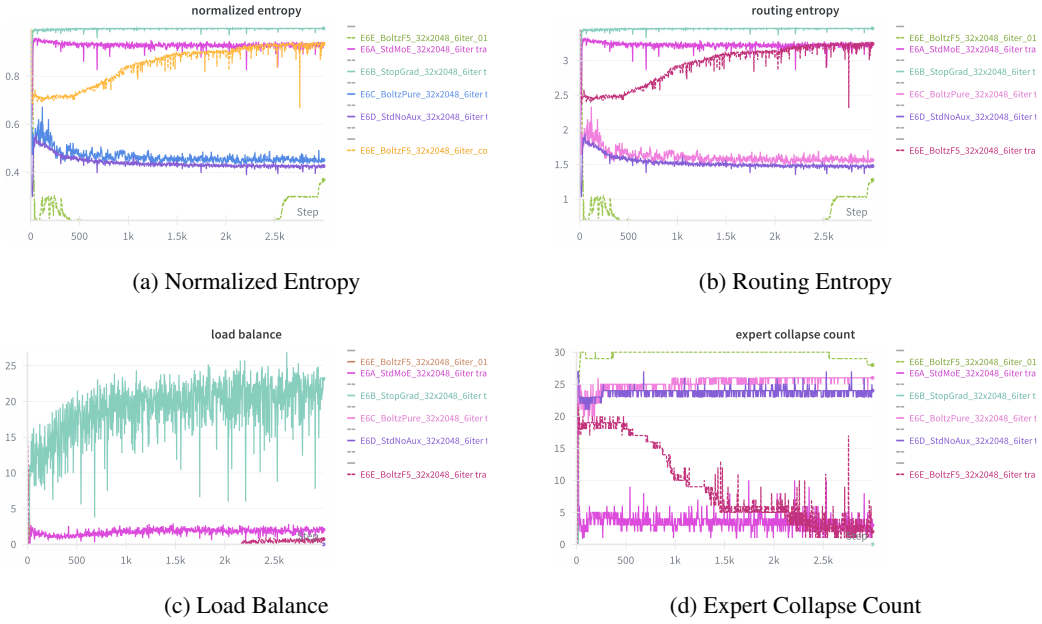
Routing	ARC-C	ARC-E	BoolQ	COPA	HSwag	LAMB.	OBQA	PIQA	RACE	SciQ	Wino.	PPL↓	Avg↑
Standard MoE	0.249	**0.492**	0.588	**0.660**	0.351	**0.203**	**0.316**	**0.667**	0.261	0.683	0.513	**35.27**	**0.453**
Stop-Gradient	**0.271**	0.482	0.575	0.620	0.338	**0.203**	0.300	0.650	0.269	**0.687**	0.504	36.84	0.445
Boltzmann Pure	0.242	0.442	0.465	0.650	0.303	0.165	0.300	0.631	0.257	0.656	0.510	46.18	0.420
Std (no aux)	0.256	**0.492**	**0.614**	0.620	**0.353**	0.201	0.284	0.651	**0.288**	0.677	**0.522**	35.51	0.451
F5 ($\beta_r = 0.1$)	0.253	0.465	0.541	0.600	0.315	0.150	0.290	0.618	0.268	0.661	0.511	62.14	0.425
F5 (compiled)	0.242	0.472	0.611	0.660	0.334	0.157	0.312	0.638	0.262	0.651	0.525	42.58	0.442

Standard MoE achieves the lowest perplexity (35.27) and highest accuracy (0.453, Table 7). Stop-Gradient, which led at 8 experts, now drops to third (36.84 PPL, 0.445 avg), uniform routing loses its advantage when expert differentiation matters more. Boltzmann Pure degrades sharply (46.18 PPL) due to catastrophic expert collapse. Notably, F5 with $\beta_r = 0.1$ performs worst on perplexity (62.14), while the compiled F5 with default $\beta_r = 1.0$ is reasonable (42.58), highlighting the sensitivity of Boltzmann routing to temperature initialization at higher expert counts.

D.3.2 Routing Metrics

Table 8: E6: Routing metrics at convergence. $H_{\max} = \ln 32 \approx 3.47$. Load balance max $= 32.0$.

Routing	**Norm. Entropy** (max $= 1.0$)	**Routing Entropy** (max $= 3.47$)	**Load Balance** (max $= 32.0$)	**Collapsed** (out of 32)
Standard MoE	0.935	3.240	2.046	3
Stop-Gradient	**0.999**	**3.463**	**23.163**	**0**
Boltzmann Pure	0.451	1.563	0.0	26
Std (no aux)	0.426	1.476	0.0	24
F5 ($\beta_r = 0.1$)	0.368	1.277	0.0	28
F5 (compiled)	0.934	3.237	0.734	2



(a) Normalized Entropy

(b) Routing Entropy

(c) Load Balance

(d) Expert Collapse Count

Figure 3: E6: Routing dynamics over 30k steps (32 experts, dense-dominated). Stop-Gradient maintains perfect balance throughout. Boltzmann Pure, F5 ($\beta_r = 0.1$), and Std (no aux) all suffer severe collapse ($\geq$24/32 experts dead). Only Standard MoE and F5 (compiled) keep most experts alive.

Key observations (Table 8 and fig. 3). Scaling to 32 experts dramatically changes the collapse landscape. Std (no aux), which only lost 3/8 experts in E2, now loses 24/32, the combinatorial space at 32 experts is insufficient for self-organization without explicit balancing. Despite this, Std (no aux) achieves near-identical perplexity to Standard MoE (35.51 vs. 35.27), suggesting that the dense layers absorb most of the modeling burden. Boltzmann Pure (26 collapsed) and F5 $\beta_r = 0.1$ (28 collapsed) show even worse collapse, yet their perplexity gap with Std (no aux) (24 collapsed) is dramatic: 46.18 and 62.14 vs. 35.51. This reveals that **collapse count alone does not determine downstream performance**; the quality of remaining active experts and the effectiveness of token-to-expert assignment matter critically.

D.4 Experiment E9: MoE-Focused Architecture

Configuration: 32 experts $\times$ 1024 intermediate, dense SwiGLU $= 512$ intermediate. By shrinking the dense layers, the MoE layer becomes the dominant source of model capacity, amplifying routing quality differences.

D.4.1 Downstream Evaluation

Table 9: E9: 32-expert, MoE-focused. Dense $= 512$, MoE $= 32 \times 1024$, top-2, 30k steps.

Routing	ARC-C	ARC-E	BoolQ	COPA	HSwag	LAMB.	OBQA	PIQA	RACE	SciQ	Wino.	PPL↓	Avg↑
Standard MoE	0.235	**0.455**	**0.618**	0.590	0.320	**0.190**	0.284	0.634	0.257	**0.659**	0.519	42.30	**0.433**
Stop-Gradient	0.233	0.441	0.586	**0.640**	0.313	0.183	0.270	**0.645**	**0.266**	0.646	0.499	43.48	0.429
Boltzmann Pure	0.218	0.409	0.575	0.590	0.286	0.111	0.274	0.596	0.251	0.614	0.516	68.72	0.404
Std (no aux)	**0.238**	0.463	0.591	0.580	**0.322**	0.181	**0.294**	0.634	**0.266**	0.665	0.511	**42.23**	0.431
Boltzmann F5	0.236	0.426	0.622	0.560	0.298	0.158	0.278	0.612	0.254	0.648	**0.522**	57.53	0.419

With the MoE-focused architecture, routing quality differences are amplified (Table 9). Standard MoE and Std (no aux) achieve comparable perplexity (42.30 and 42.23), both substantially lower than Boltzmann Pure (68.72) and F5 (57.53). Stop-Gradient (43.48 PPL) falls behind both standard variants, reinforcing the trend from E6. Boltzmann Pure's perplexity is now 63% worse than Standard MoE, compared to 31% in E6.

D.4.2 Routing Metrics

Table 10: E9: Routing metrics at convergence. $H_{\max} = \ln 32 \approx 3.47$. Load balance max $= 32.0$.

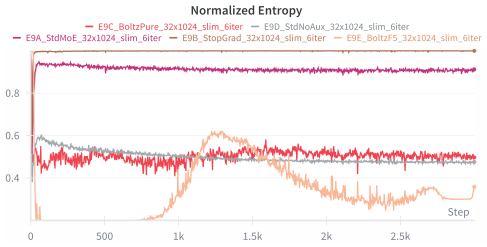
Routing	**Norm. Entropy** (max $= 1.0$)	**Routing Entropy** (max $= 3.47$)	**Load Balance** (max $= 32.0$)	**Collapsed** (out of 32)
Standard MoE	0.912	3.160	0.984	6
Stop-Gradient	**1.000**	**3.464**	**25.583**	**0**
Boltzmann Pure	0.497	1.721	0.0	22
Std (no aux)	0.475	1.647	0.0	23
Boltzmann F5	0.358	1.242	0.0	27

Key observations (Table 10 and fig. 4). The MoE-focused architecture exposes a striking pattern: Standard MoE now collapses 6 experts (up from 3 in E6), suggesting that reduced dense capacity forces more pressure onto the MoE layer, making it harder to maintain expert diversity even with aux loss. Boltzmann F5 collapses worst (27/32), while Boltzmann Pure (22/32) and Std (no aux) (23/32) are similarly severe. Again, collapse count does not fully predict performance: Std (no aux) has 23 collapsed experts yet achieves 42.23 PPL, while Boltzmann Pure with 22 collapsed achieves much worse 68.72 PPL. The surviving experts in Std (no aux) appear to specialize more effectively.

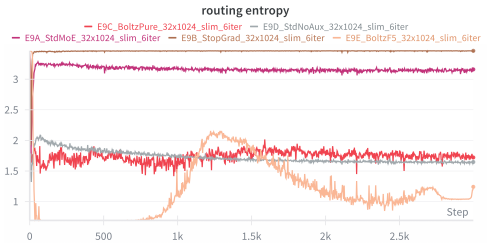
D.5 Cross-Scale Analysis

Table 11: Summary across scales. WikiText PPL (↓) and average accuracy (↑). Best per column in **bold**.

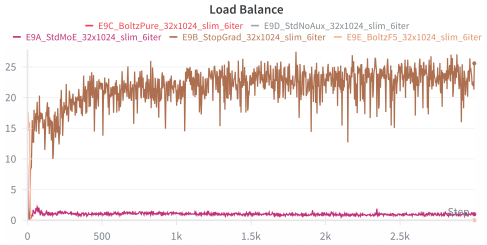
	8×1024 (E2)		**32×2048 (E6)**		**32×1024 slim (E9)**	
Routing	PPL↓	Avg↑	PPL↓	Avg↑	PPL↓	Avg↑
Standard MoE	40.36	**0.440**	**35.27**	**0.453**	42.30	**0.433**
Stop-Gradient	**40.17**	0.437	36.84	0.445	43.48	0.429
Boltzmann Pure	41.17	**0.440**	46.18	0.420	68.72	0.404
Std (no aux)	40.23	**0.440**	35.51	0.451	**42.23**	0.431
Boltzmann F5	42.82	0.433	42.58	0.442	57.53	0.419



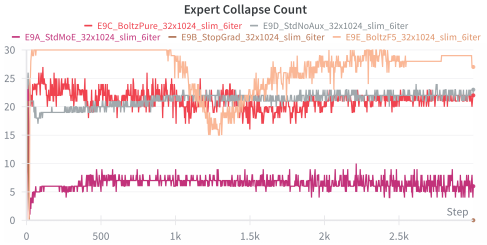
(a) Normalized Entropy



(b) Routing Entropy



(c) Load Balance



(d) Expert Collapse Count

Figure 4: E9: Routing dynamics over 30k steps (32 experts, MoE-focused). The MoE-focused architecture amplifies collapse: Standard MoE now loses 6/32 experts (vs. 3/32 in E6), while Boltzmann F5 collapses catastrophically (27/32). Stop-Gradient remains the only variant with zero collapse.

Table 12: Expert collapse across scales. Number of collapsed experts (<1% tokens) at step 30k.

Routing	**E2 (8 experts)**	**E6 (32 experts)**	**E9 (32 experts slim)**
Standard MoE	0	3	6
Stop-Gradient	**0**	**0**	**0**
Boltzmann Pure	4	26	22
Std (no aux)	3	24	23
Boltzmann F5	0	2	27

D.5.1 Discussion

Finding 1: Stop-Gradient degrades at scale. At 8 experts, Stop-Gradient achieves the lowest perplexity (40.17) and perfect load balance (Table 11). At 32 experts it rises to 36.84 PPL (3rd place), and in the MoE-focused configuration to 43.48 PPL (3rd again). Its near-perfect load balance masks a lack of meaningful specialization: when expert count grows and the model relies more on the MoE layer, uniform-but-content-agnostic routing becomes a liability.

Finding 2: Boltzmann Pure collapse worsens with scale. Perplexity increases from 41.17 (8 experts) to 46.18 (32 experts) to 68.72 (32 experts, MoE-focused, Table 11). Without load balancing, the energy landscape funnels most tokens to a few low-energy attractors, leaving the majority of experts unused. This is tolerable at 8 experts (4/8 collapsed) and catastrophic at 32 (22 to 26 collapsed, Table 12).

Finding 3: Collapse count does not determine performance. Perhaps the most surprising result (Tables 11 and 12): Std (no aux) with 24 collapsed experts in E6 achieves 35.51 PPL, while Boltzmann Pure with 26 collapsed achieves 46.18 PPL, a 30% gap despite only 2 more dead experts. In E9, Std (no aux) with 23 collapsed achieves 42.23 PPL, while Boltzmann Pure with *fewer* collapsed (22) achieves 68.72 PPL, 63% worse. The critical factor is the **quality of the surviving experts**: a learned gate (even without aux loss) produces better-specialized active experts than energy-based routing, which tends to concentrate on suboptimal low-energy states.

Finding 4: Aux loss matters more in MoE-focused architectures. Standard MoE collapses 0 experts at 8 experts, 3 at 32 experts (E6), and 6 at 32 experts MoE-focused (E9, Table 12). As the MoE layer bears more of the computational burden, maintaining expert diversity becomes harder. Yet Boltzmann F5 (compiled) manages just 2 collapsed experts in E6 with reasonable performance (42.58 PPL), suggesting that the Boltzmann routing mechanism with proper temperature initialization can prevent collapse.

Finding 5: Dense layers mask routing quality. Moving from E6 to E9 widens the perplexity gap between Standard MoE and Boltzmann Pure from 11 points to 26 points (Table 11). When dense layers are large, they absorb most of the modeling burden and mask MoE routing differences. The MoE-focused architecture isolates these differences, making it a better testbed for routing research.

Finding 6: Temperature initialization is critical at scale. In E6, F5 with $\beta_r = 0.1$ (soft initialization) collapses 28/32 experts and achieves 62.14 PPL, while F5 with $\beta_r = 1.0$ collapses only 2/32 and achieves 42.58 PPL (Tables 7 and 8). At higher expert counts, initializing with too-soft routing delays expert differentiation, leading to a collapse cascade during early training from which the model never recovers.