THE MOVING FRONTIER OF NATIVE SPEECH–LANGUAGE PRETRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal large models are typically built through a late-fusion recipe: language models and modality-specific components are pretrained separately, then connected and jointly trained. This approach builds on advances in unimodal pretraining and has become a standard route to multimodal capabilities. Recently, frontier systems have begun to explore a different recipe, learning perception and language jointly from scratch. These developments suggest that native multimodal pretraining is a viable alternative and motivate a closer examination of the principles governing its scaling. In this work, we study this paradigm through continuous speech and text, using speech–language modeling as a concrete setting in which to investigate how model capacity, training compute, and modality-specific supervision interact. Across over 140 completed configurations spanning 149M–2.72B parameters, we demonstrate that randomly initialized models can jointly acquire speech recognition and text-modeling capabilities. We characterize how scale, capacity placement, and supervision shape speech and text learning, and use boundary and initialization controls to probe these relationships. A scaling law connects these empirical findings to loss prediction across configurations. Together, these results provide an empirical foundation for native speech–language pretraining and a quantitative guide to allocating its training resources.

1 INTRODUCTION

Learning across modalities now supports visual understanding and the generation of images, audio, and video. Much of this progress builds on reusable pretrained components: vision–language systems connect image encoders with language models (Li et al., 2023), while image, music, and video generators build on learned perceptual representations or pretrained backbones (Rombach et al., 2022; Copet et al., 2023; Blattmann et al., 2023). This modular strategy transfers advances in individual modalities into broader capabilities. For multimodal language models, separately pretrained components are then connected and aligned, as in visual and audio–language systems (Li et al., 2023; Tang et al., 2024; Chu et al., 2024). Following the early-/late-fusion distinction in prior multimodal work (Chameleon Team, 2024), we use *late-fusion training* for this pretrain-then-integrate recipe, distinguishing training order from the architectural location of fusion.

Recent frontier models suggest an alternative. MiniMax M3 reports mixed-modality training from step zero, while Kimi K3 jointly optimizes language and vision from the start and trains its vision encoder from scratch (MiniMax, 2026; Kimi Team, 2026). These reports motivate native training as a practical research direction. Its scaling problem extends beyond selecting a model size: finite computation must be divided between perceptual capacity, language capacity, and modality-specific exposure. Emerging native multimodal studies examine architectural and data-mixture choices (Shukor et al., 2025b; Wu et al., 2026). We investigate how these choices jointly determine the capabilities learned from continuous speech and text.

Continuous speech exposes this allocation problem particularly clearly. Acoustic perception and language modeling share a decoder, but have different capacity and computational demands: speech introduces long continuous sequences that must be contextualized and compressed, whereas text directly supervises language prediction. Increasing an encoder, enlarging a decoder, and supplying more speech therefore spend compute in different ways. Total parameter count and aggregate data volume alone need not describe the resulting capability balance.

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.

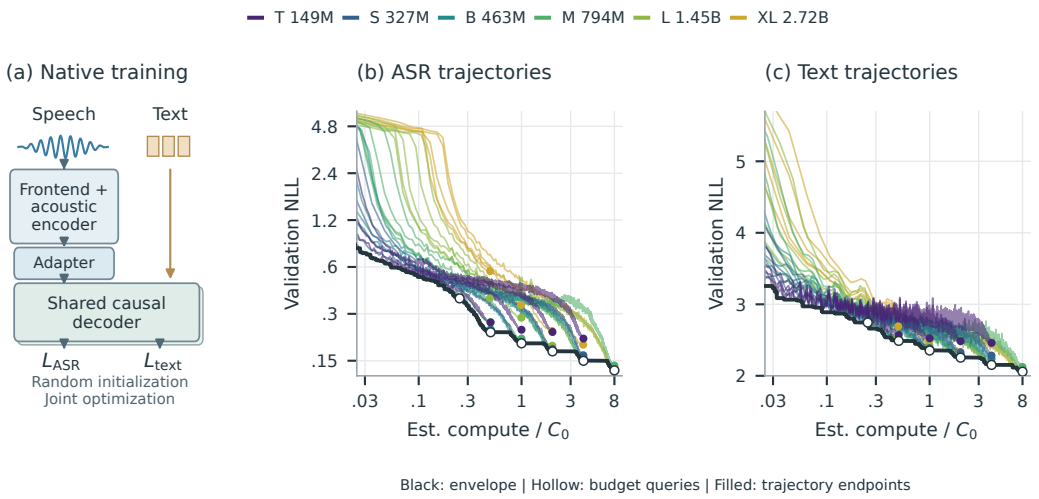


Figure 1: **Native training and its observed scaling envelope.** A randomly initialized acoustic path and shared language decoder learn from speech and text. Right: 27 completed balanced-size trajectories; colors denote total model size and black traces the best recorded checkpoint within estimated compute. Intermediate checkpoints are not independent short-budget runs. Visualization details are given in Appendix A.

We train the acoustic frontend, encoder, adapter, and language decoder jointly from random initialization, using speech-conditioned transcript prediction and text language modeling (Figure 1). This is a native training recipe despite retaining a distinct acoustic encoder. Having established feasibility under this protocol, we use scaling relations to organize three questions: *How should model size change with the compute budget? Where should capacity and data exposure be allocated? Can the resulting speech and text losses be predicted for configurations not used in fitting?* Independent training horizons and controlled allocation, mixture, and shape comparisons separate these questions as far as the experimental design permits.

Our experiments reveal several aspects of native learning: useful model sizes change with compute, acoustic capacity and speech supervision have non-monotonic effects, and text learning can support recognition. Boundary and initialization controls examine acoustic context and pretrained priors. We use scaling laws to organize these observations into a quantitative account of capacity–exposure trade-offs, with held-out prediction testing its generalization. At a specified arithmetic budget, nondominated response pairs define a speech–language *frontier*, connecting learning behavior to configuration choices.

Our contributions are threefold. **A systematic study of native pretraining:** we establish joint speech recognition and text modeling from random initialization and study scale, capacity allocation, supervision mixture, and complementary architectural and initialization controls. **Empirical learning relationships:** we reveal non-monotonic capacity–supervision effects, shifting scale preferences, and the roles of acoustic context and shared language learning. **A quantitative scaling account:** we model acoustic, decoder, and embedding capacity alongside modality-specific exposure, test loss prediction on held-out configurations, and examine its value for recognition-based selection. Together, these experiments characterize native speech–language learning and connect its empirical regularities to model design.

2 Related Work

Native and assembled multimodal learning. Joint speech–text representation learning includes mSLAM and SpeechT5 (Bapna et al., 2022; Ao et al., 2022), while Whisper demonstrates the value of large-scale speech supervision (Radford et al., 2023). SALMONN and Qwen2-Audio integrate pretrained speech and language components (Tang et al., 2024; Chu et al., 2024). Chameleon studies early-fusion image–text modeling with discrete representations (Chameleon Team, 2024); dMel and

encoder-free spectrogram models explore alternative speech interfaces (Bai et al., 2024; Fan et al., 2026). Our focus is joint learning from continuous speech and text with random trainable weights, and the resource trade-offs that arise within this setting.

Scaling, capacity allocation, and data mixture. Language-model scaling links loss to model size, data, and compute (Kaplan et al., 2020; Hoffmann et al., 2022). Encoder–decoder studies show that capacity location and data quality also matter (Ghorbani et al., 2022; Bansal et al., 2022). Native multimodal scaling extends these questions to vision–language training (Shukor et al., 2025b; Wu et al., 2026), and SODA studies interleaved discrete audio and text (Manakul et al., 2026). DoReMi optimizes domain weights through proxy models (Xie et al., 2023); data-mixing laws predict responses across mixtures, scales, and budgets (Ye et al., 2025; Shukor et al., 2025a). Data-constrained scaling further separates new exposure from repeated data (Muennighoff et al., 2023). We combine these perspectives by measuring acoustic, decoder, and embedding capacity alongside actual modality exposure, then evaluating prediction across configurations and budgets.

3 Experimental setup

3.1 Model and training objective

We study continuous speech and text in a shared autoregressive model. A convolutional frontend maps 128-bin mel features to a 12.5 Hz sequence; a bidirectional Transformer encoder (Vaswani et al., 2017) and adapter map this sequence into decoder space. The causal Prefix decoder predicts transcript tokens after the acoustic prefix and models text without that prefix. All trainable weights start randomly. We use a fixed Qwen3 tokenizer and tied input/output embeddings.

Each update optimizes a weighted sum of globally token-normalized objectives,

$$\mathcal{L} = \lambda_{\text{text}} L_{\text{text}} + \lambda_{\text{asr}} L_{\text{asr}}, \qquad L_m = -\frac{1}{n_m} \sum_{i=1}^{n_m} \log p_\theta(w_i \mid w_{<i}, c_m), \tag{1}$$

where c_m is the acoustic prefix for ASR and empty for text. We set $\lambda_{\text{text}} = \lambda_{\text{asr}} = 1$ in all joint-training experiments; equal weights do not imply equal token exposure. We report both responses and their fixed-weight sum, retaining the speech–language trade-off.

Internet speech–transcript pairs offer a scalable source of supervision beyond fixed benchmark training sets. Speech training uses an internal Internet speech–transcript pool, with clips of at most 30 seconds and no added AISHELL, LibriSpeech, or other public ASR training splits. Text sources include DCLM (Li et al., 2024), Nemotron (NVIDIA et al., 2025), and Dolma-derived corpora (Team Olmo et al., 2025; Soldaini et al., 2024); Appendix D specifies releases. Each independent horizon uses AdamW (Loshchilov & Hutter, 2019), peak learning rate 6×10^{-4}, 3% warmup, and full cosine decay. The balanced scale path uses one text and one ASR microbatch per update; mixture experiments vary modality-specific packing budgets.

3.2 Capacity, exposure, and compute

We distinguish acoustic parameters P_a (frontend, encoder and adapter), decoder non-embedding parameters P_d, and tied embedding parameters P_e. Total size is $P_a + P_d + P_e$; non-embedding size is $N_{\text{core}} = P_a + P_d$. Cumulative supervised text and transcript targets are T, Y. Capacity allocation and data mixture can therefore be expressed as

$$q_{\text{ac}} = \frac{P_a}{P_a + P_d}, \qquad x_{\text{asr}} = \frac{Y}{T + Y}. \tag{2}$$

Neither ratio is an audio-frame or FLOP fraction. Balanced models allocate roughly equal non-embedding capacity to the acoustic path and decoder. The six sizes T, S, B, M, L, and XL have approximately 149M, 327M, 463M, 794M, 1.45B, and 2.72B total parameters. Model-size axes use total parameters.

We use the same frozen operator-counting convention across runs:

$$C_{\text{op}} = \sum_{u=1}^{U} \widehat{F}(\theta_{\text{arch}}, Q_{t,u}, Q_{a,u}, Z_{a,u}, M_{a,u}), \tag{3}$$

where Q_t, Q_a are decoder positions, Z_a encoder positions and M_a mel frames. The profiler counts two FLOPs per multiply–accumulate and approximates training as three times forward arithmetic, including vocabulary operations (Kaplan et al., 2020; Hoffmann et al., 2022). It excludes activation recomputation and execution overhead; C_{op} is not GPU-hours. The reference B run defines $C_0 = 3.6173 \times 10^{19}$ FLOPs. Matching capacity, exposure and compute are different controls; each comparison states which quantities are matched.

3.3 Evaluation and analysis population

We fit ASR and text losses measured on shared, fixed validation sets at the end of each training schedule. The main results evaluate pretrained checkpoints directly on AISHELL-1 (Bu et al., 2017) and duration-filtered LibriSpeech clean/other (Panayotov et al., 2015), without benchmark-specific training or fine-tuning. Text evaluation uses a fixed 22-task DCLM-style suite (Li et al., 2024). These metrics complement the fitted validation NLLs. Appendix E gives the evaluation protocol; Appendix N separately studies downstream adaptation.

The experimental program covers scale, capacity allocation, modality mixture, architecture, and initialization. These experiments serve different scientific questions; their scope is broader than the regression population. The primary five-factor analysis uses 116 cold joint-training configurations, with encoder-free/text-only and shape controls analyzed separately. Low-budget runs remain part of the study of learning dynamics; the nominal $0.25C_0$ slice is outside the primary fitting domain. Appendix A gives the domain definition and matched-test comparisons with the broader budget range.

4 Scaling native speech–language pretraining

4.1 Model scale under a fixed compute budget

Native joint training yields useful speech recognition and text modeling without pretrained trainable components. The balanced M $8C_0$ model reaches 4.24% AISHELL CER and 2.59%/6.50% LibriSpeech clean/other WER with direct pretraining-to-test transfer and no benchmark-specific fine-tuning (Table 10, Z001).

Figure 1 motivates the scaling question: different model sizes contribute to the observed loss envelope at different budgets. Figure 2 tests that picture using independently trained horizons. The displayed scale grid contains 24 endpoints at 0.5–$4C_0$, with all six sizes represented at every displayed budget. At $0.5C_0$, balanced L has ASR NLL 0.3774 versus 0.2644 for T, whereas at $4C_0$ the ordering reverses to 0.1549 versus 0.2081. The useful scale region therefore depends on budget. ASR and text need not select the same configuration, and sparse size coverage does not identify a precise continuous optimum.

The training trajectories also reveal a transition in low-exposure speech learning: several large-model runs remain at high ASR loss before a rapid decrease, consistent with the acquisition of speech–language alignment (Figure 1). This behavior motivates analyzing low-budget dynamics separately from the smoother regime modeled by the scaling law. The broader-domain and exposure-cutoff comparisons in Appendix A quantify how this choice affects predictive performance.

Parameter size alone does not specify the experiment. Width and depth co-vary on the balanced path, and a larger model sees fewer training targets at the same compute. Allocation and mixture interventions are needed to separate these influences.

4.2 Allocating acoustic and language capacity

The L allocation sweep compares five encoder–decoder configurations at four common compute references, keeping non-embedding capacity approximately fixed and transcript-target share near 17% (Figure 3, left). More acoustic capacity does not monotonically improve ASR, while decoder-heavy models generally achieve lower text loss. At $1C_0$, q-third and q-two-thirds (approximately one-third and two-thirds acoustic non-embedding capacity) obtain ASR NLL 0.2530/0.2535, compared with 0.2834 for the balanced configuration. The observed allocation profile is therefore not monotonic.

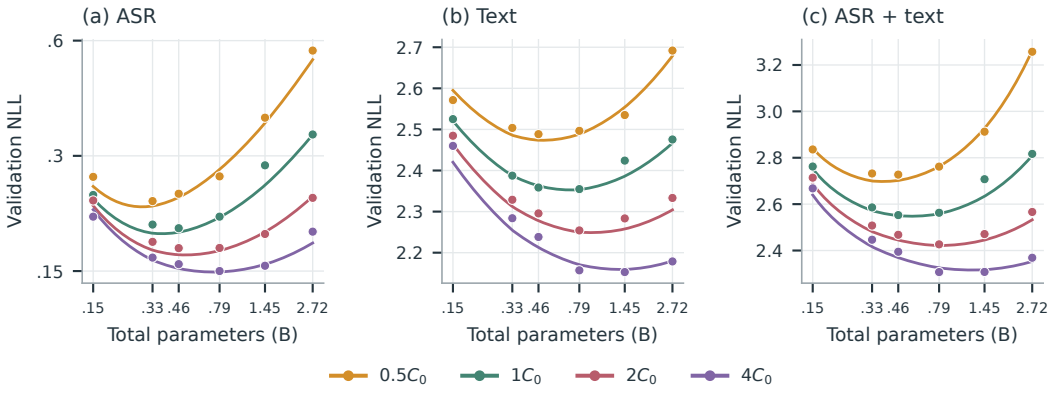


Figure 2: **Independent IsoFLOP endpoints and the fitted law.** Dots are balanced-family observations; lines evaluate the five-factor law fitted on all 116 retained rows. ASR NLL uses a logarithmic axis; text and the fixed-weight total use linear axes. These are full-fit illustrations, not held-out prediction scores. Curves follow log-interpolated input paths within measured size ranges; no $0.25C_0$ predictions are shown.

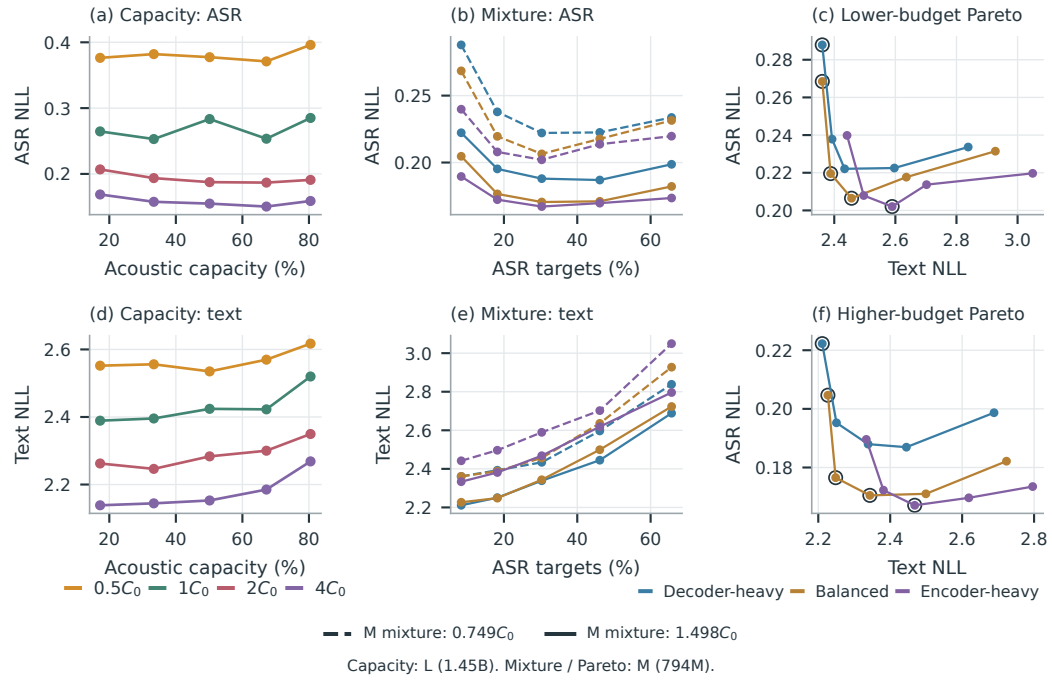


Figure 3: **Capacity and data allocation jointly shape the trade-off.** Left: 20 L allocation endpoints at four compute budgets. Middle: all 30 M allocation–mixture cells at two references, approximately $0.749C_0$ (dashed) and $1.498C_0$ (solid). Right: their ASR–text responses; rings mark nondominated measured cells within each budget. Colors have the column-specific meanings shown. Lines connect measurements, not fitted allocation or mixing laws.

At $4C_0$, the balanced and q-two-thirds results are close in ASR, but their text losses differ. Thus the data support a budget- and objective-dependent allocation choice rather than a fixed optimal encoder fraction. Earlier exposure-matched allocation and dedicated shape controls provide complementary evidence, but are not pooled with this compute-matched sweep (Appendix G).

4.3 Allocating speech and text exposure

The complete M grid crosses three capacity allocations with five transcript-target shares at each of two approximately matched compute references. Increasing speech share initially improves ASR but increases text loss (Figure 3, middle). At the lower reference, the central share has the lowest observed ASR NLL within all three allocations; the decoder-heavy 30% and 46% points are nearly tied, while the highest-share endpoint worsens both losses. At the higher reference, some responses flatten near the center. The pattern supports a useful interior region, rather than a precisely identified universal optimum.

The two-response view makes the decision consequence explicit (right). Raising the compute budget improves the attainable losses, while the preferred choice within a budget depends on the desired speech–language balance. Allocation and mixture must therefore be considered together. The B mixture experiments likewise show a high-speech point changing from dominated to nondominated across two references (Figure 9 in Appendix P); these distinct slices do not establish a universal optimal speech fraction.

4.4 Boundary cases: acoustic context and shared language capacity

Removing an encoder or a supervision source tests what the interior allocation trends cannot establish alone. We examine three complementary boundaries; full controlled comparisons appear in Appendix I.

Encoder-free does not mean context-free. Without a deep acoustic encoder, the decoder must model the audio prefix itself. Allowing audio tokens within each sample to attend bidirectionally, while keeping transcript tokens causal, reduces AISHELL CER by 16–18% relative to a causal-prefix implementation across three training horizons. At 125k updates, CER falls from 5.64% to 4.74% and LibriSpeech other WER from 11.23% to 9.11%; LAMBADA NLL changes by less than 1%. A dedicated encoder is therefore not the only way to supply useful acoustic context. The comparison evaluates the bidirectional-prefix implementation, including its attention backend.

Text supervision can also help speech. The mixture curves show that allocating more supervision to speech is not always beneficial. Cold ASR-only controls provide a complementary endpoint: at fixed architecture and ASR exposure, adding text to the B balanced model reduces other WER from 14.18% to 12.67% and LAMBADA NLL from 11.21 to 3.71. This adds text compute; together with the compute-matched mixture sweep, it supports retaining text supervision rather than maximizing the speech fraction.

Does joint learning displace text modeling? A text-only anchor fixes the shared language capacity: the same 299M decoder, 8.19B text targets, and 125k updates. Its text NLL is 2.3322, versus 2.3586 and 2.3210 for text1:ASR1 and text1:ASR2 joint schedules. Thus speech learning coexists with largely preserved text likelihood at fixed decoder capacity and text exposure. The acoustic path adds parameters and compute; this is a same-decoder comparison, not an equal-total-budget claim.

5 A quantitative scaling account

5.1 A five-factor response model

We model each response using acoustic capacity, decoder non-embedding capacity, embedding capacity and the two target exposures:

$$x = (P_a/10^8,\ P_d/10^8,\ P_e/10^8,\ T/10^9,\ Y/10^9). \tag{4}$$

Allocation and mixture enter through their components, rather than as duplicated regressors. For tied embeddings, $P_e = Vd$, where V is the model vocabulary size and d the decoder width. Our Qwen3-tokenized models use $V = 151{,}936$ embedding entries, held fixed across runs. Embeddings account for approximately 11–52% of total parameters along our balanced scale path. We therefore distinguish embedding from decoder capacity to represent this substantial share explicitly, and assess its incremental predictive value through ablations. Updates and encoder presence are not additional inputs in this joint-training domain.

Table 1: **Predictive performance under shared data splits.** RMSE on the same outer test rows within each protocol; all models use the corresponding training rows. Model complexity is not matched.

	Grouped			Size-out			Forward budget		
Model	ASR	Text	Total	ASR	Text	Total	ASR	Text	Total
Two-factor power	0.0444	0.1278	0.1228	0.1664	0.1275	0.1201	0.0335	0.0736	0.0518
Five-factor power	0.0160	0.0361	0.0418	0.0161	0.0352	0.0419	0.0067	0.0283	0.0251
Matérn-5	0.0214	0.0405	0.0553	0.0251	0.0630	0.0835	0.0204	0.1116	0.1285
Power + NTK	0.0243	0.0535	0.0591	0.1624	0.0727	0.0821	0.0159	0.0398	0.0258

For ASR, text and the fixed-weight total, we fit separate five-term power laws,

$$\widehat{L}_m(x) = E_m + \sum_{j=1}^{5} A_{m,j} x_j^{-\alpha_{m,j}}. \tag{5}$$

We constrain $E, A \geq 0$ and $\alpha \in [0.001, 3]$, minimizing log-loss residuals. This empirical decomposition organizes capacity and exposure trade-offs. At fixed compute, increasing one component can reduce another component or the affordable exposure; thus monotone coordinate terms can yield non-monotone allocation profiles along feasible paths. Controlled comparisons and held-out tests assess this account, rather than assigning a separate causal meaning to each exponent. The two-factor baseline is

$$L_{0,m}(N, D) = E_m + A_m(N/10^8)^{-\alpha_m} + B_m(D/10^9)^{-\beta_m}, \quad N = P_a + P_d,\ D = T + Y. \tag{6}$$

We also compare linear, Matérn and NTK predictors on the five inputs, and the hybrid

$$\widehat{L}_m(x) = L_{0,m}(N, D) \exp g_m(x), \tag{7}$$

with an NTK log-residual model (Jacot et al., 2018). All estimators receive the same training rows. Separate response fits do not force predicted total loss to equal the sum of the two separately predicted components.

5.2 Held-out prediction and scope

We use three complementary tests. *Grouped five-fold* validation holds out geometry–budget groups, keeping every mixture at the same geometry and budget together. *Size-out* validation excludes an entire embedding-size stratum and all its allocations and mixtures. *Forward-budget* validation trains at up to approximately $2C_0$ and tests thirteen $4C_0/8C_0$ endpoints. Kernel hyperparameters and preprocessing are selected only within the outer training fold. Figure 4 and Table 1 report predictions that exclude each scored response, unlike the full-fit curves in Figure 2.

The five-factor power law gives ASR RMSE 0.01600/0.01613/0.00675 across grouped, size-out and forward-budget tests, versus 0.0444/0.1664/0.0335 for the two-factor baseline. Its mean absolute percentage errors are 4.17%/4.37%/3.32% for ASR and 1.11%/1.09%/0.87% for text, using observed loss as the denominator. Text and total also improve over the two-factor baseline in RMSE. This comparison changes model flexibility as well as factorization, so it does not isolate the causal value of any one input. In particular, embedding and other capacity coordinates remain correlated; accurate predictions do not establish independently identifiable exponents.

Input ablations show that the largest predictive gains come from separating modality exposure and acoustic/decoder capacity. Four inputs approach five-input accuracy; the extra embedding coordinate gives modest, split-dependent refinements rather than a uniform improvement (Appendix C.1). Domain sensitivities use explicitly matched test sets (Appendix A).

5.3 From predictions to choices

For a specified compute budget, the model can rank supported configurations by their two predicted responses:

$$\mathcal{P}(C) = \mathrm{Pareto}\{(\widehat{L_{\mathrm{text}}}(x), \widehat{L_{\mathrm{asr}}}(x)) : \widehat{C}_{\mathrm{op}}(x) \leq C,\ x \in \mathcal{X}_{\mathrm{supported}}\}. \tag{8}$$

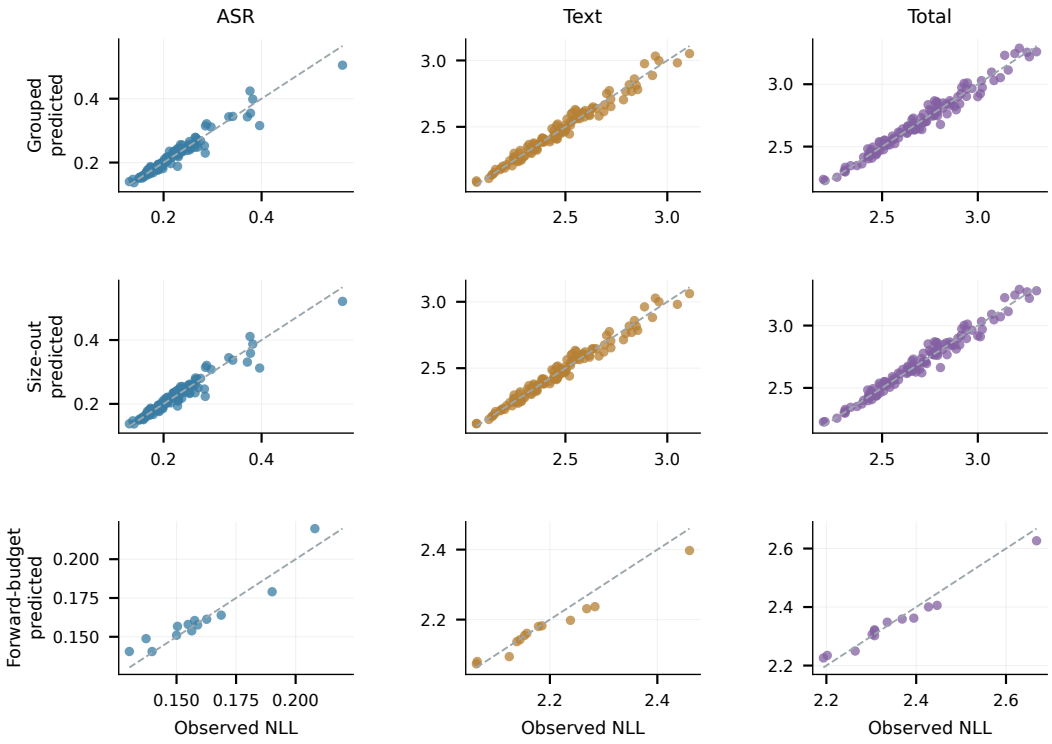


Figure 4: **Held-out predictions of the five-factor law.** Columns show ASR, text and total validation NLL. Rows test grouped configurations, unseen size strata and higher budgets. Dashed lines indicate equality. The first two protocols each score 116 endpoints; the third scores thirteen. These are retrospective tests on the declared domain, not forecasts issued before the experiments.

Before training, architecture fixes the capacity inputs; packing calibration estimates per-update compute and supervision, from which a budget determines the horizon and expected T, Y (Appendix C.3). Held-out scores use realized exposures, separating loss prediction from input-estimation error. Accurate NLL prediction does not guarantee CER/WER ranking: size-out selection performs well in the M grid and higher-budget L slices, but low-budget L rankings can reverse (Appendix C.2). The law supplies loss-based shortlists, not guaranteed recognition optima; downstream verification remains necessary.

6 Discussion and Conclusion

Continuous speech and text provide a concrete setting for investigating native multimodal pretraining. Our systematic experiments establish joint learning from random initialization and reveal how model scale, capacity placement, and modality supervision shape the resulting capabilities. The observations extend beyond budget dependence: neither more acoustic capacity nor more speech supervision is uniformly better, text supervision can support recognition, and acoustic context need not reside in a separate encoder.

Scaling laws connect these empirical relationships to loss prediction and configuration selection. Targeted seed replications quantify configuration-dependent variability, with detailed checks at five specified configurations (Appendix M). Initialization controls and a small total-budget-matched pilot examine separately learned priors (Appendix H). Together, these results provide a basis for studying native multimodal learning and choosing its architecture and training allocation.

AI USE STATEMENT

Generative AI tools assisted hypothesis refinement, methodology discussion, implementation and testing of training-management and analysis code, interpretation, literature search, manuscript drafting, translation, figures, tables, and nonofficial reviews. Numerical outcomes come from executed training and evaluation, not synthetic model-generated results. Suggestions were checked against recorded artifacts; no theoretical proof or official peer-review decision is claimed. Human authors retain responsibility for scientific claims, attribution, data governance, and final verification.

ETHICS STATEMENT

Training uses internal Internet speech–transcript data and public text resources. Source availability alone does not establish consent, licensing suitability, or absence of personal information. We do not release raw speech or claim a complete waveform-level contamination audit. Results concern aggregate offline modeling and recognition, not deployment readiness or equitable performance across populations. Any release requires appropriate provenance, privacy, and licensing review.

REPRODUCIBILITY STATEMENT

Appendices specify data/model conventions, arithmetic accounting, evaluation, and the completed configuration catalog. The anonymous numerical supplement and code reproduce figures, tables, and prediction analyses without training access. The five-factor analysis uses the numerical catalog, fitting domains, and validation protocols specified in Appendix A. The internal speech corpus is not released; numerical-analysis reproducibility does not require access to it.

REFERENCES

Junyi Ao, Rui Wang, Long Zhou, Chengyi Wang, Shuo Ren, Yu Wu, Shujie Liu, Tom Ko, Qing Li, Yu Zhang, Zhihua Wei, Yao Qian, Jinyu Li, and Furu Wei. SpeechT5: Unified-modal encoder-decoder pre-training for spoken language processing. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5723–5738, 2022. doi: 10.18653/v1/2022.acl-long.393. URL https://aclanthology.org/2022.acl-long.393/.

He Bai, Tatiana Likhomanenko, Ruixiang Zhang, Zijin Gu, Zakaria Aldeneh, and Navdeep Jaitly. dMel: Speech tokenization made simple. *arXiv preprint arXiv:2407.15835*, 2024. URL https://arxiv.org/abs/2407.15835.

Yamini Bansal, Behrooz Ghorbani, Ankush Garg, Biao Zhang, Colin Cherry, Behnam Neyshabur, and Orhan Firat. Data scaling laws in NMT: The effect of noise and architecture. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 1466–1482, 2022. URL https://proceedings.mlr.press/v162/bansal22b.html.

Ankur Bapna, Colin Cherry, Yu Zhang, Ye Jia, Melvin Johnson, Yong Cheng, Simran Khanuja, Jason Riesa, and Alexis Conneau. mSLAM: Massively multilingual joint pre-training for speech and text. *arXiv preprint arXiv:2202.01374*, 2022. URL https://arxiv.org/abs/2202.01374.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. URL https://arxiv.org/abs/1911.11641.

Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. URL https://arxiv.org/abs/2304.08818.

Hui Bu, Jiayu Du, Xingyu Na, Bengu Wu, and Hao Zheng. AISHELL-1: An open-source mandarin speech corpus and a speech recognition baseline. *arXiv preprint arXiv:1709.05522*, 2017. URL https://arxiv.org/abs/1709.05522.

Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024. URL https://arxiv.org/abs/2405.09818.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, et al. Qwen2-Audio technical report. *arXiv preprint arXiv:2407.10759*, 2024. URL https://arxiv.org/abs/2407.10759.

Aidan Clark, Diego De Las Casas, Aurelia Guy, Arthur Mensch, Michela Paganini, Jordan Hoffmann, Bogdan Damoc, Blake Hechtman, Trevor Cai, Sebastian Borgeaud, George Bm Van Den Driessche, Eliza Rutherford, Tom Hennigan, Matthew J Johnson, Albin Cassirer, Chris Jones, Elena Buchatskaya, David Budden, Laurent Sifre, Simon Osindero, Oriol Vinyals, Marc'Aurelio Ranzato, Jack Rae, Erich Elsen, Koray Kavukcuoglu, and Karen Simonyan. Unified scaling laws for routed language models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 4057–4086, 2022. URL https://proceedings.mlr.press/v162/clark22a.html.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of NAACL-HLT*, pp. 2924–2936, 2019. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300/.

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.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Defossez. Simple and controllable music generation. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2306.05284.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=mZn2Xyh9Ec.

Ruchao Fan, Yiming Wang, Yuxuan Hu, Bo Ren, Yufei Xia, Xiaofei Wang, Yao Qian, Shujie Liu, and Jinyu Li. LLM can read spectrogram: Encoder-free speech-language modeling. *arXiv preprint arXiv:2606.10231*, 2026. URL https://arxiv.org/abs/2606.10231.

Behrooz Ghorbani, Orhan Firat, Markus Freitag, Ankur Bapna, Maxim Krikun, Xavier Garcia, Ciprian Chelba, and Colin Cherry. Scaling laws for neural machine translation. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2109.07740.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/c1e2faff6f588870935f114ebe04a3e5-Abstract-Conference.html.

Maor Ivgi, Yair Carmon, and Jonathan Berant. Scaling laws under the microscope: Predicting transformer performance from small scale experiments. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 7354–7371, 2022. doi: 10.18653/v1/2022.findings-emnlp.544. URL https://aclanthology.org/2022.findings-emnlp.544/.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://arxiv.org/abs/1806.07572.

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. URL https://arxiv.org/abs/2001.08361.

Rabeeh Karimi Mahabadi, Sanjeev Satheesh, Shrimai Prabhumoye, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. Nemotron-CC-Math: A 133 billion-token-scale high quality math pretraining dataset. *arXiv preprint arXiv:2508.15096*, 2025. URL https://arxiv.org/abs/2508.15096.

Kimi Team. Kimi K3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026. URL https://arxiv.org/abs/2607.24653.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Gadre, Hritik Bansal, Etash Guha, et al. DataComp-LM: In search of the next generation of training sets for language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/19e4ea30dded58259665db375885e412-Abstract-Datasets_and_Benchmarks_Track.html.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the International Conference on Machine Learning*, pp. 19730–19742, 2023. URL https://proceedings.mlr.press/v202/li23q.html.

Weixin Liang, Lili Yu, Liang Luo, Srinivasan Iyer, Ning Dong, Chunting Zhou, Gargi Ghosh, Mike Lewis, Wen-tau Yih, Luke Zettlemoyer, and Xi Victoria Lin. Mixture-of-Transformers: A sparse and scalable architecture for multi-modal foundation models. *Transactions on Machine Learning Research*, 2025. URL https://arxiv.org/abs/2411.04996.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Potsawee Manakul, Woody Haosheng Gan, Martijn Bartelds, Guangzhi Sun, William Held, and Diyi Yang. Scaling open discrete audio foundation models with interleaved semantic, acoustic, and text tokens. *arXiv preprint arXiv:2602.16687*, 2026. URL https://arxiv.org/abs/2602.16687.

MiniMax. MiniMax M3: Frontier coding, 1m context, native multimodality—all in one model. Technical blog, 2026. URL https://www.minimax.io/blog/minimax-m3.

Niklas Muennighoff, Alexander M. Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. Scaling data-constrained language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/9d89448b63ce1e2e8dc7af72c984c196-Abstract-Conference.html.

NVIDIA et al. NVIDIA Nemotron Nano 2: An accurate and efficient hybrid Mamba-Transformer reasoning model. arXiv preprint arXiv:2508.14444, 2025. URL https://arxiv.org/abs/2508.14444.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An ASR corpus based on public domain audio books. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5206–5210, 2015. doi: 10.1109/ICASSP.2015.7178964.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1525–1534, 2016. doi: 10.18653/v1/P16-1144. URL https://aclanthology.org/P16-1144/.

Tomer Porian, Mitchell Wortsman, Jenia Jitsev, Ludwig Schmidt, and Yair Carmon. Resolving discrepancies in compute-optimal scaling of language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://arxiv.org/abs/2406.19146.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 28492–28518, 2023. URL https://proceedings.mlr.press/v202/radford23a.html.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, 2016. doi: 10.18653/v1/D16-1264. URL https://aclanthology.org/D16-1264/.

Siva Reddy, Danqi Chen, and Christopher D. Manning. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, 2019. doi: 10.1162/tacl_a_00266. URL https://aclanthology.org/Q19-1016/.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bjorn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022. URL https://arxiv.org/abs/2112.10752.

Xian Shi, Xiong Wang, Zhifang Guo, Yongqi Wang, Pei Zhang, Xinyu Zhang, Zishan Guo, Hongkun Hao, et al. Qwen3-ASR technical report. *arXiv preprint arXiv:2601.21337*, 2026. URL https://arxiv.org/abs/2601.21337.

Mustafa Shukor, Louis Bethune, Dan Busbridge, David Grangier, Enrico Fini, Alaaeldin El-Nouby, and Pierre Ablin. Scaling laws for optimal data mixtures. In *Advances in Neural Information Processing Systems*, volume 38, 2025a. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/bc1d640f841f752c689aae20b31198c1-Abstract-Conference.html.

Mustafa Shukor, Enrico Fini, Victor Guilherme Turrisi da Costa, Matthieu Cord, Joshua Susskind, and Alaaeldin El-Nouby. Scaling laws for native multimodal models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025b. URL https://openaccess.thecvf.com/content/ICCV2025/html/Shukor_Scaling_Laws_for_Native_Multimodal_Models_ICCV_2025_paper.html.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, et al. Dolma: An open corpus of three trillion tokens for language model pretraining research. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15725–15788, 2024. doi: 10.18653/v1/2024.acl-long.840. URL https://aclanthology.org/2024.acl-long.840/.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of NAACL-HLT*, pp. 4149–4158, 2019. doi: 10.18653/v1/N19-1421. URL https://aclanthology.org/N19-1421/.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. SALMONN: Towards generic hearing abilities for large language models. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=14rn7HpKVk.

Team Olmo et al. Olmo 3. arXiv preprint arXiv:2512.13961, 2025. URL https://arxiv.org/abs/2512.13961.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Haoyuan Wu, Aoqi Wu, Hai Wang, Jiajia Wu, Jinxiang Ou, and Bei Yu. Scaling native multimodal pre-training from scratch. *arXiv preprint arXiv:2607.22043*, 2026. URL https://arxiv.org/abs/2607.22043.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/dcba6be91359358c2355cd920da3fcbd-Abstract-Conference.html.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Jiasheng Ye, Peiju Liu, Tianxiang Sun, Jun Zhan, Yunhua Zhou, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/cc84bfabe6389d8883fc2071c848f62a-Abstract-Conference.html.

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/.

A CURRENT ANALYSIS POPULATION AND FIVE-FACTOR VALIDATION

This section specifies the fitting domain, model estimation, validation splits, and sensitivity analyses for the 116-observation five-factor law.

Analysis domain. The primary analysis contains 116 observations from the cold joint-training family. Encoder-free/text-only and dedicated width–depth controls are outside this model's domain; they remain separate architectural comparisons. Each included observation is identified with its input coordinates and response values in `data/scaling_inputs.json`.

For the budget-domain sensitivity, the 126-row comparison catalog includes ten additional nominal $0.25C_0$ endpoints: R013/R018/R023/R028/R043/R107 and X002/X006/X007/X009. Nominal membership uses audited cost within 1% of $0.25C_0$. Other batches with distinct references below $0.5C_0$ remain; the rule is not a blanket $C \geq 0.5C_0$ filter. The primary 116 observations have unit weight. Both catalogs are fixed for this retrospective analysis.

Fitting and validation. The five power-law inputs are normalized as $x = (P_a/10^8, P_d/10^8, P_e/10^8, T/10^9, Y/10^9)$. Nonnegative floors and amplitudes and exponents in $[0.001, 3]$ are fitted by least squares in log response, with three fixed exponent initializations 0.15, 0.5 and 1.25. The two-factor baseline uses $(P_a + P_d, T + Y)$ and the same training rows. Kernels use $\log(1 + P_a/10^8)$, $\log(1 + P_d/10^8)$, $\log(P_e/(151936 \cdot 896))$, $\log(1 + T/10^9)$ and $\log(1 + Y/10^9)$, standardized using training rows only. Ridge candidates are $\{0.001, 0.01, 0.1, 1, 10\}$, Matérn lengths $\{0.75, 1.5, 3\}$ and NTK depths $\{1, 2, 3\}$. Three inner grouped folds select kernel hyperparameters by log-RMSE, choosing the lowest effective degrees of freedom within one standard error of the best score. The hybrid is a two-factor power trend multiplied by an NTK log-residual correction. Full parameters, per-fold IDs and predictions are supplied in `data/scaling_inputs.json`.

Groups collect all included observations sharing width, encoder depth, decoder depth and audited budget bucket. Five-fold assignments are deterministic, balanced using reference group sizes, and preserved after exclusions. Thus another budget for a held-out geometry can remain in grouped training. The size-out test removes an entire embedding-size stratum, including every allocation and mixture at that width; there are six strata. The forward-budget test trains at $C/C_0 \leq 2.02$ and evaluates thirteen endpoints at $C/C_0 \geq 3.98$. Grouped and size-out scores pool one out-of-fold prediction per retained observation. All scalers and hyperparameter choices exclude outer test responses. Full-data fits shown as curves are distinct from these held-out predictions.

The thirteen forward-budget endpoints are R017, R022, R027, R032, R044, R105, X018, X023, X025, X026, X028, X029, and Z001.

Comparison A: two domains, common 116-point test population. Table 2 compares Broad (126 configurations before holdout) with Budget (116). Grouped and size-out scores are restricted to the same 116 configurations retained by Budget; forward-budget scores use the same thirteen high-budget endpoints. The ten extra quarter-budget observations enter a Broad training fold only when assigned to its training side; they never expose a held-out response. Predictions are then scored on the common IDs. Counts describe the complete eligible catalogs, not each fold's training size.

Table 2: Comparison A: Broad versus Budget on common 116-point grouped/size-out tests and common 13-point forward-budget tests. G/S/F denote those three protocols. Values are RMSE.

Domain	Catalog	ASR G	ASR S	ASR F	Text G	Text S	Text F
Broad	126	0.02891	0.02579	0.01482	0.03584	0.03531	0.02926
Budget	116	0.01600	0.01613	0.00675	0.03614	0.03523	0.02832

Comparison B: four domains, common 112-point test population. We additionally apply an exposure criterion of at least 300M ASR targets, either alone (Exposure) or together with the budget criterion (Both). The four eligible catalogs contain 126, 116, 115, and 112 configurations. Their intersection supplies the 112 scored grouped/size-out points in Figure 5 and Table 3; forward-budget tests still score the same thirteen endpoints. This is a different test population from Comparison A: its RMSE values must not be directly substituted for the 116-point values. Within each comparison,

common test IDs isolate fitting-support changes from test-set difficulty. The criteria define analysis domains, not independently measured alignment thresholds.

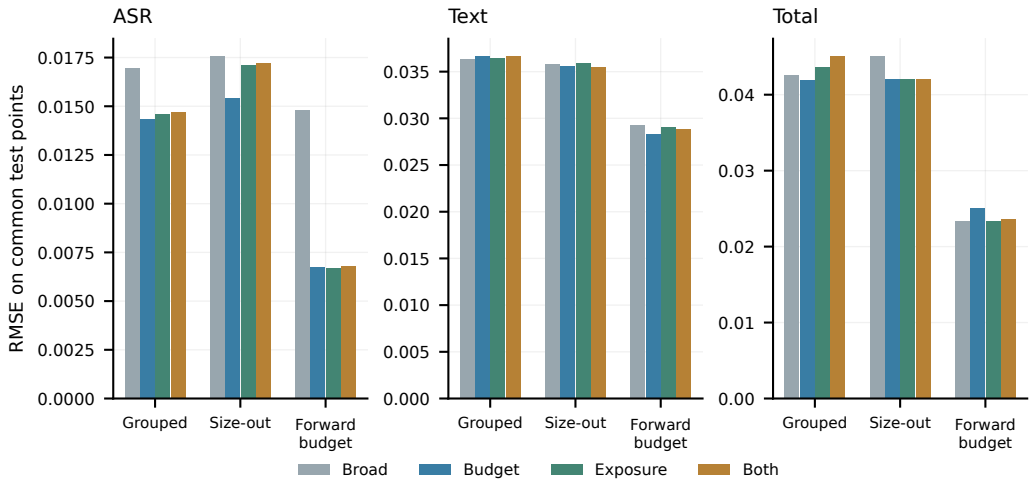


Figure 5: Comparison B: fitting-domain sensitivity on common 112-point grouped/size-out tests and common 13-point forward-budget tests. Broad includes the quarter-budget slice; Budget is primary; Exposure requires at least 300M ASR targets; Both applies both criteria.

Table 3: Comparison B: four-domain RMSE on common 112-point G/S tests and 13-point F tests. Catalog counts are before outer holdout; per-fold training counts vary. Comparison A in Table 2 instead uses 116 G/S points.

Domain	Catalog	ASR G	ASR S	ASR F	Text G	Text S	Text F
Broad	126	0.01695	0.01759	0.01482	0.03631	0.03579	0.02926
Budget	116	0.01433	0.01542	0.00675	0.03664	0.03565	0.02832
Exposure	115	0.01458	0.01710	0.00671	0.03641	0.03593	0.02903
Both	112	0.01472	0.01723	0.00679	0.03669	0.03551	0.02890

Trajectory visualization. Figure 1 uses 27 balanced-size trajectories after removing the six quarter-budget scale runs. Intermediate cost is approximated by terminal audited cost times step divided by horizon, not an exact prefix ledger. The envelope is the best recorded validation checkpoint within that estimated budget, not a model-switching training algorithm or an independent short-horizon experiment. For readability, the ASR overview masks ten interior values exceeding 1.8 times their local eleven-evaluation median and three adjoining values from the same short-lived XL excursion. The first and last five evaluations remain unfiltered. Raw points and the explicit display mask are supplied in the numerical supplement and `data/trajectory_display_mask.json`; text values, terminal observations, and regression responses are unchanged. The text panel displays NLL 2–5.7 without an inset. Hollow markers query the observed envelope at fixed budgets. Smooth IsoFLOP law curves interpolate the five input coordinates in log space between measured sizes before evaluating the model; they do not establish a continuous family of executable architectures or a verified continuous optimum.

B Interactions between capacity and exposure

B.1 From separable terms to structured interactions

The additive law in Equation 5 provides a compact account of capacity and exposure, but does not explicitly let the return to one input depend on another. Such coupling is useful in related scaling studies of routed models and data mixtures (Clark et al., 2022; Shukor et al., 2025a; Ye et al.,

2025). We therefore examine whether structured interactions improve prediction on the same 116 configurations. This is a retrospective functional-form sensitivity analysis; the primary law, fitting population and predictions in the main text remain unchanged.

Let $L_0(x)$ denote the five-factor law and let z_j be the standardized logarithm of its normalized input x_j. Standardization and all fitted coefficients use only the training side of a split. We consider residual models

$$\log \widehat{L}(x) = \log L_0(x) + b + g(z). \tag{9}$$

Two compact choices are a bounded acoustic-capacity/ASR-exposure correction and a cross-block bilinear correction:

$$g_{\mathrm{sat}}(z) = c \tanh(z_a/k) \tanh(z_Y/k), \tag{10}$$

$$g_{\mathrm{cross}}(z) = \mathbf{z}_P^\top B \mathbf{z}_D, \quad \mathbf{z}_P = (z_a, z_d, z_e)^\top, \quad \mathbf{z}_D = (z_T, z_Y)^\top. \tag{11}$$

We fit both an unrestricted 3×2 matrix and a rank-one matrix between the capacity and data blocks. The latter has four effective interaction degrees of freedom rather than six, excluding the intercept. The bounded form limits the correction when either standardized coordinate is far from its training range. We also test a linear acoustic–ASR interaction with all five log main effects retained.

Alternative parameterizations include a multiplicative law with a floor; amplitudes modulated by allocation $q = P_a/(P_a+P_d)$ or mixture $h = Y/(T+Y)$; and effective-capacity/exposure aggregation. The amplitude-modulated family is

$$\widehat{L} = E + \sum_{j=1}^{5} A_j x_j^{-\alpha_j} \exp\{\beta_j (r - 1/2)\}, \qquad r \in \{q, h\}. \tag{12}$$

These are adaptations motivated by joint scaling laws, not replications of the cited models. Our CES variant uses separately weighted generalized means of (P_a, P_d, P_e) and (T, Y), each with its own shape parameter, followed by two additive power terms. A further variant allows only the acoustic-capacity and ASR-exposure exponents to depend smoothly on h, constrained between zero and three. Exposure alternatives use either a seven-direction, nonredundant log-mixture dictionary or a shared transfer weight in $\log(T + wY)$ and $\log(Y + wT)$. Neither these effective coordinates nor the interaction coefficients are interpreted as causal transfer efficiencies.

B.2 Validation and predictive results

We preserve the grouped, size-out and forward-budget outer splits of Appendix A. For regularized structured corrections, inner validation matches the intended task: grouped folds for grouped prediction, withheld widths for size-out prediction, and increasing-budget splits for forward prediction. Ridge penalties are chosen from $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 100\}$ using a one-standard-error rule favoring stronger shrinkage. The saturation scale $k \in \{0.5, 1, 2\}$ and transfer-weight candidates are selected within the same inner loop. Nonlinear families use fixed parameter bounds and multiple training-loss initialization points. No outer response selects a penalty, bound or initialization.

The saturated acoustic–ASR correction reduces ASR MAPE from 4.17/4.37/3.32% to 3.93/4.04/2.82% across G/S/F. Cross-block interactions also improve grouped and size-out ASR prediction, but offer little forward-budget gain. More flexibility is not uniformly beneficial: allocation-dependent amplitudes reduce grouped ASR error to 3.52% while increasing size-out error to 6.15%. Mixture-dependent amplitudes improve grouped and size-out text prediction, but not forward-budget text prediction. The CES and mixture-dependent-exponent variants likewise do not consistently improve over the additive reference.

We additionally withhold each of the fifteen M allocation–mixture cells across both compute references, scoring 30 endpoints. Training excludes any further point with the same width and encoder/decoder depths and ASR target share within 0.01 of the withheld cell. This is a test of an unseen combination within the M grid, not a new-scale test. ASR MAPE is 3.36% for the additive law, 2.91% for the saturated correction and 3.01% for rank-one interactions. Paired group-resampled error intervals remain compatible with no gain for saturation on this test and on the 13-point forward-budget test. These intervals condition on the fitted predictions; they do not include the uncertainty of searching across functional forms.

Table 4: Interaction sensitivity on shared outer splits. Entries are MAPE (%); G/S/F denote grouped, size-out and forward-budget prediction. Each response is fitted separately. The primary additive law is the reference, not a model selected from this table. The numerical supplement contains the wider fixed-family comparison and all fold predictions.

	ASR			Text			Total		
Form	G	S	F	G	S	F	G	S	F
Additive five-factor	4.174	4.368	3.318	1.113	1.094	0.872	1.190	1.183	0.911
Multiplicative + floor	8.100	9.466	18.097	1.810	2.234	2.378	2.479	3.226	3.021
Mixture-dependent amplitudes	3.910	3.987	3.237	0.869	0.917	0.988	1.012	1.103	0.917
Allocation-dependent amplitudes	3.523	6.153	3.947	1.171	1.367	0.972	1.218	1.524	1.020
Effective-exposure dictionary	4.171	4.341	3.351	1.089	1.089	0.774	1.170	1.178	0.866
Saturated acoustic–ASR	3.933	4.041	2.823	1.091	1.071	0.842	1.162	1.165	0.789
Main effects + acoustic–ASR	4.075	4.168	3.317	1.058	1.056	0.867	1.157	1.190	0.831
Capacity–exposure matrix	3.649	3.992	3.311	1.075	1.055	0.862	1.162	1.203	0.855
Rank-one capacity–exposure	3.752	4.053	3.319	1.073	1.052	0.861	1.159	1.204	0.800
Shared transfer weight	4.172	4.353	3.345	1.109	1.093	0.799	1.187	1.181	0.890
CES capacity and exposure	4.236	6.729	3.450	1.137	1.135	1.080	1.373	1.503	1.149
Mixture-dependent exponents	4.497	4.750	3.552	1.114	1.128	0.886	1.155	1.146	0.850

B.3 Regularization, identifiability and low-budget behavior

To distinguish coefficient stability from a fixed-penalty diagnostic, we repeat 500 group bootstrap fits per response for the simple acoustic–ASR residual term, refitting the additive law and reselecting the ridge penalty inside each resample. Its ASR coefficient remains negative in all resamples, whereas the text coefficient interval includes zero. This supports a response-specific predictive correction, not a general cross-modal mechanism. Correlated capacity coordinates and the considerably higher condition number of quadratic features further limit individual-coefficient interpretation.

Changing amplitude-modulation bounds also affects generalization. With six initialization points per fit, widening the allocation-modulation bound from $|\beta_j| \leq 1$ to $|\beta_j| \leq 6$ increases size-out ASR MAPE from 5.35% to 8.11%; the intermediate bound gives 6.23%. The intermediate value differs slightly from Table 4 because this diagnostic uses more initialization points and tighter optimization tolerances. Boundary solutions therefore warrant sensitivity analysis rather than interpretation as independently measured elasticities.

Finally, the ten quarter-budget configurations remain outside fitting and hyperparameter selection. On these points, the additive ASR MAPE is 24.91%. The acoustic–ASR residual correction gives 24.35% without ridge shrinkage and 24.61% at penalty 100; the eight structured alternatives range from 24.60% to 25.16%. Thus the limited improvement is not explained solely by strong regularization. These tested forms do not remove the low-budget discrepancy, but this result neither rules out all interaction models nor identifies its dynamical cause.

Overall, bounded and low-rank interactions are useful compact alternatives within the observed design. Their gains do not justify a uniform replacement of the separable law, especially when the intended prediction direction changes. Complete model predictions, numerical settings, bootstrap records and regularization paths are provided in `data/interaction_scaling.json`.

C Feature ablations and recognition-based configuration selection

C.1 Which input distinctions improve prediction?

Embedding capacity is explicitly represented because the vocabulary matrix contributes approximately 11–52% of total parameters across the balanced model family. This is a capacity-accounting motivation, distinct from whether the extra coordinate improves prediction on every split. We compare progressively separated input sets using the original 116 configurations and exactly the same outer folds as Table 1. With $N = P_a + P_d$, the two-, three-, four-, and five-input models use $(N, T + Y)$, (N, T, Y), (P_a, P_d, T, Y), and (P_a, P_d, P_e, T, Y), respectively. Each is an additive

power law with one floor and one amplitude/exponent pair per input. They therefore have 5, 7, 9, and 11 nominal parameters per response; the comparison does not match model complexity. The fitting loss, bounds, three initialization points, and optimizer tolerances are unchanged. No held-out response selects the input set or fitting initialization. The original two-/five-input predictions are retained; the new fits use only their original training IDs.

Table 5: Feature ablation on shared outer splits. G/S/F denote grouped, size-out, and forward-budget RMSE. Parameter counts include floors, amplitudes, and exponents; they are not effective degrees of freedom.

Response	Inputs	Parameters	G	S	F
ASR	2	5	0.04438	0.16644	0.03347
ASR	3	7	0.01704	0.03300	0.00740
ASR	4	9	0.01577	0.01642	0.00711
ASR	5	11	0.01600	0.01613	0.00675
Text	2	5	0.12783	0.12753	0.07364
Text	3	7	0.09900	0.09831	0.04939
Text	4	9	0.03643	0.03520	0.02840
Text	5	11	0.03614	0.03523	0.02832
Total	2	5	0.12278	0.12007	0.05184
Total	3	7	0.10603	0.10522	0.05748
Total	4	9	0.04184	0.04359	0.02508
Total	5	11	0.04184	0.04195	0.02508

Separating text and ASR exposure reduces grouped ASR RMSE from 0.04438 to 0.01704; separating acoustic and decoder capacity further reduces it to 0.01577. The five-input value is 0.01600. For text, capacity separation produces the larger improvement, from 0.09900 to 0.03643. Adding embeddings yields modest, split-dependent refinements: the five-input model improves size-out and forward-budget ASR relative to four inputs, but not grouped ASR; text results are nearly unchanged. Thus the strongest predictive gains come from distinguishing modalities and capacity locations, not from an assertion that every added coordinate is indispensable.

Table 6: Full-population five-input coefficients in Equation 5, using parameter inputs divided by 10^8 and exposure inputs by 10^9. Amplitudes below 10^{-9} are numerically negligible; their exponents are omitted because they do not meaningfully characterize the fitted response. Exact coefficients remain in the numerical release.

	ASR		Text		Total	
Input	A	α	A	α	A	α
P_a	0.008106	1.3374	$< 10^{-9}$	–	0.004257	2.4756
P_d	0.1298	0.0722	0.602	0.2821	0.7335	0.2467
P_e	0.01819	2.9589	$< 10^{-9}$	–	$< 10^{-9}$	–
T	0.02702	0.9681	0.815	0.3854	0.8572	0.3890
Y	0.07702	0.8729	$< 10^{-9}$	–	0.05246	1.1204

The fitted floors are approximately zero, 1.4758, and 1.4911 for ASR, text, and total. In the full text fit, the acoustic, embedding, and ASR-exposure amplitudes are near zero, leaving decoder capacity and text exposure as the active terms. This is a property of this fit, not proof of causal irrelevance. The transformed five-input design has rank five and condition number 11.44; embedding capacity has variance-inflation factor 26.79. Predictive utility and separately identifiable exponents are therefore different questions. `data/feature_ablation.json` supplies all predictions, training/test IDs, RMSE/MAE/MAPE, convergence records, and coefficients.

C.2 Does NLL-based selection recover good recognition configurations?

We evaluate six fixed candidate sets: the five L allocations at each of $0.5, 1, 2, 4C_0$, and all fifteen M allocation–mixture cells at each of the two matched compute references. The sets contain 50 distinct

configurations. Within each set, every candidate is predicted by the *same* size-out model, trained with that entire width stratum removed. We choose the candidate minimizing predicted ASR NLL, then score its already completed AISHELL/LibriSpeech results. This is retrospective selection using realized exposures, not a claim that training budgets were prospectively selected by this procedure. We do not combine predictions from differently fitted folds within a candidate set.

Table 7: Within-set Spearman correlations with recognition error. Predicted uses size-out ASR-NLL predictions; observed uses measured ASR NLL. Positive values indicate agreement of the lower-is-better rankings. C/Wc/Wo denote CER and clean/other WER. Each L set has five candidates; each M set has fifteen.

		Predicted NLL			Observed NLL		
Candidate set	n	C	Wc	Wo	C	Wc	Wo
L allocation $0.5C_0$	5	-0.70	-0.90	-0.70	0.30	0.50	0.30
L allocation $1C_0$	5	-0.10	-0.30	-0.10	0.80	0.90	0.80
L allocation $2C_0$	5	0.60	0.90	1.00	0.90	0.90	0.70
L allocation $4C_0$	5	1.00	0.70	0.90	0.40	0.50	0.70
M lower	15	0.81	0.90	0.84	0.85	0.95	0.90
M upper	15	0.82	0.80	0.75	0.92	0.95	0.94

Table 8: Recognition performance of the candidate selected by size-out predicted ASR NLL. Scores are percentages; regret is the selected error minus the best measured error *for that metric within the same set*, in percentage points. Zero does not imply statistical significance.

		Selected scores			Regret (pp)		
Candidate set	ID	CER	WERc	WERo	CER	WERc	WERo
L allocation $0.5C_0$	X005	13.269	8.699	21.767	0.760	0.521	0.425
L allocation $1C_0$	X016	8.937	6.142	15.600	1.328	1.045	2.171
L allocation $2C_0$	X021	5.605	3.648	8.915	0.246	0.042	0.000
L allocation $4C_0$	X028	4.219	2.963	7.331	0.000	0.019	0.084
M lower	R078	5.838	4.170	10.365	0.000	0.000	0.000
M upper	R102	4.786	3.455	8.583	0.407	0.156	0.498

The lower-reference M selection attains the best measured value on all three recognition metrics. For the upper-reference M set, regret is 0.407/0.156/0.498 points for CER/clean/other WER. The L $4C_0$ selection is within 0.084 points on all three metrics. However, the low-budget L sets reveal ranking errors: at $1C_0$, size-out selection incurs 2.171 points of other-WER regret. Even selecting by measured NLL does not always minimize recognition error, as the observed-NLL correlations show. Both predictive error and the distinction between token likelihood and sequence error matter.

An additional forward-budget check predicts the five L $4C_0$ allocations with a model trained only through approximately $2C_0$. It selects X029, with CER/clean/other regrets of 0.015/0.031/0.000 points. This reuses the L $4C_0$ candidate set and is not a seventh independent set. These results support NLL-based shortlisting in supported regimes, followed by downstream verification; they do not establish a universal CER/WER-optimal allocation. Full candidate predictions, observed responses, selected IDs, and fold identities are in `data/recognition_selection.json`; all 140 completed recognition summaries are in `data/recognition_results.json`.

C.3 Interpreting and using the law before training

A quantitative account of resource trade-offs. The response model summarizes how capacity and supervision jointly organize the observed losses. It is not a model of every optimization transient. Although each positive power-law term decreases with its own input, an allocation change under a fixed budget moves several inputs together: increasing acoustic capacity can reduce decoder capacity or the affordable number of updates, while changing the packing mixture changes both target exposures and per-update cost. Interior optima can therefore arise along a constrained configuration path without requiring a non-monotone coordinate term. This explains how the model can

represent a resource trade-off; whether it captures a particular empirical profile must still be checked against that profile.

The full-data text fit places negligible amplitude on acoustic capacity, embedding capacity and ASR exposure (Table 6). Within this model and domain, most of its text-response variation is represented by decoder capacity and text exposure. This is consistent with the complementary same-decoder/text-exposure control in Appendix I; it does not establish that the other inputs have zero causal effect. The separate ASR response and the boundary comparisons motivate an asymmetric dependence of the two capabilities on shared resources. Input ablations and held-out predictions test the usefulness of these distinctions. Low-exposure plateaus and rapid ASR-loss decreases are additional dynamical observations, not mechanisms derived from the static endpoint law.

From a proposed configuration to predicted losses. For a candidate architecture and sampling/packing policy z, parameter counting gives $P_a(z), P_d(z), P_e(z)$ exactly. Data metadata and representative packing calibration can estimate the mean supervised text and ASR targets per update, $\bar{t}(z), \bar{y}(z)$, and the mean operator cost $\bar{f}(z)$ using the same FLOP convention as the experiments. No endpoint loss is required for this calibration. With an approximately stationary policy and budget C, the prospective inputs are

$$\widehat{U}(z, C) = \left\lfloor \frac{C}{\bar{f}(z)} \right\rfloor, \qquad \widehat{T} = \widehat{U}\bar{t}(z), \qquad \widehat{Y} = \widehat{U}\bar{y}(z). \tag{13}$$

Schedules with changing mixtures instead sum their planned per-update costs and exposures. Evaluating the two fitted responses at these capacity and exposure coordinates gives a candidate pair $(\widehat{L}_{\mathrm{asr}}, \widehat{L}_{\mathrm{text}})$. Filtering dominated candidates implements the shortlist in Eq. 8; downstream objectives determine the final choice. This procedure applies within the supported architecture and training-policy domain, not to arbitrary new modality interfaces.

Input estimation and response prediction are separate checks. The reported held-out tests use realized exposure ledgers and assess the response model conditional on those coordinates. Prospective use also incurs error in $\bar{f}, \bar{t}, \bar{y}$; its end-to-end accuracy is not measured by the conditional scores alone. An independent planning evaluation would freeze the calibration and candidate schedule before training, then compare estimated exposures and predicted losses with their realized values. The component predictions define the two-objective shortlist. The separately fitted total response is an additional analysis of the fixed-weight training objective, not an algebraic replacement for their sum or a task-independent utility function.

D Training data, architecture, and optimization

Speech and text data. All speech pretraining in this study uses only an internal Internet ASR pool of speech–transcript segments of at most 30 seconds. Internet-derived supervision provides a scalable setting for studying pretraining beyond fixed benchmark training sets. No AISHELL, LibriSpeech, or other public ASR training split is added to that pool. The main recognition results test pretrained checkpoints directly, with no intermediate benchmark-specific adaptation. Appendix N separately reports benchmark-specific fine-tuning from these checkpoints. The corpus is proprietary and is not released. Reported audio hours count cumulative processed exposure, including repeated sampling, rather than unique corpus hours; the scaling analysis uses realized supervised-target exposures.

Public text sources include DCLM Baseline (Li et al., 2024); Nemotron-CC v2 and Nemotron pretraining code/SFT releases (NVIDIA et al., 2025); Nemotron-CC-Math v1 (Karimi Mahabadi et al., 2025); and Dolma 3 Mix 6T components (Team Olmo et al., 2025), including Common Crawl, OLMoCR science PDFs, Stack-Edu, FineMath, RedPajama/ProofPile arXiv, and Dolma 1.7 English Wikipedia (Soldaini et al., 2024). Version-specific dataset cards are available for Nemotron-CC v2, Nemotron-CC-Math v1, and Dolma 3 Mix 6T. These references identify the source families and releases; our mixture need not reproduce their default weights. FineWeb2 and SkyPile are not used in the principal training results.

Data choice and reproducibility. Using an internal corpus makes end-to-end training replication more difficult; the released numerical materials reproduce the analyses, not the original corpus. Our

data choice serves the study of capacity and supervision across a broad exposure range. The largest single run processes approximately 1.507 million audio hours and 92.51 billion text targets. Restricting such a range to a much smaller fixed corpus would introduce substantial data reuse as an additional experimental axis, whose effective value requires separate treatment (Muennighoff et al., 2023). We therefore use a consistently processed Internet speech pool rather than make repeated training on a small benchmark corpus the object of study. Cumulative exposure alone does not establish unique corpus size or an infinite-data regime. Dataset suitability also depends on language coverage, transcription quality and preprocessing consistency, not only nominal hours. Related native multimodal scaling studies likewise use mixtures containing private data (Shukor et al., 2025b;a); this is context for the design choice, not a substitute for reporting its reproducibility boundary.

The speech pool was independently collected from the Internet and processed through cleaning, annotation and selection pipelines, rather than assembled from public ASR benchmark datasets. No benchmark split was deliberately incorporated into pretraining, and the main recognition results use no benchmark-specific fine-tuning. Independent collection describes provenance; it does not by itself rule out web-hosted copies or other incidental overlap. The checks and evidence available for this study are specified in Appendix E.

Architecture. The acoustic input has 128 mel bins at 100 frames per second before frontend downsampling. Three convolutional frontend layers and an overall downsampling factor of eight produce a 12.5 Hz acoustic sequence. The standard encoder is bidirectional and the decoder's Prefix attention is causal. The main grid jointly varies width and encoder/decoder depth along a balanced capacity path. Consequently, it does not separately identify a continuous width law and a continuous depth law. The dedicated allocation and shape interventions probe these confounds. Text embeddings and the output projection are tied. Attention computation uses FlashAttention-2 where supported (Dao, 2024). Both total and non-embedding counts are retained rather than treating them as interchangeable measures.

Table 9: Shared optimization conventions for the primary cold channel. Every horizon is an independent schedule; intermediate checkpoints are not new budget endpoints. Scale comparisons characterize this common recipe, not independently hyperparameter-optimized models at each size.

Quantity	Setting
Initialization	Random for every trainable component
Primary grid seed	42; supplementary seeds in Appendix M
Optimizer / peak learning rate	AdamW / 6×10^{-4}
Weight decay / warmup	0.1 / 3% of that run's horizon
Learning-rate schedule	Full-horizon cosine
Distributed training	Eight GPU ranks per run
Precision / activation checkpointing	Mixed precision / configuration-specific
Scale grid and main L allocation	One text + one ASR microbatch per update
Earlier B exposure-matched allocation	One text + two ASR microbatches per update
Per-sample decoder-position cap	8,192
Packing buffer / audio window	4,096 / 512
Maximum examples per pack	96
Validation / ordinary save interval	500 updates in the scale grid

Aggregate modality-specific pack budgets are varied only in the identified mixture experiments. A pack budget is distinct from a per-sample context-length cap: a long eligible example can be emitted as a singleton even when it exceeds an aggregate budget. Loss weights remain fixed; changing aggregate pack budgets changes exposure and gradient noise, not the definition of the per-modality token mean.

Interrupted runs restore model, optimizer, scheduler, random-number states, sampler, data position, and exposure ledger. They retain their original scientific identity and learning-rate horizon. A recovery or an evaluation repair is not an additional statistical replicate. Only completed, admissible endpoints enter the numerical tables; active training is not projected into a claimed final result.

Table 10: Selected completed runs under the shared formal evaluation protocol. These rows span different compute and initialization settings and are not a compute-matched ranking. CER/WER are percentages; LibriSpeech uses the fixed duration-filtered test selections. Core-v2 is a raw score.

Configuration	ASR NLL	Text NLL	CER	WER_c	WER_o	Core-v2
463M / central mixture	0.1675	2.3446	4.48	3.33	8.38	0.1088
787M / wide geometry	0.1483	2.2065	4.25	2.93	7.12	0.1548
427M / deep shape	0.2377	2.4742	7.84	4.63	11.71	0.0894
489M / wide shape	0.2431	2.5275	7.50	4.89	12.59	0.0763
489M / wide, half budget	0.3228	2.6239	10.52	6.97	17.10	0.0736
911M / random–random	0.1903	2.2754	5.41	4.02	10.25	0.1289
911M / random–pretrained	0.1718	2.1112	4.90	3.69	9.27	0.2060
M / $8C_0$ (Z001)	0.1397	2.1244	4.24	2.59	6.50	0.1544

E Evaluation definitions and limitations

Primary likelihood responses. The shared validation manifests contain 211 text chunks and 256 speech utterances, with 399,158 and 14,230 supervised targets respectively. Their NLLs are measured in nats per supervised target. The fixed optimization weights give equal weight to the two modality means, not to every target pooled across modalities. A pooled validation mean would give the speech targets only approximately 3.44% of the weight. This is why the paper uses two responses rather than presenting their sum as a preference-free measure of quality.

Recognition evaluation. Recognition scores measure direct transfer from pretraining to the benchmark test sets, not performance after training or fine-tuning on their corresponding training partitions. AISHELL-1 test contributes all 7,176 utterances. LibriSpeech's official test-clean and test-other splits contain 2,620 and 2,939 utterances; the 30-second model limit excludes 9 and 7 respectively. All comparisons use the same retained 2,611 and 2,932 utterances, greedy decoding, and fixed normalization. We report micro CER for AISHELL and micro WER for LibriSpeech. These are duration-filtered LibriSpeech results, not the unqualified official full-split scores. Conditional transcript NLL is additionally computed with teacher forcing. Where a combined test NLL is reported, targets, including the evaluator's EOS convention, determine its weights.

Text evaluation. The fixed 22-task suite contains zero- and few-shot tasks, including HellaSwag (Zellers et al., 2019), LAMBADA (Paperno et al., 2016), ARC (Clark et al., 2018), CommonsenseQA (Talmor et al., 2019), PIQA (Bisk et al., 2020), CoQA (Reddy et al., 2019), BoolQ (Clark et al., 2019), SQuAD (Rajpurkar et al., 2016), and the COPA, OpenBookQA, Winograd, WinoGrande, Jeopardy, LSAT-AR and selected BIG-bench tasks in the shared DCLM-style suite (Li et al., 2024). HellaSwag zero-shot and 10-shot are separate tasks. The internal Core-v2 statistic is

$$S_{\text{Core}} = \frac{1}{22} \sum_{j=1}^{22} \frac{a_j - b_j}{1 - b_j}, \tag{14}$$

where a_j is task accuracy and b_j its fixed reference baseline. Continuation exact-token-match accuracy is used for the applicable language-modeling tasks; their NLL is not inserted into this formula. The score can be negative. This is a DCLM-style within-study aggregate, not a cross-paper leaderboard score. Appendix R gives all task definitions and systematic results across scale, budget, and mixture.

Interpretation. No ASR-only result is taken as evidence of general environmental-audio understanding, instruction following, or speech generation. Fixed manifests, integrity hashes, and consistent evaluators prevent silent protocol changes but do not prove that an internal web-derived training corpus is free of evaluation contamination. The corpus disclosure and contamination limitations must therefore accompany the reported task scores.

What the existing overlap checks establish. Archived bounded transcript probes compared the 12,719 test utterances with 124,999, 125,000, and 22,000 sampled training rows, retrieving respectively one, zero, and zero normalized-transcript candidates. A separate 241,999-row probe against

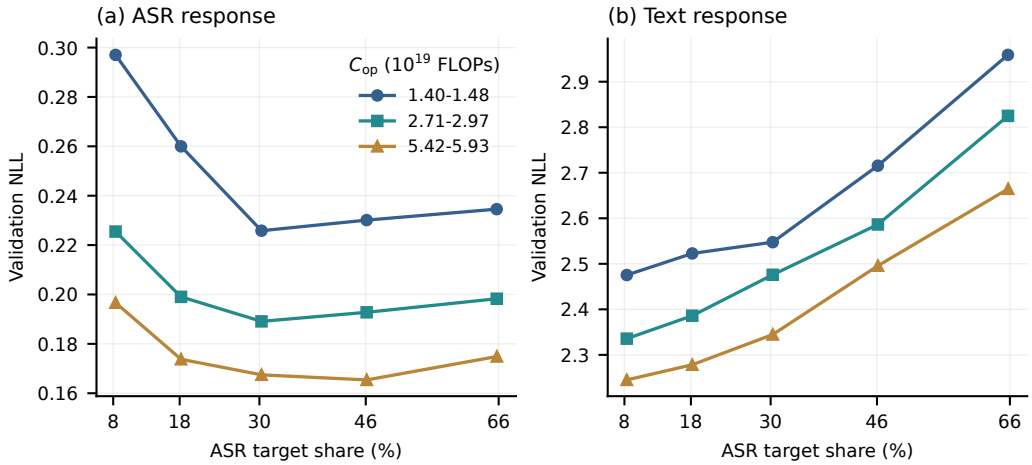


Figure 6: Speech and text respond differently to mixture allocation. Fifteen completed B (463M) trajectories cover five ASR target shares at each of three budget bands. The legend gives the actual arithmetic cost range within each band in 10^{19} FLOPs; budgets are not exactly equal across all shares. Lines connect measurements, not a fitted continuous optimum.

512 development utterances retrieved none. These probes are not random full-corpus samples, and some coverage overlaps. They do not establish a contamination rate; the test candidate lacks waveform-level verification. Thus, source documentation and consistent evaluation support within-study interpretation but do not certify decontamination. The numerical supplement preserves the bounded audit counts and scope without disclosing the internal corpus inventory or manifest identities.

F Compute accounting and experimental coordinates

The arithmetic convention is held fixed: one multiply–accumulate counts as two FLOPs, and training is approximated as three times forward arithmetic. Analytic operation counting and the two-times-forward backward approximation follow established compute-scaling practice (Kaplan et al., 2020; Hoffmann et al., 2022). Our profiler separately accounts for frontend, encoder, adapter, decoder, and vocabulary operations, using realized ledger totals converted to mean per-update exposures under fixed attention and packing approximations. We call the resulting quantity base algorithmic training compute, C_{op}. Activation recomputation, kernel implementation, hardware utilization, and replayed failed work are not included. GPU-hours are an implementation-dependent resource measure rather than the estimand of the scaling comparisons; they are not asserted equal by a small C_{op} mismatch.

The 33-point balanced scale grid spans approximately 9.04×10^{18} to 2.89×10^{20} base FLOPs (0.25–$8C_0$). At a fixed budget, changing model size alters the affordable exposure; at a fixed size, capacity allocation and modality packing also change the number of targets processed (Figure 7). We therefore compare experiments by their realized arithmetic coordinates and retain text and ASR exposure separately. The M allocation–mixture references are approximately $0.749C_0$ and $1.498C_0$; they are not relabeled as the scale grid's $1C_0$ and $2C_0$ budgets.

The speech ledger records mel frames, encoder positions, decoder positions, transcript targets, source examples seen, and sampled audio hours. These are not independent scaling variables. Across the 33 balanced scale–budget endpoints, the text/transcript target ratio has a sample coefficient of variation of approximately 0.089%. Thus this grid varies overall exposure much more than modality mixture. Separate mixture and allocation interventions are needed to distinguish their predictive contributions.

The numerical catalog contains 141 completed representative configurations, of which 140 have arithmetic coordinates. The text-only reference retains its likelihood but has no reported cost. The

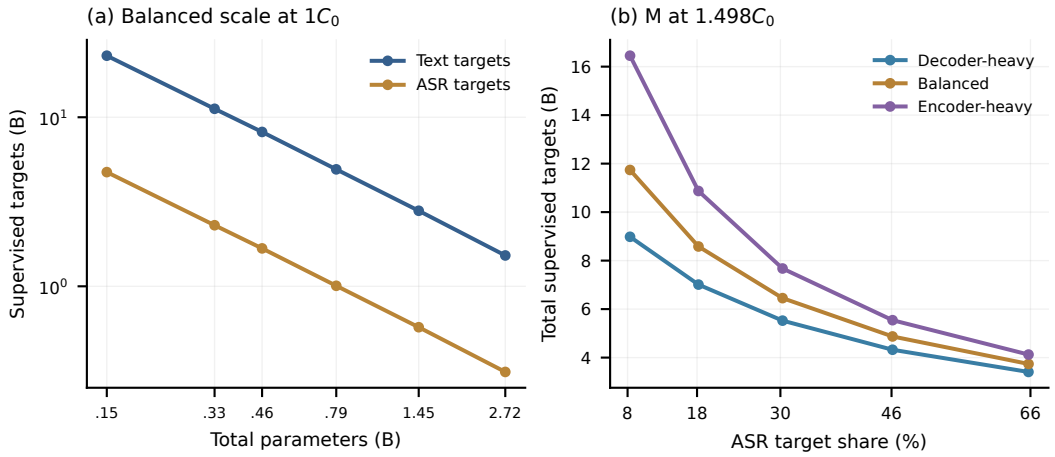


Figure 7: Equal compute does not imply equal modality exposure. Left: the six balanced sizes at approximately $1C_0$ see different numbers of text and ASR targets. Right: all fifteen M allocation–mixture cells at approximately $1.498C_0$ have different total target counts despite matched base compute. Coordinates use realized terminal ledgers; lines connect measurements.

primary 116-point fit is a subset of this catalog; architectural controls and the additional mixture endpoint remain distinct experiments.

G Allocation transfer and additional controls

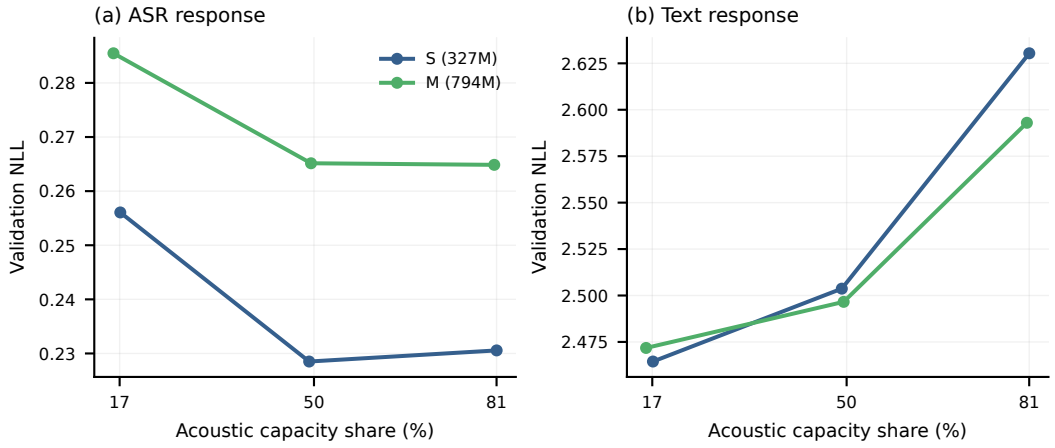


Figure 8: Allocation at S (327M) and M (794M), approximately $0.5C_0 = 1.809 \times 10^{19}$ base FLOPs. The S family has an interior ASR-NLL minimum; balanced and encoder-heavy M are nearly tied. Text NLL increases with acoustic capacity share in both families. All six points are completed trajectories; lines are visual guides.

For 327M, the $4/26$, $12/16$, and $20/6$ configurations give ASR NLL 0.2561, 0.2285, and 0.2306, and text NLL 2.4645, 2.5037, and 2.6304. For 794M, the $5/35$, $16/21$, and $26/8$ configurations give ASR NLL 0.2855, 0.2651, and 0.2649, and text NLL 2.4718, 2.4966, and 2.5930. These results support objective-dependent allocation while limiting any claim of a fixed universally optimal split.

The crossed allocation/mixture points and the deep/wide shape pairs are retained in the full inventory. They are too sparse to support an arbitrary continuous width law or a high-order allocation–mixture interaction surface. The controlled shape pair is particularly useful because its non-embedding capacity and arithmetic costs are close, whereas the older wide-geometry family changes

several architectural attributes at once. Its separate budget progression remains a geometry stress test, not an interchangeable fifth point on the original balanced model family.

H How do pretrained priors reshape the frontier?

Component-pretraining and joint-training controls complement the from-scratch scaling study. They include acoustic and language priors trained in-house as well as externally pretrained components, allowing us to examine separately learned capabilities, subsequent adaptation, and retention. We hold architecture fixed in the initialization factorial and cross text replay with decoder learning rate in the retention factorial. These experiments address the effect of priors and training objectives; the cold scaling grid separately measures resource allocation within native training.

Why a different geometry for component-pretraining controls? These controls use an approximately 0.3B acoustic stack and a 0.6B language decoder to connect naturally to available pretrained components: Qwen3-0.6B on the language side and the roughly 0.3B acoustic encoders of Whisper-medium and Qwen3-ASR on the speech side (Yang et al., 2025; Radford et al., 2023; Shi et al., 2026). This component-compatible scale facilitates random/pretrained and in-house/external comparisons; it is not an additional point on the balanced scaling path. Exact acoustic internals depend on the source, giving 900.67M and 911.43M joint variants. Controlled comparisons hold the actual architecture fixed rather than treating these approximate size labels as interchangeable (Appendix L).

All recognition results below use the same test protocol as the main experiments: AISHELL-1 CER and duration-filtered LibriSpeech test-clean/test-other WER, with the same sample selection, greedy decoding, and text normalization. Our adaptation runs use the same Internet ASR pool and no public ASR training split or benchmark-specific fine-tuning. This describes training performed in this study; the upstream data of external pretrained components are separate. Text-task results use the same DCLM suite; LAMBADA continuation NLL is distinguished from the fixed training-validation text NLL. We report adaptation horizons in updates rather than identifying them with the compute-matched budgets of the cold grid. A shared update count does not imply equal operator compute across architectures or replay schedules. Where used, boldface marks the lowest observed error or NLL within a controlled comparison, not statistical significance.

Acoustic and language priors are complementary. The initialization factorial uses Qwen3-ASR E24 and Qwen3-0.6B D28 geometry (911.43M total parameters), one text and two ASR micro-batches per update, and 66,000 joint updates. Each side starts randomly or from its pretrained weights (Shi et al., 2026; Yang et al., 2025); both remain trainable. Table 11 compares all four combinations under the same terminal decoding protocol.

Table 11: **Complete encoder×decoder initialization factorial.** R/P mean random/pretrained. Architecture and the joint schedule are shared; upstream pretraining costs are excluded from this adaptation budget. CER/WER are percentages; LAMBADA is a separate text-only test NLL.

Encoder	Decoder	AISHELL CER	WER_c	WER_o	LAMBADA NLL
R	R	5.41	4.02	10.25	2.621
R	P	4.90	3.69	9.27	2.053
P	R	4.32	3.42	8.44	2.637
P	P	**3.69**	**2.95**	**7.51**	**2.048**

Pretraining the acoustic encoder improves recognition with either decoder initialization, and pretraining the decoder improves recognition with either encoder initialization. Together they reduce other WER from 10.25% to 7.51%, a 26.8% relative reduction. Their language effects differ: acoustic pretraining alone leaves LAMBADA NLL near 2.63, whereas decoder pretraining lowers it to about 2.05. The validation comparison with a random encoder agrees: a pretrained decoder changes ASR/text NLL from 0.1903/2.2754 to 0.1718/2.1112. The two priors are not interchangeable sources of generic capacity.

The corresponding four-way comparison at 33,000 updates gives the same ordering (Table 12). At this shorter adaptation horizon, using both priors reduces other WER from 13.83% to 8.29%. Ex-

tending adaptation to 66,000 updates improves all four configurations; the pretrained pair remains strongest within the tested horizons. These observations establish a finite-adaptation-budget advantage, not a crossover budget or a lifetime-compute advantage.

Table 12: **Initialization at a shorter adaptation horizon.** Same 911.43M geometry and text1:ASR2 schedule as Table 11, trained for 33,000 updates. R/P denote random/pretrained weights. Errors are percentages.

Encoder	Decoder	AISHELL CER	WER_c	WER_o	LAMBADA NLL
R	R	7.3975	5.5375	13.8349	3.0125
R	P	6.4000	4.4449	11.5457	2.1675
P	R	4.6447	3.7813	9.3594	3.0252
P	P	**4.4509**	**3.2893**	**8.2891**	**2.1563**

Smaller decoder updates do not replace language replay. A separate 2×2 factorial crosses text replay with decoder learning rate, fixing ASR exposure and 66,000 updates (Table 13). Its 900.67M native-encoder architecture differs from the initialization factorial despite identical E/D layer counts; Appendix L details the acoustic internals. The two factorials must not be pooled as a single-factor comparison.

Table 13: **Retention is an objective-allocation effect.** All four adaptations match ASR exposure and updates. Replay adds text compute, so this is not a fixed-total-compute mixture test. Drift is relative decoder-parameter L_2 distance from initialization.

Text replay	Decoder LR multiplier	Text NLL	ASR NLL	Drift
0	1.0×	11.0794	0.19197	0.6759
1	1.0×	2.1042	0.15087	0.6037
0	0.1×	8.8126	0.17174	0.2961
1	0.1×	**1.9483**	**0.14924**	0.2540

Within the retention factorial, a tenfold smaller decoder rate more than halves drift without replay, yet text NLL remains 8.8126. Replay at the original rate gives 2.1042 and improves ASR; combining replay and the smaller rate gives 1.9483 and the lowest ASR loss. Reducing decoder drift without replay therefore does not recover the text-modeling benefit of replay; proximity to initialization is not an adequate substitute for continued language supervision under this protocol.

The full-test results support this distinction (Table 14). Replay improves all three recognition errors at either decoder rate, while substantially lowering LAMBADA NLL. With replay, lowering the decoder rate improves CER and language retention, but not both LibriSpeech errors. Thus the best validation-loss setting is not uniformly best on every external metric.

Table 14: **Recognition and language retention under one evaluation protocol.** The four configurations are those of Table 13; LAMBADA is an external text-task NLL, not the validation loss in that table.

Text replay	Decoder LR	AISHELL CER	WER_c	WER_o	LAMBADA NLL
0	1.0×	4.5177	3.6707	8.8138	7.8795
0	0.1×	4.4404	3.5887	8.3902	6.1371
1	1.0×	4.2820	**3.0853**	**7.6481**	2.0458
1	0.1×	**4.1712**	3.1025	7.7073	**1.8049**

Initialization, temporary freezing, and adaptation budget. Tables 15 and 16 hold the 900.67M native acoustic-encoder/Qwen-decoder architecture fixed and compare random initialization, pretrained decoder initialization, and freezing the pretrained decoder during the first 10% of updates. The acoustic sequence entering the decoder is 12.5 Hz throughout. All rows are evaluated on the same complete test selections and text suite.

Table 15: **Fixed-architecture ASR-only adaptation.** Updates are the complete training horizon. Temporary freezing changes optimization, not the evaluation protocol. All recognition errors are percentages; LAMBADA is continuation NLL.

Updates	Decoder strategy	CER	WER_c	WER_o	LAMBADA NLL
33k	Random	6.3533	5.2954	12.6178	11.8950
33k	Pretrained	5.6488	4.4297	10.8264	7.7083
33k	Freeze 10%	5.4102	4.1131	10.2503	7.8910
66k	Random	4.8585	4.0121	9.6188	12.0369
66k	Pretrained	4.5177	3.6707	8.8138	7.8795
66k	Freeze 10%	4.7144	3.5925	8.4952	8.0405

Table 16: **The same architecture with text replay.** The schedule uses one text and two ASR micro-batches per update. Comparisons within each horizon isolate decoder initialization and temporary freezing; comparisons with ASR-only add text exposure and compute.

Updates	Decoder strategy	CER	WER_c	WER_o	LAMBADA NLL
16.5k	Pretrained	7.4996	5.0017	13.0814	2.2709
16.5k	Freeze 10%	6.9985	4.6680	12.1008	2.2824
33k	Random	6.4258	4.4830	11.4503	3.0317
33k	Pretrained	5.3615	3.7546	9.2678	2.1557
33k	Freeze 10%	4.9625	3.3542	8.7050	2.2257
66k	Random	4.7535	3.3294	8.5104	2.5743
66k	Pretrained	4.2820	3.0853	7.6481	2.0458
66k	Freeze 10%	4.0834	2.8794	7.2551	2.0845

Pretrained language initialization improves recognition at both 33k and 66k updates. Temporary freezing can improve recognition further, but its language-retention effect is not uniformly beneficial. The contrast between the two schedules is larger: ASR-only adaptation leaves high LAMBADA NLL even with a pretrained decoder, whereas text replay preserves useful language modeling. This complements the learning-rate factorial by separating the initial prior, temporary protection of that prior, and continued text supervision.

Acoustic-source transfer and short-budget adaptation. The initialization factorial controls architecture; comparisons across pretrained acoustic sources instead evaluate transfer packages. Table 17 separates two such settings. With the in-house acoustic prior and text replay, initializing the decoder from Qwen improves recognition and language modeling together. With Whisper-medium (Radford et al., 2023) and ASR-only adaptation, the language prior also improves recognition, but the resulting text NLL remains high. Strong recognition after adaptation does not by itself demonstrate preserved general language capability.

Table 17: **Additional acoustic-prior controls.** Each pair shares its acoustic source and adaptation schedule. All rows use the common full-test protocol. Comparisons between pairs do not isolate encoder architecture: upstream data, initialization, interfaces, and adaptation schedules differ. R/P denote random/Qwen-pretrained decoder weights.

Acoustic source / adaptation	Decoder	CER	WER_c	WER_o	LAMBADA
In-house acoustic prior; 66k updates; text1:ASR2					
In-house	R	4.0968	3.1177	7.7321	2.5994
In-house	P	3.6243	2.7230	6.6943	2.0122
Whisper-medium; 16.5k updates; ASR-only					
Whisper-medium	R	5.7061	4.6413	10.9543	11.3935
Whisper-medium	P	5.0475	4.0845	9.8115	8.0899

Native learning and component reuse answer different budget questions. The component-reuse controls are not restricted to externally supplied models: they include an in-house acoustic

prior (Table 17) and a 596M text-only decoder trained in-house for 66,000 updates (Table 18), each used in subsequent joint training. These are distinct comparisons, not a single factorial with both in-house priors crossed. With this text source fixed, adding an acoustic prior after 33,000 joint updates reduces ASR NLL by 26.7%, while text changes only from 2.2639 to 2.2523 (Appendix J). This isolates an acoustic-prior benefit rather than confounding it with a different language source.

Table 18: **A shared in-house language prior.** All rows initialize the decoder from the same 66k-update text-only checkpoint. R/P refer to random/Qwen3-ASR-pretrained acoustic weights. Joint adaptation uses text1:ASR2; upstream text pretraining is additional to the reported adaptation horizon.

Joint updates	Encoder	AISHELL CER	WER_c	WER_o	LAMBADA NLL
33k	R	6.6835	4.9579	12.6331	2.5298
33k	P	4.6285	3.4762	8.6363	2.5288
66k	R	5.0857	3.7375	9.9031	2.3939

A total-cost comparison would require $C_{\text{total}} = C_{\text{audio prior}} + C_{\text{text prior}} + C_{\text{joint}}$. For externally supplied priors, upstream training costs are unavailable here and are not included in the reported joint-training budgets. Equal adaptation compute would therefore not establish equal total compute; reuse can also amortize priors across downstream systems. We use the component-pretraining controls to study how prior learning changes adaptation and retention, and the cold experiments to characterize native scaling. These complementary questions do not require assuming that external pretraining is cost-free.

H.1 Native and staged training under explicit compute budgets

Relation to the broader paradigm comparison. Comparing native training with staged component reuse at a common total budget is an important question. Each route may favor different capacity allocations, modality exposures and stage-wise budgets: transferring a configuration optimized for one route to the other can leave the latter far from its attainable frontier. Characterizing these within-route trade-offs provides a foundation for comparing the paradigms, and our native-scaling study contributes to that foundation. The following comparisons examine particular recipes, rather than each paradigm's optimized frontier, and remain supplementary to the native-scaling study.

Using components pretrained in-house makes upstream computation traceable. We compare fresh native joint training with a staged recipe that separately learns acoustic and language priors, transfers their weights, and then jointly trains the same 900.67M Prefix architecture. Both joint phases use one text and two ASR microbatches per update and the common recognition/text evaluation protocol. The staged joint phase starts with a fresh optimizer and its own learning-rate schedule; it is not a continuation of the native trajectory.

Under the operator-counting convention of Section 3.2, the native 66,000-update run costs approximately 5.924×10^{19} FLOPs. The in-house acoustic-source and language pretraining costs are approximately 2.186×10^{19} and 1.674×10^{19} FLOPs, respectively. Charging both stages to the same total budget leaves approximately 2.064×10^{19} FLOPs for joint training, yielding a fixed 22,992-update staged horizon. The acoustic-source cost includes its ASR training system, not just the encoder weights transferred to the joint model. These are algorithmic cost estimates, not matched wall-clock times.

Table 19: **Native versus staged training at an approximately matched total arithmetic budget.** One completed run per recipe, with the same 900.67M joint architecture. CER/WER are percentages and use the shared duration-filtered LibriSpeech selections. Bold marks the better observed value in each column, not statistical significance.

Recipe	CER ↓	WER_c ↓	WER_o ↓	LAMBADA NLL ↓	Core-v2 ↑
Native, 66k	**4.7535**	**3.3294**	**8.5104**	2.5743	0.1354
Staged, 22,992 joint	5.0370	3.5201	8.6077	**2.5552**	**0.1394**

Realized exposure gives a staged total of 5.92393×10^{19} FLOPs versus 5.92411×10^{19} for native training, a 0.003% difference under this convention. At this point, native training has slightly lower recognition errors, while staged training has slightly better LAMBADA NLL and Core-v2. Neither recipe dominates all reported metrics.

Which components should receive upstream compute? We also compare pretraining one component with pretraining both. The joint architecture and training recipe are unchanged. At each of two additional total budgets, the single-component route receives 66,000 joint updates; charging the second component's pretraining cost leaves fewer joint updates for the two-component route. All transferred priors in these comparisons are trained in-house. The acoustic-only route transfers an encoder learned in an ASR system into the same Prefix joint architecture; it does not change the joint fusion interface.

Table 20: **One versus two pretrained components at matched total budgets.** Each pair has its own total arithmetic budget, including upstream pretraining. All runs use the same 900.67M joint architecture. Bold identifies the better observed value within each pair; differences are not significance claims. CER/WER are percentages.

Pretrained component(s)	Joint updates	CER ↓	WER_c ↓	WER_o ↓	LAMBADA NLL ↓
Total budget $\approx 8.110 \times 10^{19}$ FLOPs					
Encoder only	66,000	**4.0968**	3.1177	**7.7321**	2.5994
Encoder + LLM	47,349	4.3268	**3.1063**	7.7760	**2.4018**
Total budget $\approx 7.598 \times 10^{19}$ FLOPs					
LLM only	66,000	**4.4290**	**3.1215**	**7.9038**	**2.3635**
Encoder + LLM	41,642	4.4395	3.1292	7.9591	2.4348

At the first budget, adding language pretraining improves LAMBADA NLL, while recognition changes are mixed. At the second, the LLM-only route has slightly lower recognition errors and LAMBADA NLL than pretraining both components; its Core-v2 is also higher (0.1389 versus 0.1336). Thus, pretraining an additional component does not uniformly improve the observed endpoint once its cost is charged to the same budget. These outcomes are consistent with a trade-off between acquiring component priors and retaining sufficient joint-training compute, rather than establishing its causal mechanism.

Fixed joint-training duration answers a different question. Table 21 reuses the 66,000-update runs to isolate the comparison at a common joint-training horizon. Relative to random initialization, acoustic pretraining gives the largest recognition improvement among these runs, while language pretraining gives the lowest LAMBADA NLL. These comparisons do *not* match total compute: their upstream costs differ. They describe the benefit of initialization after the same joint-training duration, not the efficiency of the complete training pipeline.

Table 21: **Common joint-training horizon, different upstream budgets.** All three runs receive 66,000 joint updates with the same architecture and joint-training recipe. These are the same runs used above, not additional repetitions. CER/WER are percentages.

Initialization	Total FLOPs / 10^{19}	CER ↓	WER_c ↓	WER_o ↓	LAMBADA NLL ↓
All random	5.924	4.7535	3.3294	8.5104	2.5743
Encoder pretrained	8.110	**4.0968**	**3.1177**	**7.7321**	2.5994
LLM pretrained	7.598	4.4290	3.1215	7.9038	**2.3635**

Each recipe is represented by one completed run. The results distinguish initialization benefits from total-budget allocation effects at this scale; they do not establish a universal ordering of native and staged training.

I ACOUSTIC-CONTEXT AND SUPERVISION BOUNDARIES

The boundary comparisons in Section 4.4 distinguish three interventions: relocating acoustic context, removing text supervision, and removing speech supervision. They are separate controlled comparisons, not additional fitting points for the joint-training law.

I.1 ACOUSTIC CONTEXT WITH AND WITHOUT A DEEP ENCODER

The encoder-free architecture retains a convolutional acoustic frontend and projection, but no deep acoustic Transformer encoder. Its 36-layer decoder uses either a causal audio prefix or an implementation in which audio tokens within a sample attend bidirectionally. Audio queries cannot access transcript tokens; transcript attention remains causal and packed samples remain isolated. Table 22 holds the E0/D36 architecture, joint schedule, and training horizon fixed within each pair. Bidirectional-prefix checkpoints use compatible greedy decoding with prefix recomputation; the test selections and normalization are the same as for causal-prefix models.

Table 22: Encoder-free audio-prefix comparison. CER/WER are percentages; LAMBADA is continuation NLL. The three horizons are independently trained, not intermediate checkpoints from one schedule.

Updates	Audio prefix	CER	WER_c	WER_o	LAMBADA
31,250	Causal	8.90	5.74	16.65	3.5125
31,250	Bidirectional audio	7.32	4.98	12.80	3.5087
62,500	Causal	6.82	4.75	12.97	2.9862
62,500	Bidirectional audio	5.70	4.00	10.65	3.0132
125,000	Causal	5.64	3.99	11.23	2.5473
125,000	Bidirectional audio	4.74	3.56	9.11	2.5691

The bidirectional-prefix implementation improves all three recognition metrics at every horizon, with 16–18% relative CER reductions. LAMBADA changes by less than 1% in either direction. This locates useful acoustic contextualization inside the shared decoder rather than requiring a dedicated deep encoder. Because the mask and attention backend change together, the result characterizes this implementation-level intervention, not an isolated mask effect.

Compute-matched replication. To complement the equal-horizon comparison, we train four seeds for each E0/D36 variant under a common operator-proxy budget of 1.4071×10^{19} FLOPs (approximately $0.389C_0$), using SDPA for both. Accounting for the additional bidirectional attention gives 25,301 causal and 25,070 bidirectional updates, each with its own complete schedule. This matches estimated arithmetic work, not exact exposure. Table 23 shows that all four paired seeds improve on AISHELL and both LibriSpeech splits with bidirectional acoustic context. The text aggregate is not uniformly ordered across pairs.

Table 23: Compute-matched encoder-free comparison at approximately $0.389C_0$: mean $\pm$ sample SD over seeds 42–45. Recognition errors are percentages; SDs are percentage points. Both implementations use SDPA and the same greedy prefix-recomputation evaluator.

Audio prefix	ASR NLL	CER	WER_c	WER_o	Core-v2
Causal	0.2839 ± 0.0050	9.52 ± 0.10	6.40 ± 0.07	17.48 ± 0.24	0.0806 ± 0.0115
Bidirectional	0.2552 ± 0.0050	7.88 ± 0.25	5.41 ± 0.09	14.12 ± 0.23	0.0778 ± 0.0164

The complementary E14/D18 experiment varies the encoder's temporal context while retaining the causal decoder interface (Table 24). Full bidirectionality improves recognition relative to both block-causal and strict-causal encoders. Together the comparisons support the value of offline acoustic context in either architectural location.

Table 24: Encoder temporal-context comparison at 125,000 updates, with the same E14/D18 architecture and joint schedule. CER/WER are percentages.

Encoder context	CER	WER_c	WER_o	LAMBADA
Block-causal	7.20	4.74	12.74	3.0552
Bidirectional	6.11	3.87	10.16	3.0183
Strict-causal	7.15	4.62	13.31	3.0783

I.2 THE ASR-ONLY ENDPOINT

All four capacity allocations in Table 25 start from random initialization. Each pair has the same architecture, 31,250 updates, and 837,905,008 supervised ASR targets. Joint training adds 2,046,383,118 text targets; it does not replace ASR examples. This is an ASR-exposure-matched comparison, whereas the main mixture grid is compute-matched. Adding text improves both LibriSpeech splits for all four allocations and greatly improves text modeling. AISHELL is not uniformly improved: the encoder-free pair has a small CER increase. The evidence supports complementary text supervision, not a claim that every recognition metric must improve under every allocation.

Table 25: Cold ASR-only versus joint text1:ASR2 training at matched ASR exposure. Joint training adds text compute. CER/WER are percentages; Core-v2 is a raw score.

E/D	Objective	CER	WER_c	WER_o	LAMBADA	Core-v2
0/36	ASR only	8.41	6.70	16.76	12.4066	-0.0126
0/36	Joint	8.90	5.74	16.65	3.5125	0.0766
6/28	ASR only	7.27	5.94	14.56	11.8005	-0.0197
6/28	Joint	7.27	5.24	13.51	3.5708	0.0795
14/18	ASR only	7.23	6.06	14.18	11.2097	-0.0201
14/18	Joint	7.10	5.05	12.67	3.7148	0.0714
26/3	ASR only	7.65	6.03	13.65	10.8601	-0.0460
26/3	Joint	6.78	5.06	12.00	4.4664	0.0397

I.3 THE TEXT-ONLY ANCHOR

The text-only anchor and the B E14/D18 joint models share the same randomly initialized language architecture: width 896, 18 decoder layers, and 299,256,448 decoder parameters including tied embeddings. All use 125,000 updates and exactly 8,185,670,182 supervised text targets. The joint models add a 163M acoustic path and either one or two ASR microbatches per text microbatch. Total parameter counts and total compute therefore differ, while language capacity and generic-text exposure are held fixed.

Table 26: Same-decoder text-only anchor. All rows share 299M decoder parameters and 8.186B text targets. N is total parameters in millions; Y is ASR targets in billions; HS is zero-shot HellaSwag accuracy (%).

Schedule	N	Y	Text val	LAMBADA	HS	Core-v2
Text only	299.3	0.000	2.3322	2.8043	32.25	0.1169
Text1:ASR1	462.6	1.676	2.3586	2.9079	30.96	0.1223
Text1:ASR2	462.6	3.352	2.3210	2.7809	31.61	0.1106

Relative to text-only training, text validation NLL changes by $+1.13\%$ for text1:ASR1 and -0.48% for text1:ASR2. LAMBADA and the aggregate scores likewise remain in a similar range but are not uniformly ordered. Thus substantial speech learning need not displace the shared decoder's text-likelihood capability at this matched language capacity and exposure. This anchor does not establish zero interference for all tasks or equal-total-compute efficiency. Its 22-task suite uses the same task

datasets as Appendix R; complete values are included in `data/boundary_controls.json` alongside the other boundary results.

Total supervision and transcript controls. We further separate generic-text exposure from transcript supervision (Table 27). A generic-only run uses 176,180 updates to match the joint text1:ASR2 run's total 11.537B supervised targets. A second run retains the original 8.186B generic-text targets and 3.352B transcript targets, but removes audio entirely. It uses 125,000 updates and adds the separately normalized generic-text and transcript losses, retaining the two text streams rather than merging their token normalization. All controls use the same language architecture and seed 42.

Table 27: Text-supervision controls sharing a 299M language architecture. The first three rows match generic-text exposure T; the fourth matches total supervision $T + Y$ using generic text; the fifth retains generic/transcript streams but removes audio. HS is zero-shot HellaSwag accuracy (percent). Compute, updates and loss organization are not all matched.

Configuration	Updates	Text val	LAMBADA	HS	Core-v2
Generic only (same T)	125,000	2.3322	2.8043	32.25	0.1169
Joint, text1:ASR1	125,000	2.3586	2.9079	30.96	0.1223
Joint, text1:ASR2	125,000	2.3210	2.7809	31.61	0.1106
Generic only (same $T + Y$)	176,180	2.2964	2.7125	32.94	0.1313
Generic + transcripts, no audio	125,000	2.3486	2.7826	31.67	0.0992

Allocating all supervision to generic text gives the strongest text scores among these controls. Removing audio while preserving transcript supervision does not consistently improve text capability over joint training: LAMBADA NLL and HellaSwag are close, while validation text NLL and Core-v2 are worse. The two additional runs differ in update count and loss organization as well as text source, so their difference does not isolate a pure data-source effect. These are single-run controls, not significance tests; inputs are in `data/text_supervision_controls.json`.

J Pretrained priors and mechanism controls

The initialization comparisons ask how upstream priors change the attainable performance after a specified joint-training phase. They do not estimate total lifetime compute efficiency. For example, with an in-house text-pretrained decoder and 33,000 joint updates, replacing a randomly initialized encoder with a Qwen3-ASR pretrained encoder changes ASR/text NLL from 0.23835/2.26393 to 0.17469/2.25230, CER from 6.68% to 4.63%, and clean/other WER from 4.96%/12.63% to 3.48%/8.64%. The upstream acoustic training budget is not known. This comparison therefore supports acoustic-prior utility within that adaptation setting, not superiority of its total-compute frontier.

An ASR-only adaptation experiment also illustrates why speech accuracy is insufficient to characterize an ALM. In the retained text-replay control, adding replay changes Core-v2 from approximately -0.0102 to 0.3126 while clean/other WER improves from 3.59%/8.39% to 3.10%/7.71%. Replay changes the objective and its resource use, and these observations are not inserted as additional cold scaling endpoints.

Modality-specific Transformer and expert models (Liang et al., 2025) likewise belong to a separate mechanism channel. A matched full-MoT comparison with bidirectional rather than causal acoustic context gives ASR NLL 0.17706 versus 0.19439 and text NLL 2.34572 versus 2.35490. Its active-parameter and mask-dependent costs differ from a dense fixed-topology path. These mechanism results may help explain representation learning, but pooling them into the primary dense regression would change the estimand.

K SUPPLEMENTARY SHAPE CONTROLS AND RECOGNITION UNCERTAINTY

These controls complement the resource-allocation study with paired architectural comparisons and recognition uncertainty estimates. Their configuration sets are specified separately from the primary regression; overlapping endpoints are not independent experiments.

K.1 RECOGNITION UNCERTAINTY FOR FIXED CHECKPOINTS

The first uncertainty cohort is R071–R082; the second is R083–R088. These labels identify observations, not additional fitting populations. Tables below retain the original per-cohort bootstrap seeds and values.

Sampling uncertainty from actual decoded examples. For the first 12 later checkpoints R071–R082, we retain only dataset, source index, reference length, and edit distance from the completed decoding outputs. Their sums reproduce the published micro CER/WER exactly. We resample utterances with replacement within each test set, using 2,000 replicates and seed 20260909. Each replicate recomputes the ratio of total edit distance to total reference length, not the mean utterance error rate. Percentile 2.5/97.5 bounds give Table 28; the separate six-checkpoint extension is in Appendix K.2.

Table 28: Recognition percentages with 95% utterance-bootstrap intervals, conditional on each trained checkpoint. These are not error bars over training runs.

ID	AISHELL CER	Libri clean WER	Libri other WER
R071	6.06 [5.85, 6.28]	4.35 [4.10, 4.60]	11.47 [10.97, 12.00]
R072	8.75 [8.50, 9.03]	5.66 [5.35, 5.96]	14.13 [13.59, 14.68]
R073	6.37 [6.17, 6.60]	5.01 [4.73, 5.28]	12.49 [11.98, 13.01]
R074	6.82 [6.59, 7.04]	4.81 [4.55, 5.09]	12.51 [12.00, 13.05]
R075	6.32 [6.11, 6.54]	5.04 [4.76, 5.32]	13.29 [12.77, 13.82]
R076	9.76 [9.47, 10.06]	6.02 [5.72, 6.34]	16.11 [15.50, 16.75]
R077	7.70 [7.47, 7.94]	4.91 [4.65, 5.18]	12.16 [11.66, 12.67]
R078	5.84 [5.64, 6.05]	4.17 [3.93, 4.41]	10.36 [9.93, 10.80]
R079	5.93 [5.73, 6.15]	4.05 [3.80, 4.28]	10.60 [10.14, 11.07]
R080	5.49 [5.30, 5.69]	3.96 [3.72, 4.20]	10.01 [9.56, 10.44]
R081	6.05 [5.84, 6.27]	3.90 [3.66, 4.13]	9.84 [9.42, 10.26]
R082	9.97 [9.70, 10.27]	6.49 [6.17, 6.80]	15.98 [15.39, 16.61]

For paired contrasts, the dataset/source-index sets and reference lengths agree across models; the same bootstrap draw is used on both sides. Table 29 reports signed differences, not overlap tests between marginal intervals. These descriptive contrasts were selected after inspecting the results and are not a multiplicity-adjusted family of confirmatory tests. Utterance resampling does not model speaker dependence, uncertainty from training seeds, corpus provenance, or evaluation contamination. We do not assert primary-validation NLL intervals without per-example validation statistics. The small differences between particular NLL endpoints therefore remain single-trajectory observations.

K.2 SHAPE CONTRASTS AND THE SECOND UNCERTAINTY COHORT

Three paired shape budgets. The low/middle/high pairs are R082/R070, R068/R069, and R084/R081. Deep has 330.13M non-embedding parameters and wide 333.18M; their total counts are 427.37M and 488.76M because the vocabulary matrix scales with width. Deep's relative ASR-NLL reductions are 0.419%, 2.247%, and 3.435%, and its text NLL is lower at every pair. Recognition is not uniformly ordered: at the high budget, deep/wide CER is 7.385%/6.046%, clean WER 3.587%/3.902%, other WER 9.167%/9.838%, and Core-v2 0.10652/0.09170. This extends the repeated likelihood direction while preserving the language-specific reversal, rather than turning a sparse shape study into a continuous law.

Additional recognition uncertainty. We extract 76,314 already decoded examples for the six new endpoints, retaining only dataset, source index, reference length, and edit distance. Their summed

Table 29: Paired recognition differences in percentage points, first checkpoint minus second. R071/72/73 are balanced central/low/high; R074 is decoder-heavy central; R078 is encoder-heavy central; R079/80 are the 463M interior mixtures.

Contrast	Metric	Difference	95% interval
R071 – R072	AISHELL CER	-2.698	[-2.896, -2.508]
R071 – R072	Clean WER	-1.306	[-1.522, -1.085]
R071 – R072	Other WER	-2.657	[-3.020, -2.274]
R071 – R073	AISHELL CER	-0.317	[-0.468, -0.160]
R071 – R073	Clean WER	-0.656	[-0.829, -0.470]
R071 – R073	Other WER	-1.015	[-1.355, -0.674]
R074 – R071	AISHELL CER	+0.761	[+0.591, +0.926]
R074 – R071	Clean WER	+0.456	[+0.282, +0.636]
R074 – R071	Other WER	+1.040	[+0.678, +1.381]
R078 – R071	AISHELL CER	-0.218	[-0.375, -0.063]
R078 – R071	Clean WER	-0.183	[-0.353, -0.013]
R078 – R071	Other WER	-1.108	[-1.434, -0.803]
R079 – R080	AISHELL CER	+0.437	[+0.301, +0.587]
R079 – R080	Clean WER	+0.086	[-0.053, +0.232]
R079 – R080	Other WER	+0.595	[+0.303, +0.880]

errors reproduce each published micro metric exactly. The same utterance-level percentile method uses 2,000 draws with seed 20260910 for this new cohort. Earlier twelve-checkpoint intervals retain their original seed and values; neither cohort estimates training-seed variance, speaker-cluster uncertainty, or contamination effects.

Table 30: New checkpoint-conditional 95% utterance-bootstrap intervals. No primary-validation NLL or training-seed intervals are inferred from these decode records.

ID	AISHELL CER	Libri clean WER	Libri other WER
R083	5.84 [5.65, 6.04]	4.40 [4.14, 4.66]	11.16 [10.58, 11.90]
R084	7.39 [6.58, 8.44]	3.59 [3.37, 3.82]	9.17 [8.74, 9.62]
R085	7.04 [6.81, 7.29]	4.49 [4.23, 4.78]	12.27 [11.75, 12.81]
R086	6.20 [5.99, 6.41]	4.00 [3.77, 4.25]	10.55 [10.09, 11.02]
R087	4.82 [4.63, 5.01]	3.44 [3.22, 3.67]	8.55 [8.17, 8.96]
R088	5.41 [5.21, 5.61]	3.86 [3.61, 4.12]	10.27 [9.81, 10.77]

For the four budget pairs and the high-budget shape pair, source indices and reference lengths match on both sides; each paired difference uses shared draws. These post-hoc contrasts describe fixed checkpoint comparisons, not a multiplicity-adjusted confirmatory family. They do not justify a universal threshold below which NLL differences are insignificant.

L Initialization and retention: protocol alignment

Layer counts alone do not establish architecture equality. The initialization factorial's acoustic stack follows the Qwen3-ASR-compatible implementation; the retention factorial uses the earlier native acoustic stack. Their decoder geometry, update count, target exposure in replay-enabled arms, and validation manifests match, but their acoustic internals do not (Table 32). The two tables identify effects *within* their respective factorials. Differences between them cannot isolate initialization, learning rate, or replay without an additional architecture-matched intervention.

Each factorial has one training seed per arm. Recognition resampling cannot estimate variation in the learned models themselves; factorial effect magnitudes are observed contrasts, not seed-population confidence estimates.

Table 31: Additional paired micro-error differences, first minus second in percentage points. The first four pairs increase training budget; the final pair changes shape at nearly matched base arithmetic cost.

Pair	Metric	Difference	95% interval
R085 – R076	AISHELL CER	-2.716	[-2.932, -2.492]
R085 – R076	Clean WER	-1.531	[-1.763, -1.306]
R085 – R076	Other WER	-3.836	[-4.266, -3.443]
R086 – R072	AISHELL CER	-2.553	[-2.753, -2.350]
R086 – R072	Clean WER	-1.655	[-1.862, -1.445]
R086 – R072	Other WER	-3.579	[-3.945, -3.223]
R087 – R071	AISHELL CER	-1.233	[-1.389, -1.080]
R087 – R071	Clean WER	-0.913	[-1.089, -0.732]
R087 – R071	Other WER	-2.919	[-3.255, -2.599]
R088 – R074	AISHELL CER	-1.403	[-1.555, -1.236]
R088 – R074	Clean WER	-0.948	[-1.125, -0.781]
R088 – R074	Other WER	-2.242	[-2.594, -1.867]
R084 – R081	AISHELL CER	+1.339	[+0.546, +2.378]
R084 – R081	Clean WER	-0.315	[-0.464, -0.163]
R084 – R081	Other WER	-0.672	[-0.943, -0.399]

Table 32: Why the random-encoder/pretrained-decoder initialization cell and the replay-enabled retention control are not the same experiment. Values come from terminal configurations, run contracts, and ledgers, not from inferred model labels.

Quantity	Initialization factorial	Retention factorial
Total parameters	911,430,272	900,674,773
Encoder/decoder layers	24/28	24/28
Encoder/decoder width	1024/1024	1024/1024
Encoder FFN width	4096	3072
Encoder KV heads	16	8
Frontend downsample width	480	341
Joint updates	66,000	66,000
ASR targets	1,769,639,470	1,769,639,470
Text targets with replay	4,321,938,754	4,321,938,754
Text targets without replay	Not this comparison	0
Random encoder/pretrained decoder ASR NLL	0.1718	0.1509 (LR 1×)

M TARGETED TRAINING-SEED REPLICATION

Given the cost of repeating the full scaling grid, we complement broad configuration sweeps with targeted multi-seed experiments across model sizes, compute budgets, and allocation settings. This combines broad coverage with local replication, as in scaling studies that distinguish expensive pretraining from repeated downstream runs (Ivgi et al., 2022) or estimate seed variability using a subset of smaller models (Porian et al., 2024). Below we detail five configurations with relatively consistent outcomes; Appendix I additionally reports paired encoder-free replications. Variability depends on configuration and budget, so these examples do not establish grid-wide stability. Replications do not enlarge the primary regression population.

Detailed configuration checks. We report four seeds (42–45) at five specified operating points: B balanced at $1C_0$, S balanced at $0.5C_0$, M balanced at approximately $1.5C_0$, and L balanced and decoder-heavy (q-low) at $2C_0$. The respective full training horizons are 125,000, 85,670, 68,670, 85,460, and 68,838 updates. M uses approximately 30% transcript-target share; the other four configurations use approximately 17%. Within each configuration, architecture, training horizon, data sources, optimization, and evaluation are fixed; the seed changes initialization and data order. These supplementary replications do not add independent configurations to the 116-point regression.

Table 33: Repeatability at five specified configurations: mean ± sample standard deviation across four training seeds ($n - 1$ denominator), not standard errors or utterance-bootstrap intervals. CER/WER means are percentages and their standard deviations are percentage points.

Configuration	ASR NLL	Text NLL	CER	WER_c	WER_o
B, $1C_0$	0.1952 ± 0.0023	2.3787 ± 0.0299	6.091 ± 0.140	3.746 ± 0.107	9.686 ± 0.238
S, $0.5C_0$	0.2353 ± 0.0055	2.5111 ± 0.0409	7.237 ± 0.264	4.588 ± 0.112	11.911 ± 0.352
M, $1.5C_0$	0.1689 ± 0.0030	2.3651 ± 0.0430	4.827 ± 0.046	3.397 ± 0.054	8.750 ± 0.169
L balanced, $2C_0$	0.1888 ± 0.0037	2.2842 ± 0.0413	5.529 ± 0.161	3.728 ± 0.074	9.688 ± 0.180
L q-low, $2C_0$	0.2024 ± 0.0035	2.2814 ± 0.0422	6.067 ± 0.068	4.007 ± 0.070	10.688 ± 0.163

Table 34: Individual results for the B, S, and M configurations. All four seeds are shown. Recognition errors are percentages, using the shared duration-filtered selections; NLLs are nats per target.

Seed	ASR NLL	Text NLL	CER	WER_c	WER_o	Core-v2	LAMBADA
B balanced, $1C_0$							
42	0.1941	2.3586	6.0297	3.7527	9.5864	0.1223	2.9079
43	0.1927	2.3772	6.1633	3.7336	9.4414	0.1218	2.9237
44	0.1960	2.4214	5.9285	3.6173	9.7161	0.1032	2.8988
45	0.1979	2.3576	6.2445	3.8786	10.0004	0.1048	2.8965
S balanced, $0.5C_0$							
42	0.2285	2.5037	6.9556	4.4544	11.6315	0.0863	3.3343
43	0.2353	2.4640	7.2152	4.6451	11.6983	0.0897	3.2775
44	0.2355	2.5635	7.1847	4.5441	11.9062	0.0807	3.2983
45	0.2419	2.5132	7.5932	4.7081	12.4099	0.0905	3.3236
M balanced, $1.5C_0$							
42	0.1705	2.3434	4.8222	3.4400	8.5543	0.1237	2.8666
43	0.1645	2.3678	4.7840	3.3199	8.7184	0.1041	2.9016
44	0.1698	2.4240	4.8919	3.4019	8.9645	0.1082	2.8771
45	0.1709	2.3250	4.8108	3.4247	8.7623	0.1214	2.8764

Table 35: Individual results for the two L capacity allocations. All four seeds are shown. Recognition errors are percentages, using the shared duration-filtered selections; NLLs are nats per target.

Seed	ASR NLL	Text NLL	CER	WER_c	WER_o	Core-v2	LAMBADA
L balanced, $2C_0$							
42	0.1875	2.2834	5.3587	3.7718	9.7066	0.1267	2.6486
43	0.1848	2.2403	5.4484	3.6326	9.5654	0.1218	2.6071
44	0.1937	2.3396	5.5801	3.8004	9.9355	0.1200	2.6547
45	0.1894	2.2736	5.7290	3.7089	9.5444	0.1318	2.6517
L decoder-heavy, $2C_0$							
42	0.2069	2.2623	6.1318	4.0521	10.8035	0.1543	2.6391
43	0.1986	2.2805	6.1099	3.9129	10.4716	0.1148	2.6353
44	0.2032	2.3403	6.0469	4.0674	10.8226	0.1284	2.6083
45	0.2009	2.2425	5.9801	3.9949	10.6528	0.1291	2.6520

Seed variation is assessed within each configuration; its standard deviation is not an uncertainty estimate for other allocations or budgets. Small differences between neighboring sweep points therefore do not identify a seed-robust unique optimum.

Across these five configurations, validation losses and recognition performance are relatively stable: ASR-NLL coefficients of variation range from 1.16% to 2.32%, while text-NLL coefficients of variation range from 1.26% to 1.85%. These case studies characterize repeatability at the stated operating points, rather than uniform stability across the allocation grid or every external capability. Core-v2 varies more than the validation losses. We report every seed within each displayed configuration, without selecting a preferred seed. The mean and sample standard deviation summarize

variation across training runs; they are distinct from the checkpoint-conditional utterance-bootstrap intervals in Appendix K.

N BENCHMARK-SPECIFIC ADAPTATION OF NATIVE CHECKPOINTS

The main scaling and recognition results evaluate pretrained checkpoints directly. Here we separately examine downstream adaptation of the T (149M) and S (327M) balanced checkpoints trained at $4C_0$. Each is independently fine-tuned on AISHELL-1 train or the complete LibriSpeech training-set union (train-clean-100, train-clean-360 and train-other-500). The two benchmark adaptations never initialize from one another. This small, single-seed study is supplementary to the pretraining analysis, not an additional fitting population or a comparison with optimized ASR systems.

Protocol. We update all parameters for three epochs using a fresh AdamW optimizer (learning rate 10^{-5}, $\beta_1 = 0.9$, $\beta_2 = 0.95$, weight decay 0.01), cosine decay with 3% warmup, and global batch size 64 on eight GPUs. The seed is 42; BF16 autocast is used with FP32 master weights. Only model weights are loaded; optimizer state, schedule, step and data position are reset. AISHELL contributes 120,098 labeled training utterances; the LibriSpeech union contributes 281,241 utterances (approximately 961 audio hours per epoch). Chinese transcript whitespace is removed; English transcripts are lowercased with normalized whitespace, preserving word order and apostrophes. We select the checkpoint with lowest token-weighted validation NLL on AISHELL dev or LibriSpeech dev-clean after each epoch; all four selected checkpoints are from epoch one. Test data are not used for selection. Greedy decoding, normalization and retained test utterances match Appendix E.

Table 36: Direct evaluation versus benchmark-specific adaptation from the same $4C_0$ pretrained checkpoint. AISHELL and LibriSpeech use separately adapted models. Errors are percentages; bold marks the lower error in each pair. LibriSpeech adaptation uses the full training-set union, and tests retain the common 30-second duration filter.

Model	N (M)	AISHELL CER		Libri clean WER		Libri other WER	
		Direct	Adapted	Direct	Adapted	Direct	Adapted
T	149	8.49	**5.85**	**3.86**	4.23	10.08	**9.57**
S	327	5.33	**3.39**	3.14	**3.12**	7.92	**7.45**

AISHELL adaptation reduces CER from 8.49% to 5.85% for T and from 5.33% to 3.39% for S. LibriSpeech adaptation improves test-other for both sizes (10.08% to 9.57% and 7.92% to 7.45%); test-clean changes from 3.86% to 4.23% for T and from 3.14% to 3.12% for S. Thus native checkpoints admit useful downstream adaptation, while benefits depend on the benchmark, model size and metric. These observations concern the stated complete-training-set settings, not every adaptation regime. Numerical inputs and the recipe are in `data/downstream_adaptation.json`.

O COMPLETE ALLOCATION–MIXTURE MEASUREMENTS

This section reports all 30 allocation–mixture measurements for M and the paired changes between its two compute references. These endpoints also appear in the complete configuration catalog.

Tight joint allocation–mixture matching. Both compute references contain the complete three-allocation by five-mixture grid. Exact realized shares and arithmetic costs are given below; Figure 11 uses shared response color scales across the two budgets.

Table 37: All 30 completed M common-reference cells. Share is the realized percentage of transcript targets; C_{op} is in 10^{19} base FLOPs.

ID	Reference	E/D	ASR share	C_{op}	ASR NLL	Text NLL
R076	Lower	5/35	8.35	2.7091	0.28788	2.36073
R096	Lower	5/35	18.20	2.7092	0.23785	2.39284

ID	Reference	E/D	ASR share	C_{op}	ASR NLL	Text NLL
R074	Lower	5/35	30.32	2.7094	0.22209	2.43386
R101	Lower	5/35	46.13	2.7093	0.22250	2.59678
R075	Lower	5/35	65.70	2.7091	0.23365	2.83834
R072	Lower	16/21	8.35	2.7090	0.26848	2.36087
R092	Lower	16/21	18.21	2.7097	0.21956	2.38811
R071	Lower	16/21	30.31	2.7083	0.20647	2.45660
R093	Lower	16/21	46.13	2.7091	0.21766	2.63588
R073	Lower	16/21	65.71	2.7093	0.23145	2.92729
R077	Lower	26/8	8.35	2.7087	0.23982	2.44189
R103	Lower	26/8	18.20	2.7087	0.20791	2.49648
R078	Lower	26/8	30.31	2.7094	0.20205	2.58943
R098	Lower	26/8	46.14	2.7092	0.21369	2.70214
R083	Lower	26/8	65.69	2.7093	0.21969	3.04940
R085	Upper	5/35	8.36	5.4186	0.22225	2.21106
X003	Upper	5/35	18.21	5.4181	0.19524	2.25057
R088	Upper	5/35	30.31	5.4195	0.18797	2.33853
X008	Upper	5/35	46.14	5.4197	0.18693	2.44496
R091	Upper	5/35	65.71	5.4187	0.19868	2.68867
R086	Upper	16/21	8.35	5.4191	0.20466	2.22672
R100	Upper	16/21	18.21	5.4182	0.17654	2.24831
R087	Upper	16/21	30.31	5.4192	0.17049	2.34345
R102	Upper	16/21	46.14	5.4196	0.17102	2.49917
R090	Upper	16/21	65.72	5.4194	0.18215	2.72327
R094	Upper	26/8	8.36	5.4188	0.18958	2.33348
X012	Upper	26/8	18.21	5.4183	0.17227	2.38122
R089	Upper	26/8	30.31	5.4192	0.16711	2.46813
X010	Upper	26/8	46.15	5.4202	0.16966	2.61859
R095	Upper	26/8	65.72	5.4195	0.17350	2.79610

Table 38: Every completed same-geometry/mixture budget pair. Reductions use the lower-budget response as denominator. Pair differences are single-trajectory observations, not training-seed confidence intervals.

Lower / higher ID	$C_{\mathrm{op}}^{hi}/C_{\mathrm{op}}^{lo}$	ASR reduction (%)	Text reduction (%)
R076/R085	2.000183	22.80	6.34
R096/X003	1.999854	17.91	5.95
R074/R088	2.000261	15.36	3.92
R101/X008	2.000398	15.99	5.85
R075/R091	2.000159	14.97	5.27
R072/R086	2.000418	23.77	5.68
R092/R100	1.999555	19.59	5.85
R071/R087	2.000966	17.42	4.61
R093/R102	2.000525	21.43	5.19
R073/R090	2.000320	21.30	6.97
R077/R094	2.000485	20.95	4.44
R103/X012	2.000327	17.15	4.62
R078/R089	2.000159	17.29	4.68
R098/X010	2.000673	20.60	3.09
R083/R095	2.000305	21.03	8.31

P FULL SCALE EXTENSION AND COMPLEMENTARY GRAPHICAL VIEWS

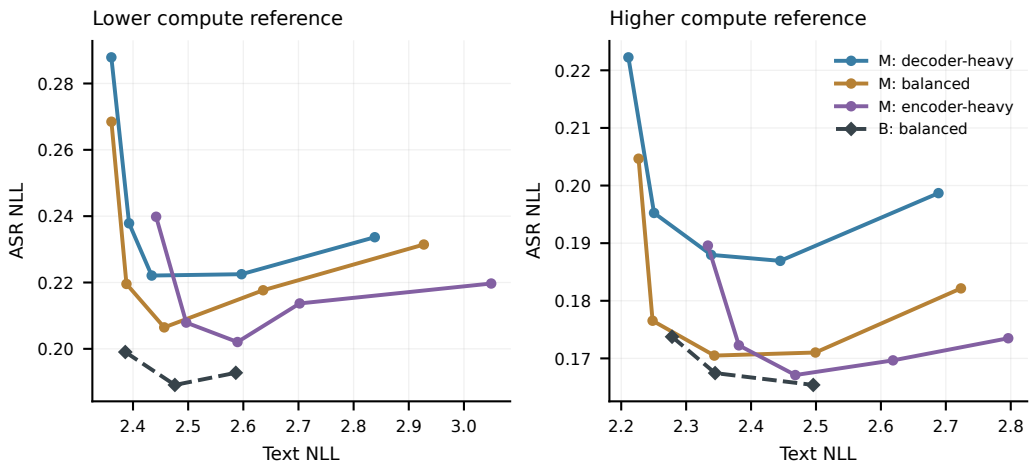


Figure 9: The complementary speech–language Pareto view. Lower-left is better. Lines connect observed mixtures within each geometry, not interpolated achievable designs. M contributes 15/15 cells at the lower/higher reference; B adds a matched 18%/30%/46% triad at each. B's 46% point is dominated within its lower-budget triad but becomes a trade-off at the higher budget. This response-space view complements the explicit mixture axis in Figure 3.

These additional views separate within-geometry mixture frontiers from scale–budget trajectories. The mixture comparison includes all 30 M cells and six B mixture endpoints.

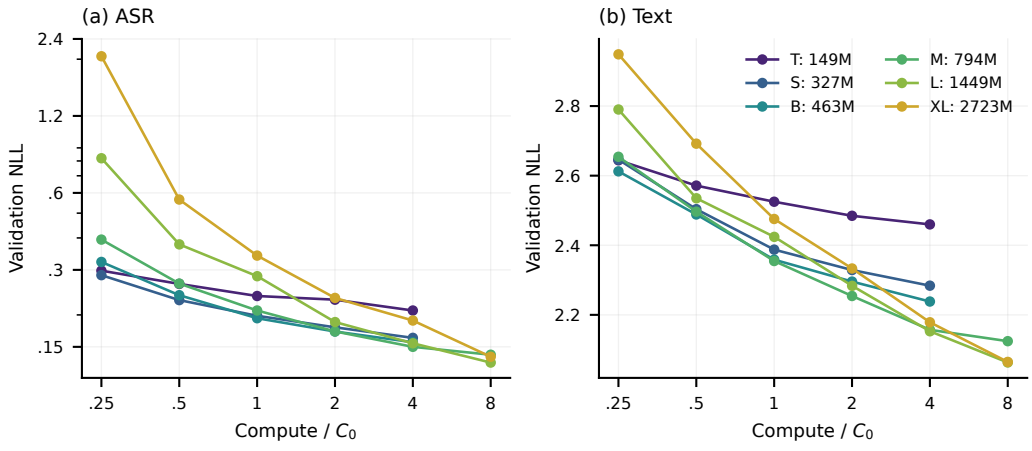


Figure 10: All 33 balanced scale–budget observations, including six completed XL horizons. Lines connect independent horizons, not intermediate checkpoints or a fitted universal curve. The logarithmic ASR axis shows the low-budget training regime. This compute-axis view complements the fixed-budget model-size comparison in Figure 2.

Table 39: All six completed XL horizons. Non-embedding/total parameters are 2.412B/2.723B, with E20/D26. CER/WER are percentages under the shared protocol. Error rates above 100% can arise from insertions and are not grounds to discard a completed run. The $2C_0$ and $8C_0$ endpoints are included.

C_{op}/C_0	Updates	ASR NLL	Text NLL	CER	$\mathrm{WER}_c/\mathrm{WER}_o$
0.25	5,811	2.0532	2.9486	101.00	96.91 / 106.01
0.50	11,621	0.5652	2.6919	21.97	15.13 / 37.44
1.00	23,243	0.3412	2.4755	12.20	7.81 / 20.28
2.00	46,485	0.2330	2.3333	7.26	4.70 / 12.24
4.00	92,970	0.1901	2.1787	5.64	3.79 / 10.09
8.00	185,941	0.1372	2.0647	3.77	2.70 / 6.82

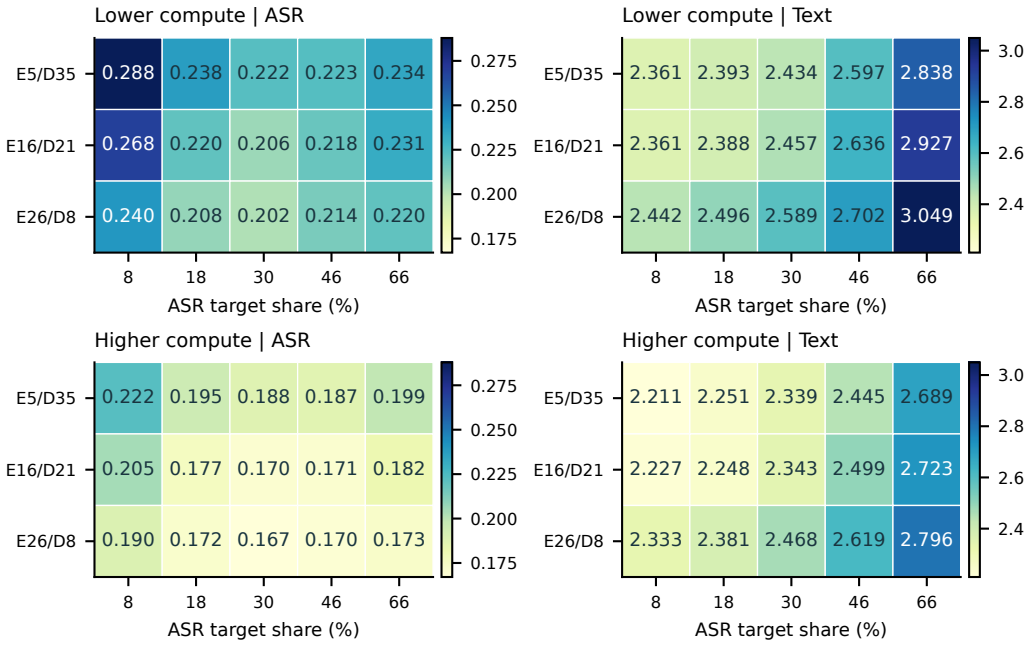


Figure 11: The complete M allocation–mixture grid: 15 cells at each of two compute references. Each response uses a shared color scale across budgets; all cells are measured endpoints. Cell values and exact target shares are retained numerically in Appendix O. All cell rectangles, labels, and colorbars are vector graphics.

Table 40: B's matched mixture triads. Pareto status concerns these measured points and both validation responses. The higher-speech point changes from dominated to nondominated within the triad; the result does not identify a continuously optimal share.

Reference	ASR share (%)	ASR NLL	Text NLL	Observed status
$C_\star$	18.2	0.19904	2.38598	Nondominated
$C_\star$	30.3	0.18911	2.47592	Nondominated
$C_\star$	46.1	0.19278	2.58638	Dominated
$C_{\star,hi}$	18.2	0.17376	2.27830	Nondominated
$C_{\star,hi}$	30.3	0.16745	2.34463	Nondominated
$C_{\star,hi}$	46.1	0.16539	2.49579	Nondominated

Q SCALING EXTERNAL TEST LOSSES

The fixed validation responses describe learning on held-out samples from our pretraining data. To examine whether the same capacity–exposure formulation also describes different evaluation distributions, we fit two complementary external responses: pooled conditional transcript NLL on AISHELL-1 and LibriSpeech, and LAMBADA continuation NLL. These checkpoints are evaluated directly, without benchmark-specific fine-tuning.

Responses and fitting protocol. The speech response pools AISHELL-1 test and the duration-filtered LibriSpeech test-clean/test-other selections from Appendix E. We divide the sum of teacher-forced negative log-likelihoods by the sum of target-token counts, rather than averaging the three dataset means. The fixed selections contain 12,719 utterances and 189,548 transcript targets including evaluator EOS. LAMBADA uses 5,153 examples and 6,494 continuation targets, excluding prompt tokens. Thus neither response is interchangeable with the primary validation loss.

We retain the same 116 configurations, five input coordinates, optimizer settings, and grouped, size-out, and forward-budget splits as the main analysis. Coefficients are fitted separately for each external response using only the training configurations of each fold. This tests the same functional form on different evaluation distributions; it is not zero-shot transfer of the original validation-loss coefficients. Grouped and size-out scores each cover 116 configurations, while forward-budget scores cover thirteen.

Table 41: Held-out prediction of external test NLLs. G/S/F denote grouped, size-out, and forward-budget tests. RMSE is in nats per target token; MAPE is percentage error relative to the observed response. Each baseline uses the same training configurations as the five-factor model.

		RMSE			MAPE (%)		
Response	Predictor	G	S	F	G	S	F
Pooled ASR	Training mean	0.1256	0.1247	0.1348	9.58	9.56	15.63
Pooled ASR	Two-factor	0.0939	0.5101	0.0589	7.36	19.87	6.75
Pooled ASR	Five-factor	0.0369	0.0472	0.0295	2.82	3.59	3.02
LAMBADA	Training mean	0.7075	0.7207	1.0897	17.56	17.63	46.11
LAMBADA	Two-factor	0.3936	0.4103	0.1566	8.39	8.62	4.94
LAMBADA	Five-factor	0.1274	0.1262	0.0872	2.68	2.70	2.44

The five-factor model predicts pooled speech NLL with 2.82–3.59% MAPE and LAMBADA NLL with 2.44–2.70% MAPE, improving over the two-factor baseline in all three protocols. Its G/S/F R^2 values are 0.910/0.852/0.560 for pooled speech and 0.967/0.967/0.915 for LAMBADA. These results extend the predictive evidence beyond the primary validation distribution. Aggregate accuracy does not imply uniform predictability across datasets: the individual AISHELL response has 10.81% forward-budget MAPE. The numerical supplement, `data/external_loss_scaling.json`, includes per-dataset speech results and all nine continuation-task fits; responses such as CS Algorithms and Dyck Languages are less well described by this law. The primary validation analysis remains unchanged.

R EXTERNAL TEXT CAPABILITIES ACROSS SCALE, BUDGET, AND MIXTURE

Coverage and protocol. We report the complete 22-task text suite for 140 speech–text configurations in the endpoint catalog, including all 116 primary regression configurations, all 33 balanced scale–budget endpoints, and the full 30-cell M allocation–mixture grid. The standalone text-only anchor is compared separately in Appendix I. All results use the same fixed task datasets and protocol, without task-specific fine-tuning. Few-shot tasks provide fixed in-context demonstrations rather than parameter updates. Component-initialization comparisons remain separate in Appendix H.

Table 42 specifies every task setting. Multiple-choice (MC) tasks select the candidate with the lowest mean continuation-token NLL. Continuation tasks report exact-token match (EM): all target continuation tokens must be predicted correctly under teacher forcing. Their NLL is averaged over continuation tokens, not examples. LAMBADA NLL is an external continuation response, distinct

from the primary fixed-validation text NLL; Appendix Q reports its separately fitted scaling relation. Few-shot demonstrations use seed 1234 and a 2,048-token prompt-plus-continuation limit. Core-v2 averages the 22 normalized task scores from Appendix E; its fixed reference baselines are listed below. HellaSwag's zero-shot and 10-shot settings contribute separate entries.

Table 42: Complete text-task protocol. MC denotes multiple-choice accuracy; EM denotes continuation exact-token match. b_j is the fixed Core-v2 reference baseline.

Task	Shots	Examples	Metrics	b_j
HellaSwag (0-shot)	0	10,042	MC	0.250
LAMBADA	0	5,153	EM / NLL	0.000
ARC-Easy	10	2,376	MC	0.250
ARC-Challenge	10	1,172	MC	0.250
CommonsenseQA	10	1,221	MC	0.403
PIQA	10	1,838	MC	0.500
CoQA	0	7,983	EM / NLL	0.000
BoolQ	10	3,270	MC	0.620
HellaSwag (10-shot)	10	10,042	MC	0.250
Language identification	10	10,000	MC	0.250
LSAT-AR	3	230	MC	0.250
SQuAD	10	10,570	EM / NLL	0.000
CS algorithms	10	1,320	EM / NLL	0.000
Dyck languages	10	1,000	EM / NLL	0.000
Operators	10	210	EM / NLL	0.000
Repeat-copy logic	10	32	EM / NLL	0.000
QA Wikidata	10	20,321	EM / NLL	0.000
Jeopardy	10	2,117	EM / NLL	0.000
COPA	0	100	MC	0.500
OpenBookQA	0	500	MC	0.250
Winograd	0	273	MC	0.500
WinoGrande	0	1,267	MC	0.500

Validation loss and external performance. Across the 140 configurations, lower validation text NLL closely tracks better external text performance: Spearman correlations are -0.949 with Core-v2, 0.987 with LAMBADA NLL, and -0.959 with zero-shot HellaSwag accuracy. The relationship also holds within the balanced scale grid and the M allocation–mixture grid (Table 43), rather than appearing only after pooling all model families. These are descriptive cross-configuration associations, not independent-run uncertainty estimates or a separate fitted downstream scaling law.

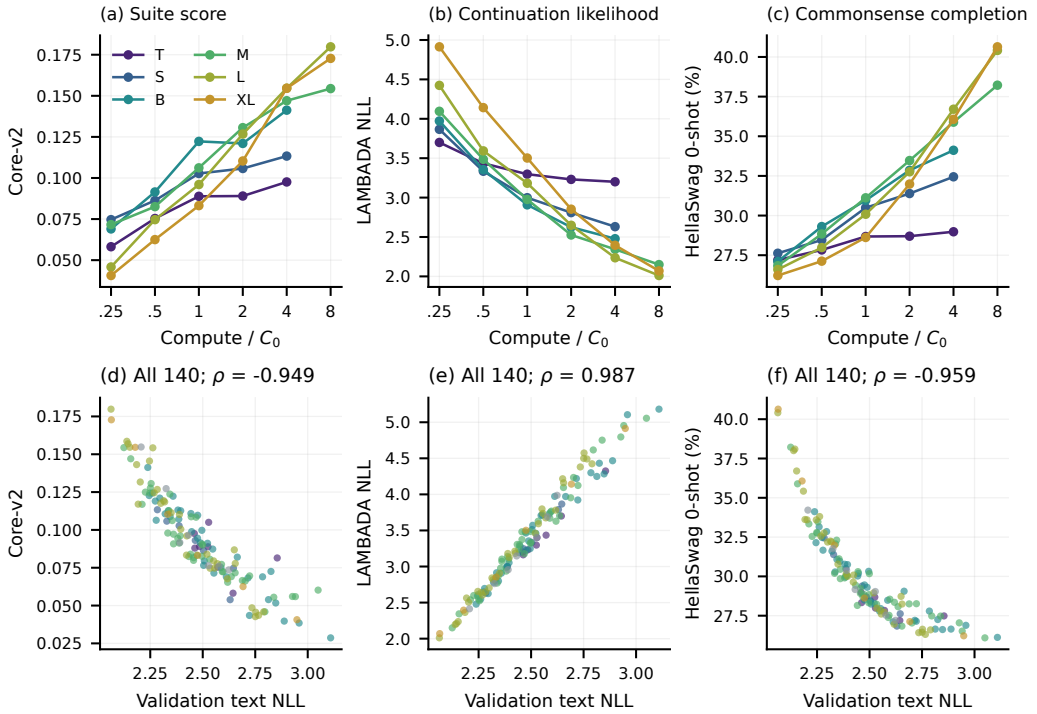


Figure 12: **Text modeling improvements transfer to external tasks.** Top: all 33 independently scheduled balanced scale–budget endpoints, with colors denoting model sizes. Bottom: all 140 text-suite-evaluated configurations against validation text NLL; gray points denote the additional 1024-width controls. Core-v2 and HellaSwag are higher-is-better; LAMBADA NLL is lower-is-better. Lines connect observations within a size family, not fitted downstream laws.

Table 43: Spearman correlation with validation text NLL. Columns are Core-v2, LAMBADA NLL and EM, and HellaSwag zero-shot and 10-shot accuracy. Negative correlations for scores and positive correlations for NLL indicate agreement.

Population	n	Core-v2	LAM NLL	LAM EM	HS 0	HS 10
Full catalog	140	-0.949	0.987	-0.978	-0.959	-0.954
Primary domain	116	-0.944	0.986	-0.976	-0.958	-0.953
Balanced scale grid	33	-0.982	0.985	-0.984	-0.989	-0.985
M allocation–mixture	30	-0.923	0.984	-0.976	-0.910	-0.896

For example, increasing the balanced M budget from $0.25C_0$ to $8C_0$ raises Core-v2 from 0.0716 to 0.1544 and HellaSwag zero-shot accuracy from 26.84% to 38.22%, while LAMBADA NLL falls from 4.0951 to 2.1498. Table 44 reports every task at the common $2C_0$ budget for all six balanced sizes. The individual tasks need not share the aggregate ordering.

Table 44: All 22 text-task scores at $2C_0$ for the six balanced sizes. Entries are percentages: MC accuracy or continuation EM as specified in Table 42.

Task	T	S	B	M	L	XL
HellaSwag (0-shot)	28.70	31.39	32.85	33.47	32.76	31.99
LAMBADA	30.99	36.29	38.39	40.29	37.69	34.76
ARC-Easy	42.68	46.63	48.70	49.66	50.55	47.69
ARC-Challenge	23.29	24.40	24.66	25.00	25.60	23.81
CommonsenseQA	28.91	35.46	39.39	24.08	21.62	22.77
PIQA	58.60	59.68	60.83	62.73	61.75	63.82
CoQA	15.70	17.64	20.64	21.37	19.80	16.17
BoolQ	54.07	45.99	41.31	56.51	59.94	53.33
HellaSwag (10-shot)	28.62	31.56	32.44	33.37	32.66	31.93
Language identification	24.75	25.19	24.73	25.58	25.85	25.34
LSAT-AR	20.00	22.61	23.48	20.00	23.48	20.87
SQuAD	14.75	19.56	25.24	25.82	24.38	18.20
CS algorithms	39.17	40.83	43.26	40.61	44.32	40.83
Dyck languages	8.50	5.30	8.60	8.00	11.30	10.50
Operators	20.95	24.29	24.76	22.38	19.52	23.33
Repeat-copy logic	6.25	3.12	0.00	0.00	3.12	3.12
QA Wikidata	20.37	36.03	38.94	38.67	33.58	30.61
Jeopardy	0.09	1.13	0.99	2.31	0.85	0.80
COPA	57.00	56.00	63.00	65.00	58.00	63.00
OpenBookQA	26.40	28.60	29.40	28.40	29.40	27.60
Winograd	59.34	57.14	59.71	59.71	60.81	58.24
WinoGrande	51.62	52.96	49.64	50.36	49.72	50.43

Mixture allocation changes downstream language performance. Figure 13 holds model size fixed and shows all M capacity allocations and speech shares at both compute references. Raising speech exposure toward the highest shares generally reduces downstream text scores and increases LAMBADA NLL; the aggregate and individual responses are not strictly monotonic at every adjacent pair. The external results thus complement the validation-loss trade-off rather than replacing it with a single capability score. The L balanced $2C_0$ mixture comparison similarly yields LAMBADA NLLs of 2.5633, 2.8412, and 4.1608 at approximately 8%, 30%, and 66% ASR target share, with zero-shot HellaSwag accuracies of 33.62%, 32.24%, and 28.24%.

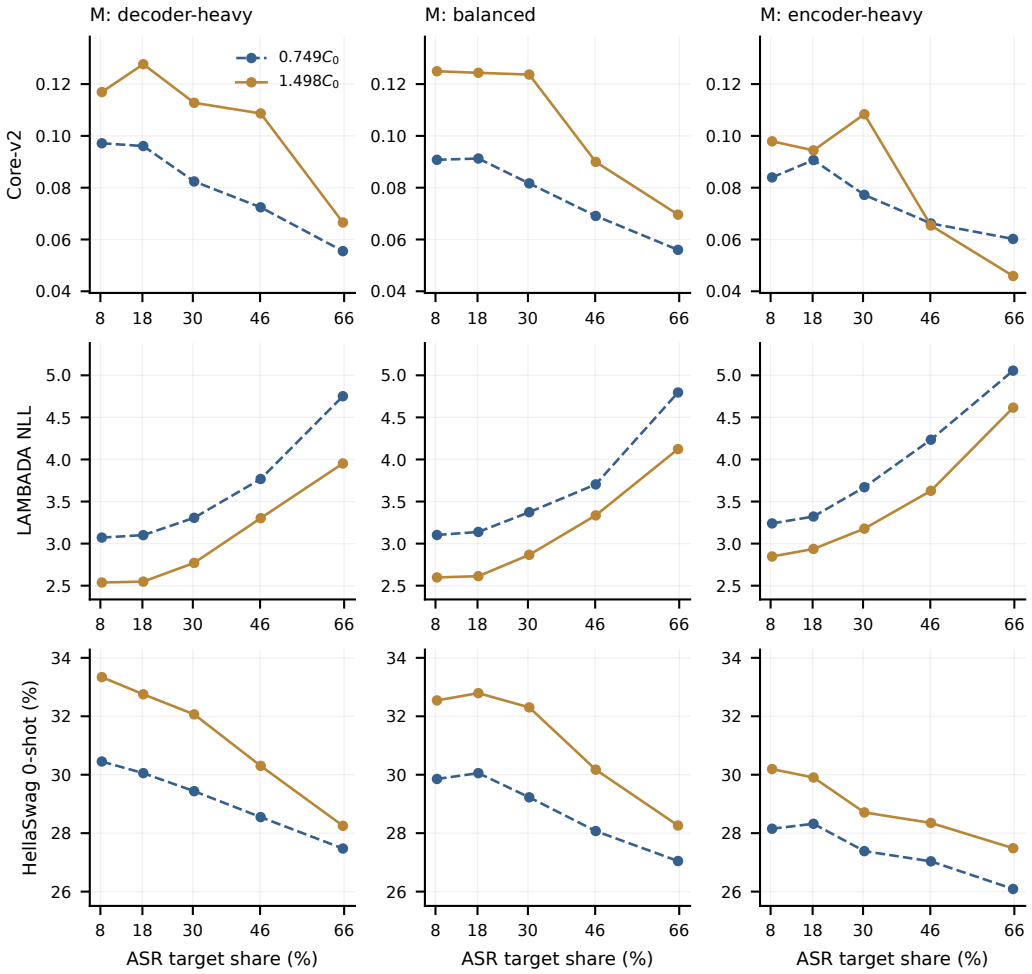


Figure 13: **The speech–language mixture trade-off extends beyond validation NLL.** All 30 M allocation–mixture cells are evaluated on Core-v2, LAMBADA, and zero-shot HellaSwag. Columns vary capacity allocation; each curve varies supervised ASR target share at the indicated compute reference. Axis ranges are shared across columns within each row.

Complete numerical results. Table 45 reports Core-v2, LAMBADA NLL and EM, and both HellaSwag settings for every evaluated configuration. IDs match Appendix S, where exact parameter sizes, target exposures, and other experiment coordinates are tabulated. The accompanying `data/text_external_all_tasks.csv` contains all $140 \times 22 = 3{,}080$ task results, including accuracy/EM, reference baseline, normalized Core contribution, and continuation NLL/token accuracy where applicable. `data/text_external_results.json` additionally supplies task definitions, configuration coordinates, and the exact plotting populations.

Table 45: Complete external text results. Model letters identify the width/size family; E/D specifies allocation. Core-v2 is a raw score; LAM NLL is in nats per continuation token; LAM EM and HS 0/10 are percentages.

ID	Model	E/D	C/C_0	Core-v2	LAM NLL	LAM EM	HS 0/10
N001	L	18/23	2.000	0.0778	4.1608	22.28	28.24/28.41
R001	B	14/18	0.389	0.0714	3.7148	25.17	27.95/27.89
R002	B	14/18	0.778	0.1108	3.1926	31.09	30.02/29.82
R003	B	14/18	1.556	0.1106	2.7809	35.88	31.61/31.33
R004	B	0/36	0.480	0.0766	3.5125	27.83	28.40/28.76
R005	B	0/36	0.961	0.1131	2.9862	32.84	30.77/30.98
R006	B	0/36	1.922	0.1244	2.5473	39.14	33.15/33.36

ID	Model	E/D	C/C_0	Core-v2	LAM NLL	LAM EM	HS 0/10
R007	B	26/3	0.315	0.0397	4.4664	18.84	26.65/26.11
R008	B	26/3	0.629	0.0539	4.2487	20.98	26.65/26.39
R009	B	26/3	1.259	0.0435	3.9222	23.33	27.15/26.97
R010	B	6/28	0.439	0.0795	3.5708	27.32	28.24/28.98
R011	B	6/28	0.877	0.1127	3.0524	32.49	30.75/30.52
R012	B	6/28	1.754	0.1190	2.6465	38.85	32.47/32.58
R013	M	16/21	0.250	0.0716	4.0951	21.23	26.84/27.29
R014	M	16/21	0.500	0.0826	3.4860	27.77	28.84/29.06
R015	M	16/21	1.000	0.1062	2.9766	33.28	31.12/31.17
R016	M	16/21	2.000	0.1306	2.5255	40.29	33.47/33.37
R017	M	16/21	4.000	0.1470	2.3438	42.32	35.89/35.73
R018	L	18/23	0.250	0.0459	4.4241	18.42	26.60/26.27
R019	L	18/23	0.500	0.0747	3.5929	26.33	27.98/28.26
R020	L	18/23	1.001	0.0960	3.1823	31.13	30.08/30.20
R021	L	18/23	2.002	0.1267	2.6486	37.69	32.76/32.66
R022	L	18/23	4.003	0.1546	2.2363	44.38	36.71/36.68
R023	B	14/18	0.250	0.0690	3.9724	23.11	27.11/27.51
R024	B	14/18	0.500	0.0915	3.3549	29.73	29.31/29.05
R025	B	14/18	1.000	0.1223	2.9079	34.27	30.96/30.78
R026	B	14/18	2.000	0.1210	2.6247	38.39	32.85/32.44
R027	B	14/18	4.000	0.1413	2.4784	40.81	34.12/33.89
R028	T	9/12	0.250	0.0582	3.6999	26.02	27.20/26.95
R029	T	9/12	0.500	0.0754	3.4347	29.26	27.83/28.15
R030	T	9/12	1.000	0.0889	3.2971	30.58	28.68/28.39
R031	T	9/12	2.000	0.0890	3.2309	30.99	28.70/28.62
R032	T	9/12	4.000	0.0976	3.2007	31.50	28.98/28.79
R034	B	26/3	0.372	0.0517	4.2837	20.84	26.62/26.39
R035	B	6/28	0.586	0.0989	3.2835	29.85	30.19/29.70
R036	787M	25/20	1.286	0.1273	2.7633	36.31	31.76/31.90
R037	787M	25/20	0.321	0.0760	3.6889	25.64	27.53/27.58
R038	787M	25/20	0.643	0.0965	3.1465	31.90	29.17/28.95
R039	S	20/6	0.500	0.0539	3.7893	24.96	26.84/27.09
R040	S	4/26	0.500	0.0931	3.2463	30.74	29.39/29.77
R041	S	12/16	1.000	0.1027	2.9988	33.90	30.49/30.04
R042	S	12/16	2.000	0.1058	2.8094	36.29	31.39/31.56
R043	S	12/16	0.250	0.0747	3.8663	24.10	27.62/27.72
R044	S	12/16	4.000	0.1133	2.6311	38.66	32.44/32.42
R045	S	12/16	0.500	0.0863	3.3343	30.51	28.46/29.07
R046	B	14/18	0.390	0.0945	3.4639	27.48	28.50/28.52
R047	B	14/18	0.386	0.0912	3.4591	26.64	28.28/28.28
R048	B	14/18	0.410	0.0385	5.1049	15.74	26.89/26.81
R049	B	14/18	0.393	0.0873	3.7080	24.88	27.79/28.27
R050	B	14/18	0.393	0.0682	4.2200	20.78	27.21/27.32
R051	B	14/18	0.779	0.1126	2.9132	32.93	30.28/30.44
R052	B	14/18	0.749	0.1097	3.2239	30.91	29.46/29.53
R053	B	14/18	0.820	0.0726	4.4318	20.10	27.26/27.31
R054	T	9/12	0.778	0.1050	3.4008	28.84	27.98/28.00
R055	T	9/12	0.802	0.0815	4.3248	21.87	27.48/27.38
R056	T	9/12	0.761	0.0880	3.1621	31.71	28.33/28.26
R057	787M	25/20	2.571	0.1548	2.4134	41.55	34.22/34.39
R058	M	16/21	0.507	0.0807	3.6281	25.62	28.23/28.32
R059	M	16/21	0.522	0.0559	4.9542	16.07	26.52/26.46
R060	M	16/21	0.495	0.0799	3.4818	26.06	28.92/28.81
R061	M	26/8	0.500	0.0724	3.7543	24.04	27.52/27.76
R062	M	5/35	0.500	0.0833	3.4807	28.06	28.86/29.01
R063	B	14/18	1.498	0.1088	2.7987	36.21	31.08/30.95
R064	B	14/18	1.640	0.0820	3.9705	25.71	29.06/29.02
R065	B	14/18	1.558	0.1227	2.5920	37.92	32.14/31.93
R066	B	26/3	0.856	0.0287	5.1816	14.88	26.12/25.79
R067	B	6/28	0.854	0.0690	4.3000	21.41	27.79/27.84
R068	427M	28/35	0.493	0.0894	3.2978	29.58	29.46/29.87
R069	489M	11/14	0.495	0.0763	3.4017	29.67	29.04/28.93
R070	489M	11/14	0.248	0.0736	3.9870	22.28	27.21/27.22
R071	M	16/21	0.749	0.0817	3.3736	28.51	29.23/29.22
R072	M	16/21	0.749	0.0908	3.1024	31.17	29.85/30.07
R073	M	16/21	0.749	0.0560	4.7960	17.62	27.05/27.31
R074	M	5/35	0.749	0.0824	3.3066	29.61	29.44/29.41
R075	M	5/35	0.749	0.0555	4.7519	18.34	27.47/27.43
R076	M	5/35	0.749	0.0972	3.0717	31.67	30.45/30.36
R077	M	26/8	0.749	0.0840	3.2399	28.82	28.15/28.26
R078	M	26/8	0.749	0.0772	3.6701	25.44	27.38/27.49
R079	B	14/18	0.749	0.1072	3.0027	32.99	29.96/30.11
R080	B	14/18	0.749	0.0751	3.7310	26.66	28.48/29.08
R081	489M	11/14	0.991	0.0917	2.9556	33.79	30.38/30.44
R082	427M	28/35	0.247	0.0754	3.8757	24.32	28.00/28.18
R083	M	26/8	0.749	0.0602	5.0555	16.24	26.09/26.16
R084	427M	28/35	0.987	0.1065	2.7572	36.85	31.57/31.83
R085	M	5/35	1.498	0.1169	2.5378	39.80	33.34/32.93
R086	M	16/21	1.498	0.1250	2.5980	37.84	32.54/32.38

ID	Model	E/D	C/C_0	Core-v2	LAM NLL	LAM EM	HS 0/10
R087	M	16/21	1.498	0.1237	2.8666	34.91	32.30/31.69
R088	M	5/35	1.498	0.1128	2.7708	35.80	32.07/32.30
R089	M	26/8	1.498	0.1084	3.1778	30.18	28.71/29.09
R090	M	16/21	1.498	0.0696	4.1235	23.27	28.26/28.39
R091	M	5/35	1.498	0.0666	3.9529	25.00	28.25/28.77
R092	M	16/21	0.749	0.0913	3.1387	30.99	30.05/30.17
R093	M	16/21	0.749	0.0691	3.7039	25.89	28.07/28.23
R094	M	26/8	1.498	0.0979	2.8471	34.08	30.19/30.07
R095	M	26/8	1.498	0.0459	4.6159	19.27	27.48/27.46
R096	M	5/35	0.749	0.0961	3.1008	31.61	30.05/30.14
R097	B	14/18	1.498	0.1064	2.6956	37.30	31.67/31.49
R098	M	26/8	0.749	0.0662	4.2346	21.09	27.04/27.19
R099	B	14/18	1.498	0.0997	3.2482	31.46	30.32/30.29
R100	M	16/21	1.498	0.1244	2.6129	37.76	32.79/32.29
R101	M	5/35	0.749	0.0724	3.7678	24.84	28.55/28.81
R102	M	16/21	1.498	0.0900	3.3358	29.56	30.17/30.31
R103	M	26/8	0.749	0.0907	3.3218	28.08	28.32/28.17
R104	XL	20/26	0.500	0.0625	4.1419	20.49	27.13/26.96
R105	XL	20/26	4.000	0.1546	2.3951	41.51	36.07/36.27
R106	XL	20/26	1.000	0.0832	3.5022	26.33	28.62/28.89
R107	XL	20/26	0.250	0.0407	4.9134	14.87	26.22/26.15
X001	L	6/38	0.500	0.0795	3.7983	24.35	27.84/28.22
X002	L	6/38	0.250	0.0456	4.5741	16.24	26.64/26.79
X003	M	5/35	1.498	0.1277	2.5481	39.20	32.75/32.67
X004	XL	20/26	2.000	0.1104	2.8529	34.76	31.99/31.93
X005	L	29/9	0.500	0.0677	3.7817	23.91	26.96/27.35
X006	L	29/9	0.250	0.0435	4.4911	17.56	26.31/26.43
X007	L	12/31	0.250	0.0427	4.4993	17.39	26.45/26.61
X008	M	5/35	1.498	0.1087	3.3027	31.34	30.30/30.63
X009	L	24/15	0.250	0.0485	4.3802	18.44	26.42/26.73
X010	M	26/8	1.498	0.0654	3.6279	26.00	28.35/28.22
X012	M	26/8	1.498	0.0944	2.9366	32.99	29.90/29.83
X013	L	6/38	1.000	0.1102	3.1805	31.22	30.16/30.56
X014	L	12/31	0.500	0.0748	3.7362	24.53	27.60/27.61
X015	L	12/31	1.000	0.1005	3.1255	31.75	30.00/30.09
X016	L	29/9	1.000	0.0839	3.4610	27.32	28.95/28.89
X017	L	24/15	1.000	0.0833	3.2124	30.60	29.64/29.95
X018	XL	20/26	8.000	0.1728	2.0698	47.80	40.64/40.82
X019	L	6/38	2.000	0.1543	2.6391	37.76	33.81/33.55
X020	L	12/31	2.000	0.1455	2.6551	37.63	33.60/33.38
X021	L	29/9	2.000	0.1113	2.9059	33.69	30.78/30.50
X022	L	24/15	2.000	0.1176	2.7034	36.93	32.18/32.46
X023	L	18/23	8.000	0.1798	2.0111	49.33	40.41/40.45
X024	L	6/38	2.000	0.1317	2.5050	39.47	33.60/33.48
X025	L	6/38	4.000	0.1586	2.1955	44.94	38.00/38.41
X026	L	12/31	4.000	0.1569	2.2093	45.16	38.10/37.97
X027	L	6/38	2.000	0.1185	2.8167	35.38	32.19/32.65
X028	L	29/9	4.000	0.1245	2.6037	37.94	32.56/32.49
X029	L	24/15	4.000	0.1432	2.3636	42.79	35.42/35.21
X030	L	6/38	2.000	0.0868	4.2167	22.49	28.73/29.12
Z001	M	16/21	8.000	0.1544	2.1498	47.22	38.22/38.06
Z002	L	18/23	2.000	0.1170	2.5633	38.15	33.62/33.35
Z003	L	18/23	2.000	0.1193	2.8412	34.72	32.24/32.20
Z004	L	24/15	0.500	0.0777	3.6745	25.25	27.72/27.93

S EXPERIMENTAL COVERAGE AND ENDPOINT CATALOG

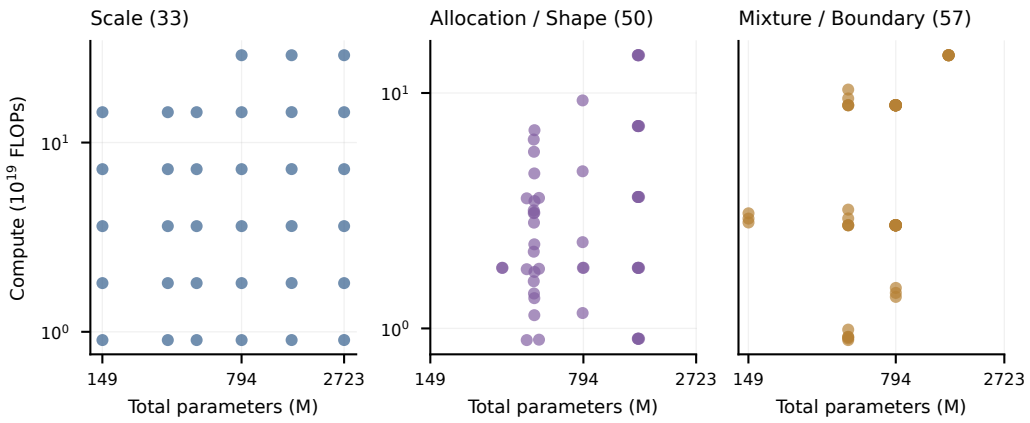


Figure 14: Completed configurations by experimental family. The 141-configuration catalog contains 140 cost-audited endpoints and one text-only reference without a reported cost. Coordinates are not jittered; overlapping points can represent different allocations or mixtures.

The catalog in `data/experiment_catalog.json` provides all 141 representative configurations, including the 116 primary regression observations and the complementary budget, architecture, and mixture experiments. Pretrained-initialization and mechanism comparisons are tabulated separately in Appendix H.

Table 46: Balanced scale–budget endpoints. Sizes use total parameters; targets are billions. Each horizon is trained independently.

Model	C/C_0	Updates	$C_{\mathrm{op}}/10^{19}$	Text targets	ASR targets	ASR NLL	Text NLL
T	0.25	88,293	0.904	5.782	1.184	0.2979	2.6442
T	0.50	176,587	1.809	11.564	2.368	0.2644	2.5714
T	1.00	353,173	3.617	23.128	4.737	0.2371	2.5251
T	2.00	706,346	7.234	46.256	9.481	0.2294	2.4847
T	4.00	1,412,692	14.468	92.511	18.963	0.2081	2.4599
S	0.25	42,835	0.904	2.805	0.574	0.2861	2.6461
S	0.50	85,670	1.809	5.610	1.149	0.2285	2.5037
S	1.00	171,341	3.617	11.220	2.297	0.1984	2.3875
S	2.00	342,681	7.234	22.441	4.596	0.1789	2.3288
S	4.00	685,362	14.468	44.882	9.200	0.1627	2.2839
B	0.25	31,250	0.904	2.046	0.419	0.3222	2.6124
B	0.50	62,500	1.809	4.093	0.838	0.2389	2.4884
B	1.00	125,000	3.617	8.186	1.676	0.1941	2.3586
B	2.00	250,000	7.235	16.372	3.352	0.1722	2.2956
B	4.00	500,000	14.469	32.743	6.709	0.1564	2.2383
M	0.25	18,777	0.904	1.230	0.251	0.3940	2.6542
M	0.50	37,555	1.809	2.459	0.503	0.2651	2.4966
M	1.00	75,109	3.617	4.918	1.007	0.2081	2.3547
M	2.00	150,218	7.234	9.837	2.014	0.1724	2.2542
M	4.00	300,436	14.469	19.674	4.028	0.1500	2.1571
M	8.00	600,874	28.938	39.349	8.065	0.1397	2.1244
L	0.25	10,683	0.905	0.700	0.143	0.8197	2.7901
L	0.50	21,365	1.810	1.399	0.286	0.3774	2.5350
L	1.00	42,730	3.621	2.798	0.573	0.2834	2.4241
L	2.00	85,460	7.241	5.596	1.146	0.1875	2.2834
L	4.00	170,920	14.482	11.193	2.291	0.1549	2.1528
L	8.00	341,544	28.938	22.366	4.580	0.1302	2.0632
XL	0.25	5,811	0.905	0.381	0.078	2.0532	2.9486
XL	0.50	11,621	1.809	0.761	0.156	0.5652	2.6919
XL	1.00	23,243	3.618	1.522	0.311	0.3412	2.4755
XL	2.00	46,485	7.235	3.044	0.623	0.2330	2.3333
XL	4.00	92,970	14.469	6.088	1.246	0.1901	2.1787
XL	8.00	185,941	28.938	12.176	2.493	0.1372	2.0647

Table 47: Completed configuration catalog. N denotes total parameters in millions; x_{asr} is the ASR supervised-target share. Blank cost denotes an unavailable measurement.

ID	Family	d	E/D	N	$C_{\mathrm{op}}/10^{19}$	x_{asr}	ASR NLL	Text NLL
N001	Mixture	1536	18/23	1449.0	7.234	0.657	0.2327	2.6548
R001	Allocation	896	14/18	462.6	1.407	0.291	0.2329	2.5531
R002	Allocation	896	14/18	462.6	2.814	0.291	0.1904	2.4437
R003	Allocation	896	14/18	462.6	5.628	0.291	0.1640	2.3210
R004	Allocation	896	0/36	465.2	1.738	0.291	0.2667	2.5042
R005	Allocation	896	0/36	465.2	3.476	0.291	0.2141	2.3818
R006	Allocation	896	0/36	465.2	6.952	0.291	0.1809	2.2599
R007	Allocation	896	26/3	464.3	1.138	0.291	0.2523	2.8881
R008	Allocation	896	26/3	464.3	2.277	0.291	0.2059	2.8131
R009	Allocation	896	26/3	464.3	4.553	0.291	0.1733	2.7225
R010	Allocation	896	6/28	461.5	1.586	0.291	0.2403	2.5077
R011	Allocation	896	6/28	461.5	3.172	0.291	0.1986	2.3950
R012	Allocation	896	6/28	461.5	6.345	0.291	0.1690	2.2739
R013	Scale	1152	16/21	794.4	0.904	0.170	0.3940	2.6542
R014	Scale	1152	16/21	794.4	1.809	0.170	0.2651	2.4966
R015	Scale	1152	16/21	794.4	3.617	0.170	0.2081	2.3547
R016	Scale	1152	16/21	794.4	7.234	0.170	0.1724	2.2542
R017	Scale	1152	16/21	794.4	14.469	0.170	0.1500	2.1571
R018	Scale	1536	18/23	1449.0	0.905	0.170	0.8197	2.7901
R019	Scale	1536	18/23	1449.0	1.810	0.170	0.3774	2.5350
R020	Scale	1536	18/23	1449.0	3.621	0.170	0.2834	2.4241
R021	Scale	1536	18/23	1449.0	7.241	0.170	0.1875	2.2834
R022	Scale	1536	18/23	1449.0	14.482	0.170	0.1549	2.1528
R023	Scale	896	14/18	462.6	0.904	0.170	0.3222	2.6124
R024	Scale	896	14/18	462.6	1.809	0.170	0.2389	2.4884
R025	Scale	896	14/18	462.6	3.617	0.170	0.1941	2.3586
R026	Scale	896	14/18	462.6	7.235	0.170	0.1722	2.2956
R027	Scale	896	14/18	462.6	14.469	0.170	0.1564	2.2383
R028	Scale	512	9/12	149.2	0.904	0.170	0.2979	2.6442
R029	Scale	512	9/12	149.2	1.809	0.170	0.2644	2.5714
R030	Scale	512	9/12	149.2	3.617	0.170	0.2371	2.5251
R031	Scale	512	9/12	149.2	7.234	0.170	0.2294	2.4847
R032	Scale	512	9/12	149.2	14.468	0.170	0.2081	2.4599
R033	Boundary	896	0/18	299.3	–	0.000	–	2.3322
R034	Allocation	896	26/3	464.3	1.345	0.170	0.2703	2.8477
R035	Allocation	896	6/28	461.5	2.118	0.170	0.2479	2.4472
R036	Shape	1024	25/20	787.4	4.650	0.291	0.1716	2.3256
R037	Shape	1024	25/20	787.4	1.163	0.290	0.3518	2.5777
R038	Shape	1024	25/20	787.4	2.325	0.291	0.2204	2.4315
R039	Allocation	768	20/6	328.0	1.809	0.170	0.2306	2.6304
R040	Allocation	768	4/26	326.4	1.809	0.170	0.2561	2.4645
R041	Scale	768	12/16	327.2	3.617	0.170	0.1984	2.3875
R042	Scale	768	12/16	327.2	7.234	0.170	0.1789	2.3288
R043	Scale	768	12/16	327.2	0.904	0.170	0.2861	2.6461
R044	Scale	768	12/16	327.2	14.468	0.170	0.1627	2.2839
R045	Scale	768	12/16	327.2	1.809	0.170	0.2285	2.5037
R046	Mixture	896	14/18	462.6	1.409	0.084	0.2970	2.4753
R047	Mixture	896	14/18	462.6	1.396	0.182	0.2600	2.5228
R048	Mixture	896	14/18	462.6	1.483	0.657	0.2346	2.9589
R049	Mixture	896	14/18	462.6	1.422	0.303	0.2258	2.5475
R050	Mixture	896	14/18	462.6	1.422	0.461	0.2301	2.7156
R051	Mixture	896	14/18	462.6	2.818	0.084	0.2255	2.3357
R052	Mixture	896	14/18	462.6	2.709	0.303	0.1891	2.4759
R053	Mixture	896	14/18	462.6	2.965	0.657	0.1983	2.8249
R054	Mixture	512	9/12	149.2	2.816	0.303	0.2120	2.5282
R055	Mixture	512	9/12	149.2	2.903	0.658	0.2297	2.8547
R056	Mixture	512	9/12	149.2	2.753	0.084	0.2530	2.4623
R057	Shape	1024	25/20	787.4	9.301	0.291	0.1483	2.2065
R058	Mixture	1152	16/21	794.4	1.833	0.303	0.2501	2.5337
R059	Mixture	1152	16/21	794.4	1.887	0.657	0.2755	2.9413
R060	Mixture	1152	16/21	794.4	1.791	0.083	0.3331	2.4563
R061	Allocation	1152	26/8	789.5	1.809	0.170	0.2649	2.5930
R062	Allocation	1152	5/35	795.3	1.809	0.170	0.2855	2.4718
R063	Mixture	896	14/18	462.6	5.419	0.303	0.1675	2.3446
R064	Mixture	896	14/18	462.6	5.934	0.657	0.1749	2.6646
R065	Mixture	896	14/18	462.6	5.636	0.084	0.1967	2.2450
R066	Allocation	896	26/3	464.3	3.098	0.657	0.1976	3.1091
R067	Allocation	896	6/28	461.5	3.089	0.657	0.2148	2.7849
R068	Shape	640	28/35	427.4	1.785	0.170	0.2377	2.4742
R069	Shape	1024	11/14	488.8	1.792	0.170	0.2431	2.5275
R070	Shape	1024	11/14	488.8	0.896	0.170	0.3228	2.6239
R071	Mixture	1152	16/21	794.4	2.708	0.303	0.2065	2.4566
R072	Mixture	1152	16/21	794.4	2.709	0.084	0.2685	2.3609
R073	Mixture	1152	16/21	794.4	2.709	0.657	0.2314	2.9273
R074	Mixture	1152	5/35	795.3	2.709	0.303	0.2221	2.4339

ID	Family	d	E/D	N	$C_{\mathrm{op}}/10^{19}$	x_{asr}	ASR NLL	Text NLL
R075	Mixture	1152	5/35	795.3	2.709	0.657	0.2337	2.8383
R076	Mixture	1152	5/35	795.3	2.709	0.083	0.2879	2.3607
R077	Mixture	1152	26/8	789.5	2.709	0.084	0.2398	2.4419
R078	Mixture	1152	26/8	789.5	2.709	0.303	0.2021	2.5894
R079	Mixture	896	14/18	462.6	2.709	0.182	0.1990	2.3860
R080	Mixture	896	14/18	462.6	2.709	0.461	0.1928	2.5864
R081	Shape	1024	11/14	488.8	3.584	0.170	0.2000	2.3908
R082	Shape	640	28/35	427.4	0.892	0.170	0.3215	2.6020
R083	Mixture	1152	26/8	789.5	2.709	0.657	0.2197	3.0494
R084	Shape	640	28/35	427.4	3.569	0.170	0.1931	2.3440
R085	Mixture	1152	5/35	795.3	5.419	0.084	0.2222	2.2111
R086	Mixture	1152	16/21	794.4	5.419	0.084	0.2047	2.2267
R087	Mixture	1152	16/21	794.4	5.419	0.303	0.1705	2.3434
R088	Mixture	1152	5/35	795.3	5.420	0.303	0.1880	2.3385
R089	Mixture	1152	26/8	789.5	5.419	0.303	0.1671	2.4681
R090	Mixture	1152	16/21	794.4	5.419	0.657	0.1822	2.7233
R091	Mixture	1152	5/35	795.3	5.419	0.657	0.1987	2.6887
R092	Mixture	1152	16/21	794.4	2.710	0.182	0.2196	2.3881
R093	Mixture	1152	16/21	794.4	2.709	0.461	0.2177	2.6359
R094	Mixture	1152	26/8	789.5	5.419	0.084	0.1896	2.3335
R095	Mixture	1152	26/8	789.5	5.420	0.657	0.1735	2.7961
R096	Mixture	1152	5/35	795.3	2.709	0.182	0.2378	2.3928
R097	Mixture	896	14/18	462.6	5.419	0.182	0.1738	2.2783
R098	Mixture	1152	26/8	789.5	2.709	0.461	0.2137	2.7021
R099	Mixture	896	14/18	462.6	5.420	0.461	0.1654	2.4958
R100	Mixture	1152	16/21	794.4	5.418	0.182	0.1765	2.2483
R101	Mixture	1152	5/35	795.3	2.709	0.461	0.2225	2.5968
R102	Mixture	1152	16/21	794.4	5.420	0.461	0.1710	2.4992
R103	Mixture	1152	26/8	789.5	2.709	0.182	0.2079	2.4965
R104	Scale	2048	20/26	2723.5	1.809	0.170	0.5652	2.6919
R105	Scale	2048	20/26	2723.5	14.469	0.170	0.1901	2.1787
R106	Scale	2048	20/26	2723.5	3.618	0.170	0.3412	2.4755
R107	Scale	2048	20/26	2723.5	0.905	0.169	2.0532	2.9486
X001	Allocation	1536	6/38	1443.2	1.808	0.170	0.3764	2.5519
X002	Allocation	1536	6/38	1443.2	0.904	0.170	0.7843	2.7522
X003	Mixture	1152	5/35	795.3	5.418	0.182	0.1952	2.2506
X004	Scale	2048	20/26	2723.5	7.235	0.170	0.2330	2.3333
X005	Allocation	1536	29/9	1447.9	1.809	0.170	0.3961	2.6172
X006	Allocation	1536	29/9	1447.9	0.904	0.170	0.8107	2.7659
X007	Allocation	1536	12/31	1459.3	0.904	0.170	1.2180	2.7505
X008	Mixture	1152	5/35	795.3	5.420	0.461	0.1869	2.4450
X009	Allocation	1536	24/15	1438.8	0.904	0.170	1.0330	2.7342
X010	Mixture	1152	26/8	789.5	5.420	0.461	0.1697	2.6186
X012	Mixture	1152	26/8	789.5	5.418	0.182	0.1723	2.3812
X013	Allocation	1536	6/38	1443.2	3.617	0.170	0.2646	2.3891
X014	Allocation	1536	12/31	1459.3	1.809	0.170	0.3821	2.5560
X015	Allocation	1536	12/31	1459.3	3.617	0.170	0.2530	2.3956
X016	Allocation	1536	29/9	1447.9	3.617	0.170	0.2850	2.5199
X017	Allocation	1536	24/15	1438.8	3.618	0.170	0.2535	2.4226
X018	Scale	2048	20/26	2723.5	28.938	0.170	0.1372	2.0647
X019	Allocation	1536	6/38	1443.2	7.235	0.170	0.2069	2.2623
X020	Allocation	1536	12/31	1459.3	7.235	0.170	0.1936	2.2463
X021	Allocation	1536	29/9	1447.9	7.234	0.170	0.1909	2.3496
X022	Allocation	1536	24/15	1438.8	7.235	0.170	0.1868	2.3001
X023	Scale	1536	18/23	1449.0	28.938	0.170	0.1302	2.0632
X024	Mixture	1536	6/38	1443.2	7.234	0.084	0.2291	2.2023
X025	Allocation	1536	6/38	1443.2	14.469	0.170	0.1688	2.1385
X026	Allocation	1536	12/31	1459.3	14.469	0.170	0.1576	2.1443
X027	Mixture	1536	6/38	1443.2	7.236	0.303	0.1843	2.3187
X028	Allocation	1536	29/9	1447.9	14.469	0.170	0.1589	2.2686
X029	Allocation	1536	24/15	1438.8	14.469	0.170	0.1504	2.1851
X030	Mixture	1536	6/38	1443.2	7.235	0.657	0.2285	2.6503
Z001	Scale	1152	16/21	794.4	28.938	0.170	0.1397	2.1244
Z002	Mixture	1536	18/23	1449.0	7.234	0.084	0.2107	2.1926
Z003	Mixture	1536	18/23	1449.0	7.235	0.303	0.1748	2.3136
Z004	Allocation	1536	24/15	1438.8	1.809	0.170	0.3710	2.5697