

Domain-Based Routing in Mixture-of-Experts Models for BabyLMs

Jeremias Bohn and Sebastian Feiertag and Nicolas Raube and
Emil Schwenger and Georg Groh

Technical University of Munich
School of Computation, Information and Technology
Germany
{firstname.lastname}@tum.de

Abstract

Recent language model improvements were mostly achieved by increasing model scale, compute, and in particular training data size. This differs significantly from human language acquisition, which relies on significantly fewer words. The BabyLM challenge tackles this issue by encouraging approaches inspired by cognitive insights to reduce data dependency, thus lowering the barrier for NLP researchers to contribute to their field. We propose a Mixture-of-Experts approach with guided domain-based routing to learn subnetworks specialized on human-interpretable topics, thus making experts focus on domain knowledge and style rather than specific token features. Additionally, we apply a CEFR-based curriculum learning to mimic human language learning. Our model underperforms compared to the 7 times larger baseline model, but scores best on the BabyLM strict-small leaderboard among models with less than 25M parameters. Our code is publicly available at <https://github.com/SmallLM-DBR-MoE-Submission-2026/Baby-DBR>, and our model can be found at <https://huggingface.co/elschr/BabyLM2026-DBR>.

1 Introduction

Recent progress in language modeling has, to a large degree, been driven by scale. Scaling laws show that model performance improves predictably as parameter count, dataset size, and compute budget grow (Kaplan et al., 2020; Hoffmann et al., 2022). As a result, state-of-the-art language models (LMs) are trained on trillions of tokens using compute budgets that reach tens of millions of dollars per training run (Grattafiori et al., 2024; Cotter et al., 2025). This dependence on massive resources stands in stark contrast to human language acquisition: children are exposed to roughly 100 million words by age 13, yet acquire robust

linguistic competence. It also places meaningful LLM pretraining research out of reach for most academics.

The BabyLM Challenge (Choshen et al., 2026) was created to address both issues. By restricting the pretraining data to 10 million words (for the strict-small track), it turns language modeling into a data-efficiency problem in which architectural and training innovations, rather than raw scale, determine success. This makes the setting both cognitively motivated and practically accessible to low-resource research environments.

In this work, we explore whether explicit, human-interpretable structure can improve data efficiency in this setting. We propose a Mixture-of-Experts (MoE) architecture in which expert routing is not learned purely implicitly, as in standard MoE models (Shazeer et al., 2017), but is instead guided by external topic labels: we partition our training corpus into eleven manually curated topic domains and use these gold labels as a supervision signal for the router during a dedicated specialization phase.

2 Related Work

Scaling Laws Kaplan et al. (2020) established that language model loss follows power laws in model size, dataset size, and compute, suggesting that training larger models on more data results in predictable gains. Hoffmann et al. (2022) refined this picture with the Chinchilla scaling laws, showing that many models of its time were undertrained relative to their size. For compute optimal training they show model size and training tokens should be scaled in equal proportion, with roughly 20 tokens per parameter. Under the BabyLM constraint of 10M words, this relationship is inverted: Data, not compute, is the binding constraint, which motivates architectural and algorithmic approaches to data efficiency rather than parameter scaling.

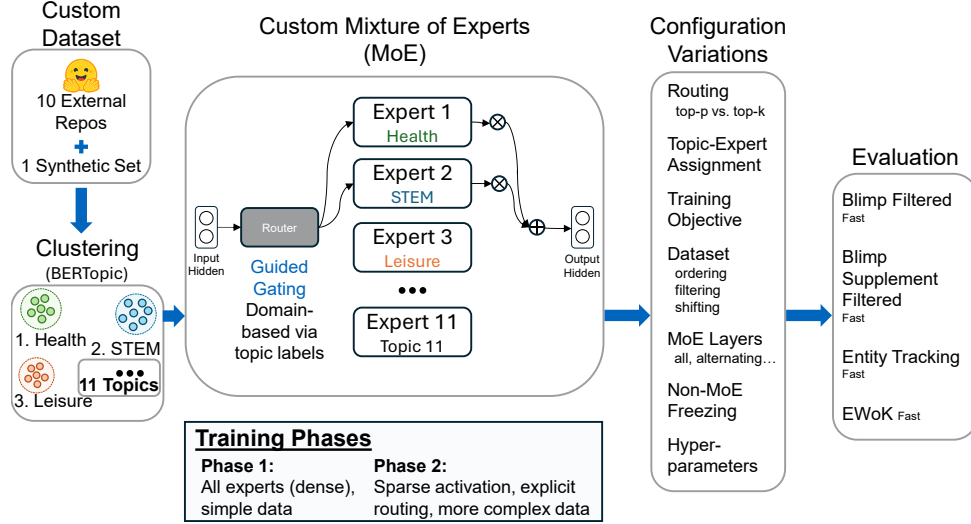


Figure 1: Overview of our core approach.

Mixture-of-Experts Mixture-of-Experts (MoEs) were introduced by [Jacobs et al. \(1991\)](#) as an alternative form of ensemble learning, allowing components of a full model to specialize on particular sub-tasks without deteriorating the overall model’s performance. The approach gained significant traction through the work of [Shazeer et al. \(2017\)](#), who introduced sparse MoEs: a learned router dispatches each token to a small subset of expert feed-forward networks, decoupling model capabilities from inference compute. This design underlies many recent large-scale models ([Meta AI, 2024](#); [Team et al., 2026](#); [Dai et al., 2024](#)). The shared-expert design we build on, in which one expert is always active to capture common knowledge while routed experts specialize, was popularized by DeepSeekMoE ([Dai et al., 2024](#)). However, routing in these models is learned implicitly, and analyses have found that experts tend to specialize along shallow token-level or syntactic patterns rather than human-interpretable domains ([Jiang et al., 2024](#)). Our work differs in that as we impose an explicit, semantically meaningful routing signal derived from topic labels.

MoE in BabyLM 2025 [Mehta et al. \(2025\)](#) find that while MoE alone already increases performance, it does so even more in combination with Multi-head Latent Attention (MLA) and Rotary Position Embeddings (RoPE). [Tapaninaho \(2025\)](#) experiments with shrinking the hidden dimension, then applying MoE-style routing to route tokens to multiple transformer blocks at the same time, and growing the hidden dimension afterward. This approach outperformed all strict-small baseline mod-

els. Overall, architectural improvements are recommended over parameter scaling in low-resource language modeling ([Mehta et al., 2025](#); [Charpentier et al., 2025](#)).

Domain-based Routing DEMix layers ([Gururangan et al., 2021](#)) condition expert selection in the feed-forward sublayer directly on document-level domain labels, training one expert per domain while sharing attention parameters across domains. Branch-Train-Merge ([Li et al., 2022](#)) instead trains separate expert LMs per domain and combines them at inference. Closest to our approach is DES-MoE ([Li et al., 2025](#)), which uses explicit domain metadata during the finetuning stage of an existing pretrained model, restricting updates to domain-specific experts via a multi-stage freezing schedule to counteract catastrophic forgetting. We differ by investigating explicit topic supervision during the pretraining phase.

3 Methods

In standard MoE models, routing is learned implicitly under load-balancing ([Shazeer et al., 2017](#)) constraints, and experts tend to specialize along shallow token-level or syntactic lines rather than human-interpretable topics. In contrast, we partition the training corpus into eleven semi-automatically defined topics prior to training and use these topic labels as an explicit routing signal, encouraging experts to develop semantically meaningful, topic-aligned specialization.

Inspired by natural learning processes, we further divide our training into two phases: During the

base phase, we use simple natural language and train our model densely, i.e., we activate all experts simultaneously, to accustom the model to the data. Afterward, in the *expert phase*, we use sparse routing to activate only specific experts corresponding to specific topics. A visualization of our approach can be found in Figure 1.

3.1 Custom Dataset and Clustering

Dataset Composition Since initial experiments showed that the challenge’s own dataset was hard to separate by topic modeling, we composed our own instead to better match our research question, partially reusing subsets of the BabyLM strict-small data. Apart from that, the remaining data comes from public sources of fixed domains, such as mathematics and news, with a significant proportion taken from the UniversalCEFR dataset (Imperial et al., 2025), which allows for curriculum learning based on the language difficulty, determined by the CEFR level. Further, datasets comprising simpler language, such as Daily Dialog (Li et al., 2017) and the Everyday Language Corpus (MultivexAI, 2026), are added.

We then filter the collected datasets to exclude long sequences (> 2048 characters) to ensure diversity while staying within the challenge’s word limit, which accumulates to a total of 9,653,393 words, which we partition into *base* (12.5%) and *expert* (87.5%) subsets. A comprehensive list and distribution of words can be found in Table 11.

Moreover, since the *base phase* requires simple speech and the corresponding number of dataset entries is relatively low, we synthesized additional data by sampling from our *base* subset and generating variations using DeepSeekV4 Flash. The exact prompt is given in Appendix B. In 70% of generations, the prompt was extended with a topic word to aim for unique examples, while a static prompt was used in the remaining 30%. In total, the generated data comprises 1,727,276 words, from which we only sample to fill up to a maximum of 10M words. As described in Section 4.6, additional instances from this subset are used for the *base phase* after additional filtering of the *expert phase* subset.

Topic Modeling To find topic groupings for each of the experts, we cluster dimensionality reduced embeddings of our examples using BERTopic (Grootendorst, 2022). This approach revealed 198 niche topics, with outliers being clustered into a single one, which we then combined

into 11 more general topics. The clustering was conducted by first prompting an LLM (Gemini 3.1 Pro) based on 10 BERTopic keywords for each topic, and manually reviewing and adjusting afterward; as part of this manual reviewing process, 8 specific topics were relocated. The final general topic descriptions are as follows:

- Math, Physics, Earth Science, Chemistry, Space
- Technology, Business, and Economy
- Health, Medicine, and Anatomy
- Nature, Zoology, and Botany
- History, Geography, and Law
- Education, Language, and Arts
- Entertainment, Sports, and Gaming
- Everyday Life, Home, and Transport
- Interpersonal Communication & Society
- Fantasy, Adventure, & Historical Fiction
- Contemporary Fiction & Character Dialogue

3.2 Model Architecture & Training Procedure

We ground our model on a modern Transformer basis, building upon the Llama 3 (Grattafiori et al., 2024) architecture implemented as a HuggingFace transformers (Wolf et al., 2020) module. We keep the original architecture and focus on changes to the feed forward network (FFN): the FFN in each layer is replaced by an MoE module comprising 11 experts, with each expert designed as a single SwiGLU module. A linear router maps each token-level hidden state to a softmax distribution over the experts, from which tokens are dispatched using either top- k or top- p selection. Optionally, a shared expert is applied to every token.

As mentioned before, the training process is then divided into two phases, each running for 10 epochs on their corresponding subset: Phase 1 (*base phase*) establishes general language competence with all experts active. It is followed by phase 2 (*expert phase*) in which the topic-supervised routing signal is applied to encourage interpretable specialization. All models use a BPE tokenizer trained from scratch on our corpus.

The hyperparameters that were kept consistent across experiments are listed in Appendix D. Additional hyperparameters are described in the corresponding sections below.

4 Experiments and Results

While our main architecture and dataset are fixed, we conduct an extensive ablation study on different factors influencing the final model’s performance,

including the routing mechanism, MoE layer locations, the gold label topic to model expert assignment strategy, dataset shifting, training data ordering and filtering, non-MoE freezing and changes to the training objective and usage of a (weighted) shared expert.

Throughout all experiments, we evaluate on the core tasks of the BabyLM challenge, i.e., BLiMP (Warstadt et al., 2020) and BLiMP Supplement (Warstadt et al., 2023), as well as Entity Tracking (Kim and Schuster, 2023) and EWoK (Ivanova et al., 2025).

Full experiment results for all ablation experiments are shown in Appendix A.

4.1 Catastrophic Forgetting

One of the most significant challenges we faced is catastrophic forgetting (McCloskey and Cohen, 1989) during the second phase, the expert specialization training. Our experiments in Appendix A show that for the *expert phase* models, the evaluation average tends to decline in comparison to their respective *base phase* evaluation results. This is primarily driven by decreases in BLiMP Supplement and Entity Tracking scores, while BLiMP scores generally rise. The basic model trained on simple speech tends to perform better on these tasks. This is also apparent in the score-by-epoch graph of our final submission model, as shown in Figure 3. The score for BLiMP Supplement suddenly decreases after the switch to the second training phase and fails to recover.

In an effort to address this problem, we trained multiple models that varied in terms of expert selection and MoE location. Furthermore, for the experiments in Table 5, at the start of the second (expert specialization) training phase, all non-MoE weights, except for the language model head, were frozen. This change aimed at improving entity tracking and BLiMP supplement scores by reducing the number of trainable parameters to expert internals. In addition, all models in this ablation had their phase one training dataset extended by a 30% sample from the expert dataset. We considered this necessary since we cannot expect the model to handle grammar and syntax of expert data it has never seen. The evaluation results are shown in Table 5.

In general, the changes did not help address the decline in average evaluation results. Instead, we further observed now that BLiMP scores also tend to decrease in the *expert phase*. Hence, the combi-

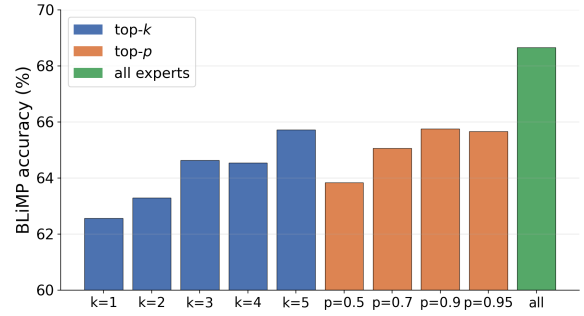


Figure 2: BLiMP fast zero-shot accuracy by routing configuration.

nation of shifting an expert dataset sample to the basic dataset and restricting training to experts only in the latter phase seemed to hurt understanding of grammar and syntax.

We did not observe any consistent improvements for this problem over the course of our experiments. We assume the problem is rooted in our training data, which may shift focus from basic understanding toward expert knowledge. This gain in knowledge is not properly reflected in the tasks we evaluated on, but may be beneficial in general autoregressive language modeling nonetheless.

4.2 MoE Adjustments

Given the specialization of our experts, the output of each of them may not be necessary for every task. Thus, we experiment with modifying the amount of active experts in a sparse MoE scenario, applying both Top- k (Shazeer et al., 2017) and Top- p (Holtzman et al., 2020) routing. Top- k routing sends each token to the k experts with the highest router scores, whereas Top- p routing sends each token to the smallest set of highest-scoring experts whose cumulative routing probability reaches the threshold p . This allows the number of experts to vary across tokens depending on router certainty.

As can be seen in Figure 2, we observe a positive correlation between the number of active experts (i.e., a higher k or p) and the accuracy on the BLiMP Benchmark. This correlation holds even up to the activation of all experts (all, so equivalent to $k = 11$ or $p = 1$), indicating that the sparse activation, while being computationally cheaper, does not boost accuracy in this separated ablation experiment. However, this may be a finding specific to the BLiMP dataset, since training a model with all experts active scored worse on the full evaluation submitted to the leaderboard (cf. Table 4). We therefore opted for $p = 0.7$ for our final sub-

mission, since this value showed a significant improvement over $p = 0.5$, while such an increment is not as prevalent after increasing p further.

Additionally to choosing experts dynamically, we further conducted experiments in which we added a shared expert (Dai et al., 2024) that is always active, allowing for a more consistent gradient flow and diverting basic knowledge away from our topic-primed experts. The intention is that a shared expert learns common basic language functions, such as grammar or basic words, that are needed for most inference tasks. Additionally, we experiment with introducing a dynamically calculated weight to scale its output. This weight is calculated based on the concatenated input vector and routing probabilities for the input. This vector is passed through a linear layer and a sigmoid function, which outputs one normalized score per token. This score is multiplied with the output of the shared expert.

Table 1 shows the results of these experiments. Both the regular shared expert and our weighted variant increase the average score to almost the same level compared to omitting the shared expert. The weighted shared expert achieves the highest result in BLiMP. The regular shared expert has the highest score for the base phase model. Additionally, using the top- k ($k = 1$) routing method instead of top- p ($p = 0.7$) leads to a better score in the base phase for the regular shared expert, but a worse score in the specialized phase. It is notable that Entity Tracking particularly improves in the *base phase* trained model. Given the mostly consistent improvements over all benchmarks after the expert phase, we decided to incorporate a weighted shared expert into our final submission model.

Furthermore, we considered replacing the Llama FFN with our custom MoE only in specific layers, e.g., by placing the MoE only in the top layers, alternating between MoE and regular SwiGLU layers, or replacing it in all layers. Table 6 shows the respective evaluation results. Not replacing all standard FFNs proved to be a solid choice for most tasks, with Entity Tracking once again profiting significantly after the *base phase*. In terms of average evaluation results, the alternating options seem to perform best across both evaluation points, and therefore being our choice for our final submission.

4.3 Router Teacher Forcing

While routing purely according to our gold labels in the training phase (to which we refer as *router*

teacher forcing) guarantees no contamination of experts with different topics, it also imposes a significant constraint on the flexibility of the optimization process and may create a mismatch between training and inference, as the model then has to rely on its own routing scores instead. When applying router teacher forcing, the routing probabilities are only used to calculate the router loss, the experts are activated according to our gold labels. When deactivated, routing probabilities are used both to calculate the routing loss and to activate experts, allowing for a *soft topic guidance*, i.e., domain routing is only indirectly enforced by weight optimization. Related to routing is the auxiliary load balancing loss. This loss tries to activate experts more uniformly in the number of tokens that are routed to each expert and how uniformly distributed the activation probabilities are.

Our experiments show that the *expert phase* model performs better if trained without router teacher forcing (cf. Table 7), and with an auxiliary loss (`p_0.7_no_teacher_with_aux-expert`). No shared expert was used for this experiment, thus results are to be compared with `p_0.7_no_shared-expert` in Table 1.

While a clear separation between experts might be desirable for interpretability, the negative effects on all benchmarks tested served as a clear indicator to remove router teacher forcing from our final submission.

4.4 Expert Assignment

Since our dataset has eleven predefined topics and the *expert phase* training requires mapping from gold-label topics to model experts, we need a sophisticated assignment algorithm. The three evaluated options are loss-based assignment, gradient-based assignment, and fixed assignment.

To prepare data for this matching phase, we group all entries from the expert dataset by their highest assigned topic probability. Then we filter out all documents with a maximum likelihood below 75%, which results in 11 high-probability expert datasets.

With a similar intuition as presented in transferability literature (Nguyen et al., 2020), we implemented a *loss-based assignment*. For the loss-based assignment, we loop over every expert dataset and model expert combination. We perform a forward pass while routing tokens only to the selected model expert and observe the language modeling (cross-entropy) loss. This process produces a ma-

Name	BLiMP	Supplement	ET	EWoK	Avg.
p_0.7_no_shared-base	62.04	60.00	14.63	48.68	46.34
p_0.7_no_shared-expert	65.07	53.20	<i>15.70</i>	50.14	46.03
p_0.7_shared_exp-base	61.22	62.00	39.96	51.00	53.55
p_0.7_shared_exp-expert	65.68	<i>57.60</i>	13.08	50.82	46.80
p_0.7_weighted_shared_exp-base	61.48	64.00	33.35	51.09	52.48
p_0.7_weighted_shared_exp-expert	66.26	55.60	13.78	51.82	46.87
k_1_shared_exp-base	61.21	62.00	39.82	50.36	53.35
k_1_shared_exp-expert	64.68	56.00	13.63	50.00	46.08

Table 1: Baseline compared to a standard shared expert, the weighted shared expert and the standard shared expert with top- k ($k = 1$). All models with shared experts consistently outperform the base except after the *expert phase* for Entity Tracking and some instances of EWoK. Best scores per task in **bold**, best after expert phase in *italics*.

trix that assigns a loss to each tuple (gold-label topic, model expert). In the final matching, gold-label topics are assigned to model experts in a 1-to-1 manner to minimize the total loss, following the assumption that a lower loss indicates a better fit.

In a comparable fashion we also decided to experiment with a *gradient-based assignment*. For the gradient-based assignment, we proceed as follows for every expert dataset: First, a forward-and-backward pass is performed that routes tokens to every expert with uniform probability. Afterward, we calculate the L2 norm of the gradients for each of the 11 experts across all layers. For all 11 expert datasets, this results in a matrix that assigns a gradient to each tuple (gold-label topic, model expert). In the final matching, gold-label topics are assigned to model experts in a 1-to-1 manner to minimize their gradient sum, following the assumption that a lower gradient indicates a better fit.

Finally, the *fixed assignment* naïvely assigns the first gold-label topic to the first model expert, and the following topics to the subsequent model experts, respectively. In this process, no performance metric is involved.

Experiment results are shown in Table 8. Unexpectedly, the fixed assignment achieved the highest average evaluation result, with none of the more sophisticated methods outperforming the naïve approach. It is subject to further investigation whether this is due to noise or due to the gradient-based or loss-based assignment methods being insufficient proxies. These findings lead us to choose a fixed assignment for our final model.

4.5 Parameter Adjustments

Though we have chosen our model size according to established scaling laws, we further tested a larger model by doubling the FFN size, context win-

dow, and vocabulary size to accommodate for potential underfitting due to sparse expert activation. To this end, we incorporate all changes discussed so far and conduct the following experiments using these findings.

Table 9 shows the results of five experiments, including a baseline with no changes, and four variants that always have an FFN doubled in size and combinations of context window (256/512) and vocabulary size (5k/12k) changes. The results clearly show that the expert model neither significantly improves nor degrades with any configuration change. For the sake of simplicity, we decided not to incorporate any tested configuration changes in our final submission.

4.6 Curriculum Learning & Complexity Reduction

Curriculum learning was a re-occurring approach in previous BabyLM challenges (Charpentier et al., 2025). We try to implement curriculum learning by sorting our dataset by the lix score, a simple measure of complexity. Developed by Björnsson (1968) (according to Anderson (1983), original source not traceable), it is calculated using the average sentence length and the percentage of words that have more than six letters. We additionally run experiments that remove the most complex data (e.g., keeping only the bottom 80% of complexity) from our dataset and replace it with samples from our synthetic dataset. Results are shown in Table 10. The average score increases from keeping only the bottom 60% to the bottom 80%. Afterward, when keeping the bottom 90%, the average score decreases again, with BLiMP accuracy still increasing slightly. Given these insights, we decided to replace the top 20% complex instances from our corpus for the final training run.

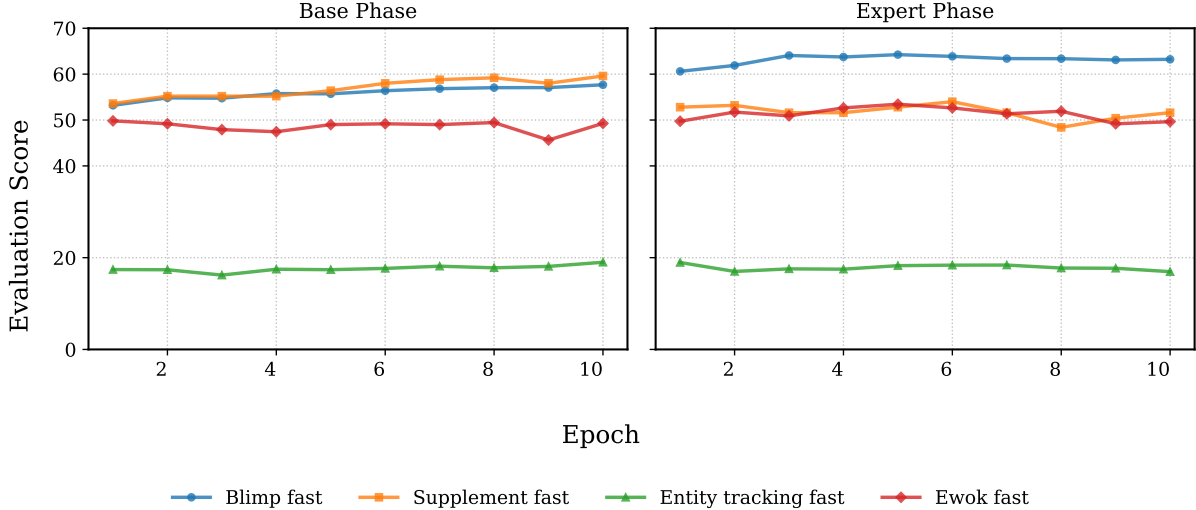


Figure 3: Submission model evaluation performance across different epochs

4.7 Final Setup

As our baselines, we use both the GPT-2-Strict-Small baseline, as reported by the BabyLM leaderboard, and a dense Llama 3 model trained on our custom dataset. The Llama baseline performs competitive with the reported GPT-2 Baseline. As can be seen in Table 2, our Baseline gets a slightly higher BLiMP score but a weaker Entity Tracking score. This is to be expected as our dataset contains fewer examples of simple children speech, which we consider particularly well-suited for tasks like entity tracking, but more examples of harder and higher quality text that improves grammar understanding as measured by BLiMP.

Name	BLiMP	Suppl.	ET	EWoK	Avg.
GPT-2	65.23	57.25	19.10	50.63	48.53
Llama 3	66.81	56.00	14.01	50.27	46.77
Base	57.68	59.6	19.0	49.27	46.39
Expert	63.23	51.6	16.95	49.64	45.36

Table 2: Zero-shot results: BabyLM-2026 GPT-2 Strict-Small baseline vs. our LLaMA dense baseline vs. our submission MoE models, evaluated after the *base phase* and the *expert phase* on BLiMP, BLiMP Supplement, Entity Tracking, and EWoK

The configuration of our submission model, shown in Table 3, was built by combining the findings of the comparative experiments mentioned in the above sections. Additionally, we submitted a second model, BabyDBRFull, that was trained using all experts, i.e., setting $p = 1.0$. The leaderboard results (best score of all epochs per task) for both models can be found in Table 4.

Configuration	Value
Top-p	0.7
Context Window	256
MoE Layers	2 MoE layers out of 4 placed alternately
Shared Expert	One weighted shared expert
Vocab Size	5,000
Training Objective	Load balancing (aux) loss + routing loss
Expert Assignment	Fixed expert assignment
Curriculum Learning	Order by text complexity (lix)
Data Complexity	The most complex 20% of training data was dropped

Table 3: Submission Model Configuration

Table 2 shows the final model evaluation results for GPT-2, LLaMA, and our submission model. Overall, the reported GPT-2 baseline, trained on the official BabyLM Strict-Small dataset, and the LLaMA baseline, trained on our base and expert dataset, outperform our expert submission model on average and on BLiMP in every setting. Some

Name	BLiMP	Suppl.	ET	EWoK	Avg.
Base	63.47	51.53	19.02	51.53	46.39
Full	65.26	49.78	18.29	50.76	46.02

Table 4: Leaderboard results for BabyDBR (Base) and BabyDBRFull (Full)

features to highlight include the resulting model from training phase one, which outperforms the GPT-2 and LLaMA baselines in BLiMP Supplement and the Llama baseline in Entity Tracking. In particular, BLiMP Supplement experiences a decline of 8 points between the *base phase* and

the *expert*, as noted in Section 4.1. While entity tracking slightly decreases and EWoK stays similar, BLiMP increases by almost 6 points. In addition, Figure 3 shows evaluation results for our submission model from epochs 1 to 10 during training on the base and expert datasets. Entity Tracking performance during training stays approximately constant.

5 Discussion

While our model does not outperform the full-sized baselines, it is worth noting that our model is significantly smaller in size, amounting to a total of 13.8M parameters versus 98.4M of the GPT-2 baseline. In particular, at submission time of this paper, our model is the best-performing on the strict-small leaderboard with less than 25M parameters (of which there are 9 verifiably below this limit).

Our experiments also show apparent dissonances between human-intent design and performance gains in language models, as seen in our router teacher forcing and expert assignment experiments: Clear separation and interpretability can hurt performance, and algorithm designs intended to exploit inert preference in random initializations may still perform worse than arbitrary choices.

As implied by our section on catastrophic forgetting and complexity reduction, the main question arises whether the data collected for our experiments is insufficient for good language modeling, or whether the given evaluation benchmarks are not a suitable proxy for measuring it. Since language models are often used for knowledge tasks, we propose a stronger focus on such evaluations in future editions of the workshop, but providing a fixed knowledge database for competition participants to keep the focus on language modeling.

6 Conclusion

We presented a structured MoE approach for the BabyLM Strict-Small track in which expert routing is supervised by externally derived, human-interpretable topic labels rather than learned purely implicitly. To this end, we constructed a custom 10M-word corpus, derived eleven topic domains via BERTopic clustering and manual curation, and conducted systematic ablations over routing mechanisms, MoE placement, shared experts, expert assignment strategies, curriculum ordering, and model size. Our best configuration, alternating MoE layers, a weighted shared expert, no router

teacher forcing, and fixed expert assignment, outperforms both the reported GPT-2 baseline and our dense LLaMA baseline on BLiMP when all experts are active, while the sparse variant roughly matches the dense baseline at lower compute cost. At the same time, our ablations paint a consistent and instructive picture: the domain-based routing model performs well given its size, and benchmark scores improve by increasing flexibility during the training process, whether through more active experts, load-balancing auxiliary losses, or router self-prediction instead of teacher forcing. The catastrophic forgetting observed in the specialization phase, however, suggests a mismatch between the training data and the evaluation target.

7 Outlook

To clarify whether the limitations we observe are fundamental or specific to our setup, an interpretability analysis of what the experts actually learn under domain supervision would show whether the intended topic specialization emerges at all, independent of benchmark scores.

While our router teacher forcing approach showed slightly worse results, it may be interesting to investigate whether the pure load balancing loss distributes general knowledge more evenly across the experts, thus improving performance similar to introducing a shared expert. Further experiments would be required to measure this effect, which we leave to future work.

Limitations

While we conducted our experiments with rigor, some caveats remain: For all our experiments, we only train our models and evaluate on the BabyLM fast subsets once per configuration using a single seed; more runs to better estimate the effect on model capabilities could improve our architecture further or suggest different findings. An extensive evaluation of the clustering result, which labeled each data point with one of 198 topic clusters was not possible due to time constraints; a manually curated dataset may improve the routing, but is very labor-intensive. Given our custom dataset, comparability with other BabyLM architectures may be limited; while our approach did not work with the provided dataset, other leaderboard entries may perform better using our own dataset.

Acknowledgments

We used Claude Opus 4.6, Claude Sonnet 4.6, Gemini 3.1 Pro, Gemini 3.5 Flash, Gemini 3.1 Flash, Deepseek V4 Flash, Deepseek V4 Pro, GPT-5 Mini to gather ideas for details of our experiments, write out syntax in the implementation, and review. The core approach and general ideas for ablation studies are our own. Deepseek V4 Flash was used to generate synthetic data.

References

- Jonathan Anderson. 1983. [Lix and rix: Variations on a little-known readability index](#). *Journal of Reading*, 26(6):490–496.
- Yuki Arase, Satoru Uchida, and Tomoyuki Kajiwar. 2022. [CEFR-based sentence difficulty annotation and assessment](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6206–6219, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Carl-Hugo Björnsson. 1968. *Läsbarhet: hur skall man som författare nå fram till läsarna?* Bokförlaget Liber.
- Mark Breuker. 2023. [Cefr labelling and assessment services](#). In Georg Rehm, editor, *European Language Grid: A Language Technology Platform for Multilingual Europe*, pages 277–282. Springer International Publishing, Cham.
- Lucas Charpentier, Leshem Choshen, Ryan Cotterell, Mustafa Omer Gul, Michael Y. Hu, Jing Liu, Jaap Jumelet, Tal Linzen, Aaron Mueller, Candace Ross, Raj Sanjay Shah, Alex Warstadt, Ethan Gotlieb Wilcox, and Adina Williams. 2025. [Findings of the third BabyLM challenge: Accelerating language modeling research with cognitively plausible data](#). In *Proceedings of the First BabyLM Workshop*, pages 399–420, Suzhou, China. Association for Computational Linguistics.
- Leshem Choshen, Ryan Cotterell, Mustafa Omer Gul, Jaap Jumelet, Tal Linzen, Aaron Mueller, Suchir Salhan, Raj Sanjay Shah, Alex Warstadt, and Ethan Gotlieb Wilcox. 2026. [BabyLM turns 4: Call for papers for the 2026 babyLM workshop](#). *Preprint*, arXiv:2602.20092.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. [Training verifiers to solve math word problems](#). *arXiv preprint arXiv:2110.14168*.
- Ben Cottier, Robi Rahman, Loredana Fattorini, Nestor Maslej, Tamay Besiroglu, and David Owen. 2025. [The rising costs of training frontier ai models](#). *Preprint*, arXiv:2405.21015.
- Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. 2024. [Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models](#). *Preprint*, arXiv:2401.06066.
- Lennart Finke, Chandan Sreedhara, Thomas Dooms, Mat Allen, Juan Diego Rodriguez, Noa Nabeshima, Thomas Marshall, and Dan Braun. 2025. [Parameterized synthetic text generation with simplestories](#). In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- S. Granger, M. Dupont, F. Meunier, H. Naets, and M. Paquot. 2020. [The international corpus of learner english \(version 3\)](#). Louvain-la-Neuve: Presses universitaires de Louvain.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. [The llama 3 herd of models](#). *Preprint*, arXiv:2407.21783.
- Maarten Grootendorst. 2022. [BERTopic: Neural topic modeling with a class-based tf-idf procedure](#). *arXiv preprint arXiv:2203.05794*.
- Suchin Gururangan, Mike Lewis, Ari Holtzman, Noah A. Smith, and Luke Zettlemoyer. 2021. [Demix layers: Disentangling domains for modular language modeling](#). *Preprint*, arXiv:2108.05036.
- 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. 2022. [Training compute-optimal large language models](#). In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, pages 30016–30030.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. [The curious case of neural text degeneration](#). In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Joseph Marvin Imperial, Abdullah Barayan, Regina Stodden, Rodrigo Wilkens, Ricardo Muñoz Sánchez, Lingyun Gao, Melissa Torgbi, Dawn Knight, Gail Forey, Reka R. Jablonkai, Ekaterina Kochmar, Robert Joshua Reynolds, Eugénio Ribeiro, Horacio Saggion, Elena Volodina, Sowmya Vajjala, Thomas François, Fernando Alva-Manchego, and Harish Tayyar Madabushi. 2025. [UniversalCEFR: Enabling open multilingual research on language proficiency assessment](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9703–9755, Suzhou, China. Association for Computational Linguistics.

- Anna A. Ivanova, Aalok Sathe, Benjamin Lipkin, Unnathi U. Kumar, Setayesh Radkani, Thomas H. Clark, Carina Kauf, Jennifer Hu, R. T. Pramod, Gabriel Grand, Vivian C. Paulun, Maria Ryskina, Ekin Akyürek, Ethan G. Wilcox, Nafisa Rashid, Leshem Choshen, Roger Levy, Evelina Fedorenko, Joshua Tenenbaum, and Jacob Andreas. 2025. [Elements of world knowledge \(EWoK \): A cognition-inspired framework for evaluating basic world knowledge in language models](#). *Transactions of the Association for Computational Linguistics*, 13:1245–1270.
- Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87.
- Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, L  lio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, and 7 others. 2024. [Mixtral of experts](#). *Preprint*, arXiv:2401.04088.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. [Scaling laws for neural language models](#). *arXiv preprint arXiv:2001.08361*.
- Najoung Kim and Sebastian Schuster. 2023. [Entity tracking in language models](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3835–3855, Toronto, Canada. Association for Computational Linguistics.
- Mahnaz Koupaee and William Yang Wang. 2018. [Wikihow: A large scale text summarization dataset](#). *Preprint*, arXiv:1810.09305.
- Junzhuo Li, Bo Wang, Xiuze Zhou, and Xuming Hu. 2025. [Dynamic expert specialization: Towards catastrophic forgetting-free multi-domain MoE adaptation](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 18478–18493, Suzhou, China. Association for Computational Linguistics.
- Margaret Li, Suchin Gururangan, Tim Dettmers, Mike Lewis, Tim Althoff, Noah A. Smith, and Luke Zettlemoyer. 2022. [Branch-train-merge: Embarrassingly parallel training of expert language models](#). *Preprint*, arXiv:2208.03306.
- Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. [DailyDialog: A manually labelled multi-turn dialogue dataset](#). In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 986–995, Taipei, Taiwan. Asian Federation of Natural Language Processing.
- Anton Lozhkov, Loubna Ben Allal, Leandro von Werra, and Thomas Wolf. 2024. [Fineweb-edu: the finest collection of educational content](#).
- Michael McCloskey and Neal J Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier.
- Sushant Mehta, Raj Dandekar, Rajat Dandekar, and Sreedath Panat. 2025. [Unifying mixture of experts and multi-head latent attention for efficient language models](#). In *Proceedings of the First BabyLM Workshop*, pages 42–51, Suzhou, China. Association for Computational Linguistics.
- Meta AI. 2024. [The llama 4 herd: The beginning of a new era of natively multimodal ai innovation](#).
- MultivexAI. 2026. [Everyday language corpus](#). <https://huggingface.co/datasets/MultivexAI/Everyday-Language-Corpus>.
- Tarek Naous, Michael J Ryan, Anton Lavrouk, Mohit Chandra, and Wei Xu. 2024. [ReadMe++: Benchmarking multilingual language models for multi-domain readability assessment](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12230–12266, Miami, Florida, USA. Association for Computational Linguistics.
- Cuong V. Nguyen, Tal Hassner, Matthias Seeger, and Cedric Archambeau. 2020. Leep: a new measure to evaluate transferability of learned representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML’20*. JMLR.org.
- Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. [Towards empathetic open-domain conversation models: A new benchmark and dataset](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381, Florence, Italy. Association for Computational Linguistics.
- Asankhaya Sharma. 2025. [The 1 billion token challenge: Finding the perfect pre-training mix](#).
- Noam Shazeer, Azalia Mirhoseini, Krzysztof Mazi  rz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. 2017. [Outrageously large neural networks: The sparsely-gated mixture-of-experts layer](#). In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24–26, 2017, Conference Track Proceedings*. OpenReview.net.
- Ana  s Tack, Thomas Fran  ois, Sophie Roekhaut, and C  drick F  iron. 2017. [Human and automated CEFR-based grading of short answers](#). In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 169–179, Copenhagen, Denmark. Association for Computational Linguistics.

- Joonas Tapaninaho. 2025. [MoEP: Modular expert paths for sample-efficient language modeling](#). In *Proceedings of the First BabyLM Workshop*, pages 540–547, Suzhou, China. Association for Computational Linguistics.
- Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, Mayank Chaturvedi, Victor Cotruta, Alice Coucke, Phil Culliton, Robert Dadashi, Lucas Dixon, Mohamed Elhawaty, Utku Evci, Clément Farabet, and 282 others. 2026. [Gemma 4 technical report](#). *Preprint*, arXiv:2607.02770.
- Peter Thwaites, Nathan Vandeweerd, and Magali Paquot. 2025. [Crowdsourced comparative judgement for evaluating learner texts: How reliable are judges recruited from an online crowdsourcing platform?](#) *Applied Linguistics*, 46(4):611–628.
- Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Bhargavi Paranjabe, Adina Williams, Tal Linzen, and Ryan Cotterell. 2023. [Findings of the BabyLM challenge: Sample-efficient pretraining on developmentally plausible corpora](#). In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 1–34, Singapore. Association for Computational Linguistics.
- Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. [BLiMP: A benchmark of linguistic minimal pairs for English](#). In *Proceedings of the Society for Computation in Linguistics 2020*, pages 409–410, New York, New York. Association for Computational Linguistics.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Menglin Xia, Ekaterina Kochmar, and Ted Briscoe. 2016. [Text readability assessment for second language learners](#). In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 12–22, San Diego, CA. Association for Computational Linguistics.

A Experiment Results

Name	BLiMP	Supplement	Entity Tracking	EWoK	Avg.
k_1_moe_full-base	66.14	55.60	20.95	51.27	48.49
k_1_moe_full-expert	61.49	51.60	18.96	48.59	45.16
k_1_moe_two_alternating-base	66.63	56.40	15.03	50.09	47.04
k_1_moe_two_alternating-expert	63.69	51.20	14.27	51.23	45.10
k_2_moe_full-base	66.25	56.00	20.57	51.41	48.56
k_2_moe_full-expert	62.67	50.80	20.07	51.41	46.24
k_2_moe_two_alternating-base	65.98	57.60	26.14	50.09	49.95
k_2_moe_two_alternating-expert	64.20	57.20	14.73	51.05	46.80
p_0.4_moe_full-base	66.18	55.20	20.93	51.32	48.41
p_0.4_moe_full-expert	62.40	52.40	20.22	48.86	45.97
p_0.4_moe_two_alternating-base	65.86	60.00	17.83	50.09	48.45
p_0.4_moe_two_alternating-expert	64.01	50.40	20.89	51.50	46.70

Table 5: Freezing Non-MoE (Phase 2) + Shift 30% to Basic Dataset + 12k Vocabulary. Best scores per task in **bold**.

Name	BLiMP	Supplement	ET	EWoK	Avg.
p_0.7_all_moe-base	62.04	60.00	14.63	48.68	46.34
p_0.7_all_moe-expert	65.07	53.20	15.70	50.14	46.03
p_0.7_two_moe_alternating-base	61.43	62.40	38.37	48.82	52.76
p_0.7_two_moe_alternating-expert	65.76	56.00	17.93	50.09	47.45
p_0.7_two_moe_top-base	61.78	62.40	24.69	50.95	49.96
p_0.7_two_moe_top-expert	65.00	54.40	13.67	50.23	45.83

Table 6: MoE All vs. Alternating vs. Top. Best scores per task in **bold**.

Name	BLiMP	Supplement	ET	EWoK	Avg.
p_0.7_no_teacher_with_aux-base	61.76	60.80	13.89	48.45	46.23
p_0.7_no_teacher_with_aux-expert	67.30	55.60	16.60	51.23	47.68
p_0.7_no_teacher_no_aux-base	61.02	60.80	15.98	50.55	47.09
p_0.7_no_teacher_no_aux-expert	66.07	52.80	14.35	50.73	45.99
k1_no_teacher_no_aux-base	62.23	60.40	24.66	49.64	49.23
k1_no_teacher_no_aux-expert	63.26	54.40	13.41	50.64	45.43

Table 7: Deactivated Router Teacher Forcing with Auxiliary Loss vs. without Auxiliary Loss vs. top- k without auxiliary loss. Best scores per task in **bold**, best after expert phase in *italics*.

Name	BLiMP	Supplement	ET	EWoK	Avg.
p_0.7_fixed_all_moe-base	62.15	61.60	13.48	50.32	46.89
p_0.7_fixed_all_moe-expert	66.03	51.60	25.23	51.23	48.52
p_0.7_grad_all_moe-base	62.04	60.00	14.63	48.68	46.34
p_0.7_grad_all_moe-expert	65.07	53.20	15.70	50.14	46.03
p_0.7_loss_all_moe-base	62.48	62.40	14.96	50.05	47.47
p_0.7_loss_all_moe-expert	65.63	55.60	19.09	51.77	48.02

Table 8: Expert Assignment Fixed vs. Gradient based vs. Loss based. Best scores per task in **bold**.

Name	BLiMP	Supplement	ET	EWoK	Avg.
default_size_default_context_5k_vocab-base	61.43	59.20	34.02	50.00	51.16
default_size_default_context_5k_vocab-expert	66.51	55.20	13.99	49.91	46.40
double_size_256_context_12kvocab-base	59.26	60.00	35.26	50.36	51.22
double_size_256_context_12kvocab-expert	66.33	53.20	13.62	51.18	46.08
double_size_256_context_5kvocab-base	60.37	61.20	21.40	49.45	48.11
double_size_256_context_5kvocab-expert	65.89	57.20	14.45	49.36	46.73
double_size_512_context_12kvocab-base	58.21	56.00	37.82	47.18	49.80
double_size_512_context_12kvocab-expert	66.38	55.20	14.85	51.18	46.90
double_size_512_context_5kvocab-base	58.36	58.80	29.12	48.18	48.62
double_size_512_context_5kvocab-expert	64.97	50.80	15.13	49.64	45.14

Table 9: Double Expert Context Window, Double MoE FFN Size, 5k/12k Vocabulary. Best scores per task in **bold**.

Name	BLiMP	Supplement	ET	EWoK	Avg.
p_0.7_curric-base	56.29	58.80	42.61	49.09	51.70
p_0.7_curric-expert	65.43	52.00	14.44	49.91	45.45
p_0.7_curric_lix_keep_0_6-base	58.63	62.80	28.96	51.00	50.35
p_0.7_curric_lix_keep_0_6-expert	63.78	55.20	27.02	50.27	49.07
p_0.7_curric_lix_keep_0_8-base	58.87	60.40	38.11	48.82	51.55
p_0.7_curric_lix_keep_0_8-expert	64.87	55.60	29.27	52.09	50.46
p_0.7_curric_lix_keep_0_9-base	58.39	61.20	13.60	49.55	45.69
p_0.7_curric_lix_keep_0_9-expert	65.35	51.60	13.78	52.27	45.75

Table 10: Curriculum Learning via Lix Score Keeping All Expert Phase Data vs. Retaining Bottom 60/80/90% of Lix Score. Best scores per task in **bold**.

B Fill Dataset Prompt

B.1 Prompt Base

Your task is to help expand a dataset for language model training by generating synthetic variations of input texts.

Given an English input example, generate a completely new example in English that mimics the structural complexity, length, and format (e.g., sentence, paragraph, or dialogue) of the input.

{topic_instruction}

Focus on creating a meaningful variation. Avoid simply rephrasing the input. The language level and vocabulary MUST be at most CEFR A2. Use natural, simple, everyday language suitable for a 5-year-old child.

The new example should have at least 4 words and at most 100 words. Avoid referring to colors.

DO NOT use titles and keep the text free of any markdown or similar formatting. Output ONLY the new example text, without any additional introductory or concluding remarks, commentary or explanations.

B.2 Prompt - Topic Instruction

B.2.1 Static - 30%

Change the topic, entities, and subject matter to create a distinct example. Entities in the new example can also remain the same as in the input if they are common and generic words a 5-year-old would know (e.g., "dog", "cat", "car"), but try to use different ones when possible.

B.2.2 Dynamic - 70%

Change the topic, entities, and subject matter to create a distinct example. If possible, make the new example involve the word "{topic_word}" to encourage topic diversity. However, if "{topic_word}" does not fit naturally, you can use it as a general theme, or choose a different common word that a 5-year-old would know.

C Dataset Composition

Parent	Name & Source	Content	Target Dataset	Basic / Expert Words	License
	FinWeb-Edu 100M (Sharma, 2025) a sample of FineWeb-Edu (Lozhkov et al., 2024)	Educational	Expert	0 / 2925611	odc-by
	Daily Dialog (Li et al., 2017)	Daily Dialogs	Base	920394 / 0	cc-by-nc-sa-4.0
	OpenAI GSM8K (Cobbe et al., 2021)	Mathematics	Expert	0 / 475550	mit
	Facebook Empathetic Dialogues (Rashkin et al., 2019); Empathetic Dialogues GitHub	Emotions / Empathy	Expert	0 / 576485	cc-by-nc-4.0
	SimpleStories (Finke et al., 2025)	Synthetic / Story	Expert	0 / 1100650	mit
	BabyLM Strict-Small Simple Wiki (Choshen et al., 2026)	Wiki	Expert	0 / 1020687	mit
	BabyLM Strict-Small Gutenberg (Choshen et al., 2026)	Multi-Domain	Expert	0 / 1622367	mit
	WikiHow Cleaned (Koupaei and Wang, 2018)	How-to descriptions	Expert	0 / 540500	cc-by-nc-sa-3.0
	Everyday Language Corpus (MultivexAI, 2026)	Everyday Language	Base	79200 / 0	cc-by-4.0
UniversalCEFR (Imperial et al., 2025)	CEFR_SP_EN (Arase et al., 2022)	News / Wiki / Educational	Base(A-B1+) / Expert(>B1+)	55379 / 92506	cc-by-nc-sa-4.0
	README_EN (Naous et al., 2024)	Multi-Domain	Base(A-B1+) / Expert(>B1+)	20105 / 35954	cc-by-nc-sa-4.0
	CEFR_ASAG_EN (Tack et al., 2017)	Question-answers	Base(A-B1+) / Expert(>B1+)	12596 / 7943	cc-by-nc-sa-4.0
	ICLE500_EN (Thwaites et al., 2025; Granger et al., 2020)	Learner-essay	Base(A-B1+) / Expert(>B1+)	4791 / 7927	cc-by-nc-4.0
	ELG_CEFR_EN (Breuker, 2023)	News / Magazines / Educational / Government	Base(A-B1+) / Expert(>B1+)	90113 / 38023	cc-by-nc-4.0
	CAMBRIDGE_EXAMS_EN (Xia et al., 2016)	Cambridge English Exams	Base(A-B1+) / Expert(>B1+)	24105 / 2507	cc-by-nc-sa-4.0

Table 11: Dataset Sources

D Hyperparameters

Hyperparameter	Value
<i>Model Architecture</i>	
Block Size	256
Max position embeddings	256
Expert Intermediate Size	512
Attention Heads	4
Key-Value Heads	4
Number of Hidden Layers	4
Hidden Activation	SiLU
MLP Bias	False
RoPE Theta	500,000.0
Attention Dropout	0.1
<i>Mixture-of-Experts (MoE) & Router</i>	
Number of Experts	11
Shared Expert Intermediate Size	512
Shared Expert Gate Init Std	0.02
Router Initialization Std	0.02
Router Aux Loss Coefficient	0.01
Router Loss Type	KL Divergence
<i>Training & Optimization</i>	
Learning Rate (Phase 1 & Phase 2)	12×10^{-4}
LR Scheduler Type (Phase 1 & Phase 2)	Cosine
Optimizer	PyTorch’s AdamW
Warmup Steps (Phase 1 / Phase 2)	25 / 100
Per-Device Train / Eval Batch Size	64 / 32
Gradient Accumulation Steps	1
Number of Training Epochs	10
Weight Decay	0.1
Max Gradient Norm	1.0
FP16 / BF16	False / False
Tokenizer Algorithm	BPE
Tokenizer Min Frequency	2
Random Seed	42

Table 12: Hyperparameters kept consistent across all experiments.