Babies Learn to Look Ahead: Multi-Token Prediction in Small LMs

Ansar Aynetdinov
Humboldt-Universität zu Berlin
aynetdia@hu-berlin.de

Alan Akbik
Humboldt-Universität zu Berlin
alan.akbik@hu-berlin.de

Abstract

Multi-token prediction (MTP) is an alternative training objective for language models that has recently been proposed as a potential improvement over traditional next-token prediction (NTP). Instead of training models to predict only the next token, as is standard, MTP trains them to predict the next k tokens at each step. While MTP was shown to improve downstream performance and sample efficiency in large language models (LLMs), smaller language models (SLMs) struggle with this objective. Recently, a curriculum-based approach was offered as a solution to this problem for models as small as 1.3B parameters by adjusting the difficulty of the training objective over time. In this work we investigate the viability of MTP curricula in a highly data- and parameter-constrained setting. Our experimental results show that even 130M-parameter models benefit from including the MTP task in the pre-training objective. These gains hold even under severe data constraints, as demonstrated on both zero-shot benchmarks and downstream tasks.

1 Introduction

Next-token prediction (NTP) is the predominant training objective for autoregressive language models. Learning to predict only one token at each generation step has guided the training of models like GPT (Brown et al., 2020; OpenAI et al., 2024) and LLaMA (Touvron et al., 2023a,b; Grattafiori et al., 2024), and Qwen (Qwen et al., 2025; Yang et al., 2025). Despite its simplicity, this training objective has led to remarkable advancements across text understanding, generation, and reasoning tasks. However, by restricting the prediction horizon to a single upcoming token, large language models (LLMs) may underexploit their ability to anticipate and plan over longer stretches of text.

Multi-token prediction (MTP) (Gloeckle et al., 2024) addresses this shortcoming by including multiple (k) subsequent tokens into the objective (see Figure 1 for an illustration). As a result, MTP was shown to improve model's downstream performance, inference speed, and training sample efficiency without significantly increasing training time. On a large scale, the MTP training objective was adopted by (Liu et al., 2024) for their Deepseek-V3 model that serves as a base model for the reasoning R1 model (Guo et al., 2025).

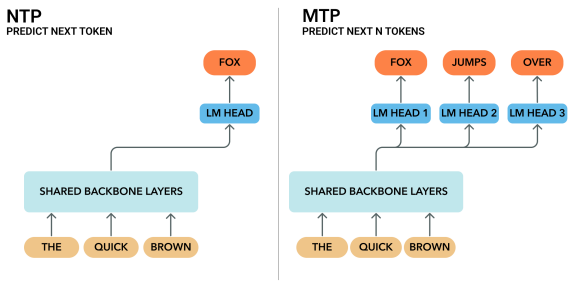

Figure 1: Visualization of MTP vs NTP. Instead of focusing on just the next upcoming token, in MTP multiple subsequent tokens are predicted at each step, using multiple parallel output heads that share a common model backbone (Gloeckle et al., 2024).

Nonetheless, MTP is not a free lunch: its benefits are most pronounced for models with sufficient capacity to handle the increased predictive complexity. When applied to smaller language models (SLMs, < 7B), the objective can even degrade performance, as these models often struggle to learn more complex morphological and semantic dependencies in parallel from the outset. To address this, Aynetdinov and Akbik (2025) proposed a curriculum-based approach to MTP for SLMs, that gradually adjusts the number of predicted tokens during training. By varying k over time, they showed that SLMs can better adapt to the MTP objective and recover some of the performance gains observed in larger models.

In this work, we push this approach even further by investigating the potential of curriculum-

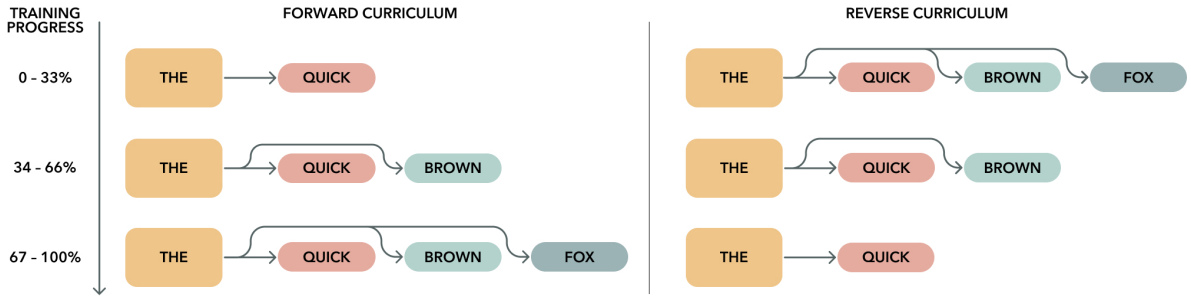

Figure 2: Visualization of the forward and reverse MTP curricula (Aynetdinov and Akbik, 2025). When training a language model on a 3-token-prediction task for n steps, the forward curriculum starts with a vanilla NTP task, adding an additional token to the task every $\frac{n}{3}$ steps. The reverse curriculum does the opposite, starting with a full 3-token-prediction task, and dropping a token from the task every $\frac{n}{3}$ steps.

based MTP training in an even more constrained regime, with models under 1B parameters, trained on just 10M words. Using a 130M-parameter GPT-2 model as our test case, we compare the vanilla NTP and static MTP training objectives against forward and reverse curricula proposed by Aynetdinov and Akbik (2025) in order to evaluate their effectiveness in both zero-shot and fine-tuned settings.

Contributions. This paper makes the following contributions:

- We extend the analysis of curriculum-based MTP objectives to models under 1B parameters trained on just 10M words.
- We provide a more detailed look at the training dynamics of MTP- vs NTP-based SLMs throughout multiple epochs.
- We showcase that very small LMs can still benefit from the MTP objective on additional tasks introduced in this iteration of the BabyLM challenge.

2 Preliminaries

In this section, we briefly formalize the multi-token prediction objective, as well as the curricula proposed by Aynetdinov and Akbik (2025), considered in this paper.

2.1 Multi-Token Prediction Objective

Large language models are usually trained with the next-token prediction (NTP) objective. Given a context sequence

$$\mathbf{x} = (x_1, x_2, \ldots, x_t),$$

the task is to predict the next token x_{t+1} by maximizing its conditional probability:

$$\mathcal{L}_{\text{NTP}} = -\sum_{t=1}^{T} \log P(x_{t+1} \mid x_1, \ldots, x_t; \theta),$$

where θ denotes model parameters.

The multi-token prediction (MTP) objective generalizes this setup to predicting a sequence of k future tokens $\mathbf{y} = (x_{t+1}, x_{t+2}, \ldots, x_{t+k})$ in parallel:

$$\mathcal{L}_{\text{MTP}} = -\sum_{t=1}^{T} \sum_{i=1}^{k} \log P(x_{t+i} \mid x_1, \ldots, x_t; \theta),$$

where probabilities are produced by k output heads that share the same model backbone.

2.2 Curriculum Schedules

The curricula vary the number of active prediction heads $k \in \{1, \ldots, k_{\max}\}$ across epochs. Updates occur at fixed intervals of $E/k_{\max}$ epochs, where E is the total training epochs. We consider two predefined variants: a forward and a reverse schedule. The forward curriculum mimicks the progression from an easy NTP to a more complext MTP task, while the reverse curriculum simulates the opposite.

Forward curriculum. Training starts with $k = 1$ and gradually increases the number of active heads:

$$k_{\text{current}}(e) = \min\left(k_{\max}, \left\lfloor \frac{e}{E/k_{\max}} \right\rfloor + 1\right).$$

Reverse curriculum. Training starts with $k = k_{\max}$ and progressively decreases the number of active heads:

$$k_{\text{current}}(e) = \max\left(1, k_{\max} - \left\lfloor \frac{e}{E/k_{\max}} \right\rfloor\right).$$

Objective	Curriculum	BLiMP (Acc.)	BLiMP Suppl. (Acc.)	EWoK (Acc.)	Entity Tracking (Acc.)	WUG Adj. Nom. (Acc.)	Eye Tracking (ΔR^2)	Self-paced Reading (ΔR^2)	Avg.
NTP	-	**62.17**	**59.48**	49.79	13.74	59.50	10.59	4.13	37.06
MTP	-	61.37	56.90	49.46	17.88	65.00	11.00	**4.30**	37.99
	Reverse	61.93	57.60	**50.22**	**18.60**	**66.00**	**11.17**	4.28	**38.54**
	Forward	61.51	58.29	49.73	13.40	60.00	10.15	4.00	36.73

Table 1: Zero-shot evaluation after training the models for 10 epochs on the 10M BabyLM dataset. **Best** scores are highlighted.

3 Experimental Setup

We aim to assess the impact of incorporating the MTP objective during pre-training of small language models in data-constrained settings. To enable a comparison with the baseline numbers published by the BabyLM challenge organizers (Charpentier et al., 2025), our experimental setup closely mirrors theirs. For further discussion of the training setup and associated computational costs, please refer to Appendix A.

Tokenizer and data. We use the provided BabyLM dataset mixture consisting of 10M words (`strict-small` track) for pre-training. We apply only minor pre-processing, mostly aimed at e.g. removing opening headers and closing footnotes in the Project Gutenberg subset, or speaker prefixes in the Childes and Switchboard subsets. Using the tokenizers provided with the baseline models, we tokenize and naively split or concatenate the text documents to fit the context window of 512. We also experiment with a tokenizer that has half the size of the baseline vocabulary, i.e. we compare tokenizers with 8K vs 16K subword tokens.

Model architecture. We conduct our experiments using a decoder-only GPT-2 transformer architecture with 130M parameters (Radford et al., 2019). As for the additional language modeling heads required for multi-token prediction, we opt for additional linear layers on top of the shared backbone. This introduces 8M or 16M additional trainable parameters depending on the vocabulary size, but keeps the amount of transformer layers the same between MTP and NTP models. For the purposes of the BabyLM challenge we keep the amount of maximum tokens in the pre-training objective limited to 2. Based on the empirical results provided by Gloeckle et al. (2024) and Aynetdinov and Akbik (2025), using more tokens in the objective would not be practically meaningful due to the increasing complexity of the objective and the smaller size of models considered in this work.

Training and evaluation. We consider 4 model configurations:

NTP A model we trained by replicating the setup used by the BabyLM organizers to train their baseline model, using a vanilla NLP objective.

MTP, Static A model trained using the 2-token prediction objective throughout all 10 epochs of pre-training.

MTP, Forward Curriculum A model trained following the forward MTP curriculum: in the first 5 epochs, we use a classical NTP objective in which we predict only the next token. Then, in the remaining 5 epochs we switch to an MTP objective to predict the next 2 tokens.

MTP, Reverse Curriculum A model trained following the reverse MTP curriculum: in the first 5 epochs, we immediately start with predicting the next 2 tokens. In the final 5 epochs, we switch to a standard NTP objective.

We train these models for 10 epochs using the hyperparameters that were used to train the baseline GPT-2 model for the `strict-small` track. For downstream zero-shot and fine-tuning evaluation we use the provided BabyLM pipeline (Charpentier et al., 2025). During evaluation all models perform only regular next token prediction for a controlled comparison of their performances.

4 Results

4.1 Result 1: Downstream Performance

4.1.1 Zero-shot evaluation

The results of the final zero-shot evaluation after training the aforementioned models for 10 epochs are listed in Table 1. We highlight some of the insights below. We also provide a comparison of

Objective	Curriculum	BoolQ (Acc.)	MNLI (Acc.)	MRPC (F1)	QQP (F1)	MultiRC (Acc.)	RTE (Acc.)	WSC (Acc.)	Avg.
NTP	-	**68.01**	50.10	80.14	61.78	63.53	**58.27**	63.46	63.61
MTP	-	66.97	50.41	**81.10**	62.09	**66.67**	56.12	63.46	63.83
	Reverse	67.77	48.92	80.26	**62.97**	66.01	57.55	63.46	**63.85**
	Forward	67.83	**50.61**	79.64	61.66	64.36	56.83	63.46	63.48

Table 2: Performance on SuperGLUE tasks after fine-tuning. **Best** scores are highlighted.

our baseline replication with the actual BabyLM baseline model on zero-shot tasks in Table 4 of Appendix B.

MTP forces SLMs to focus on patterns beyond local ones. Both the Static MTP and the Reverse Curriculum MTP models outperform the NTP model on Entity Tracking (Kim and Schuster, 2023) with a significant gap between them. This suggests that the MTP objective, even if used only for the first half of the pre-training, forces the model to "look ahead" more, i.e. better anticipate what comes next, and thus to better keep track of entity states in text sequences. This comes at the cost of being proficient at local syntactical, morphological, and semantic patterns, which is evident from all MTP-based models lagging behind the NTP model on the BLiMP benchmark (Warstadt et al., 2020) on average.

None of the models were able to acquire meaningful world knowledge. All models considered in our experiments do not score above 50 on EWoK (Ivanova et al., 2024), which means that none of the models perform better than a random guess on this benchmark. Since the MTP objective was not shown to have any significantly positive or negative impact on knowledge acquisition by language models, this is consistent with previous works. Therefore, this is evidence of scarce factual knowledge in the provided baseline dataset.

MTP leads to slightly more human-like text processing by LMs. The Eye Tracking score reflects how much of the variance in human eye fixation durations can be explained beyond what a simple regression using simple lexical features can, when taking the LM's predictions into account. If model's log probabilities for the next token can be a valuable predictor of eye fixation duration on that token, the LM mirrors the human eye movements when we read texts. The Self-Paced Reading works similarly, but also controls for spillover: it includes predictors for the preceding word's length and surprisal, so the LM only gets credit for predicting processing difficulty that is specific to the current word, independent of any carryover effects from the previous one (de Varda et al., 2024).

Slightly higher Eye Tracking and Self-Paced Reading scores of Reverse Curriculum and Static MTP models can be explained by previously discussed better anticipation of upcoming tokens. As a result, we argue that MTP mimics the way humans interact with text closer than NTP, given the nature of aforementioned scores - the MTP models tend be slightly more surprised by (i.e. assign lower probability to) the tokens or words on which the human readers tend to spend more time on.

Interestingly, the Forward Curriculum MTP model does not outperform the NTP model on these phenomena. This suggest that in our data-constrained setting with multiple training epochs the objective used early on in the training process seems to play a more important role when it comes to model performance.

4.1.2 Performance on classification tasks

When it comes to fine-tuning on downstream classification tasks, intuitively the amount of tokens in the prediction objective during pre-training of causal language models should not make a big difference. Table 2 showcases the performance of MTP and NTP models on SuperGLUE tasks (Wang et al., 2020), and there is indeed almost no difference between NTP- and MTP-based language models on average. The only task where a noticeable gap between between these models can be observed is MultiRC, where Static and Reverse Curriculum MTP models outperform the NTP model. Since MultiRC involves tracking the states of entities across multiple sentences to some extent, this can be explained by better zero-shot performance of Static and Reverse Curriculum MTP models on the Entity Tracking task.

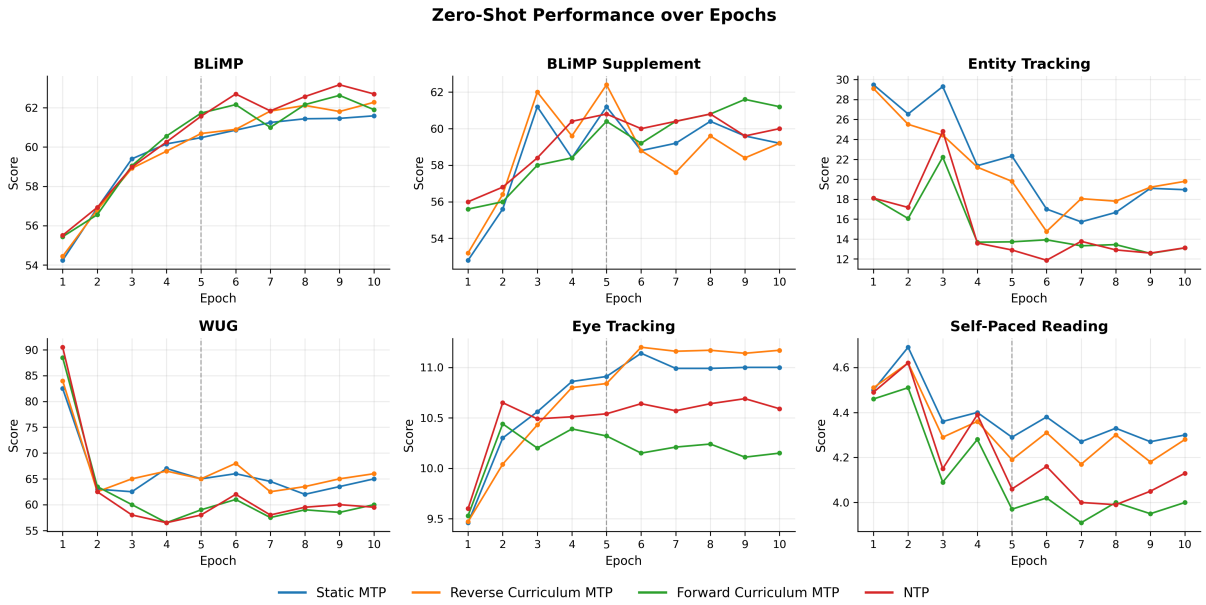

Figure 3: Zero-shot evaluation over epochs. The dotted line at epoch 5 indicates the switch in the training objective for models trained with either of the objective curricula. Tokenizer vocabulary size: **16K**.

4.2 Result 2: Performance over epochs

To assess how the timing of the pre-training objective influences downstream model performance, we analyze the zero-shot performance of our models over 10 epochs. Figure 3 shows results for all benchmarks except EWoK, where all models perform around chance level throughout all epochs. We show the performance development on EWoK in Figure 4 of Appendix B.

MTP objective acts as a regularizer in the early epochs. Prabhudesai et al. (2025) have shown that autoregressive LMs can overfit to repeated data after only a few epochs. We see a similar picture in our analysis: performance on Entity Tracking and WUG drops sharply after the first epoch.

However, models trained with the MTP objective from the beginning, i.e. Static MTP and Backward Curriculum, retain higher scores on these tasks for longer, while steadily improving on Eye Tracking. This suggests that MTP regularizes the learning signal in the earlier epochs, slowing the erosion of entity state tracking and morphological generalization. Higher accuracy on the WUG Adjective Normalization task (Hofmann et al., 2024) means that the models more consistently mirror human preferences about how to form nouns from novel adjectives, indicating stronger alignment with how humans do morphological generalization.

Forward Curriculum does not lead to improvements in the second stage of the training. After epoch 5, when Reverse Curriculum switches fully to NTP, it slightly overtakes Static MTP on BLiMP, suggesting that the model refines its representation learned in the first part of the pre-training.

In contrast, Forward Curriculum performs similarly to or worse than NTP, with improvements observed only on BLiMP Supplement. This suggests that introducing MTP in the second stage of training on repeated data does not replicate the benefits of early exposure, at least for the zero-shot tasks in the BabyLM evaluation pipeline, which use only a single language modeling head.

4.3 Ablation: Reduced vocabulary size

Aynetdinov and Akbik (2025) have shown that MTP-based byte-level SLMs outperform subword-level ones, partly because a subword token carries more semantic and morphological information than a byte, and therefore it is easier to predict multiple bytes, rather than subwords. Motivated by this observation, we additionally explore how the vocabulary size of a subword tokenizer would impact the performance and generalization abilities of even smaller MTP-based models. To this end, we trained identical counterparts of our models with a vocabulary size reduced by half.

Table 3 compares the performance of models trained on the initial vocabulary size of 16k against the models trained on half the initial vocabulary size of 8k. Generally we observe that the models showcase a very similar performance to each other,

Vocabulary Size	Objective	Curriculum	BLiMP (Acc.)	BLiMP Suppl. (Acc.)	EWoK (Acc.)	Entity Tracking (Acc.)	WUG Adj. Nom. (Acc.)	Eye Tracking (ΔR^2)	Self-paced Reading (ΔR^2)	Avg.
16k	NTP	-	**62.17**	**59.48**	49.79	13.74	59.50	10.59	4.13	37.06
	MTP	-	61.37	56.90	49.46	17.88	65.00	11.00	**4.30**	37.99
		Reverse	61.93	57.60	**50.22**	**18.60**	**66.00**	**11.17**	4.28	**38.54**
		Forward	61.51	58.29	49.73	13.40	60.00	10.15	4.00	36.73
8k	NTP	-	**61.91**	58.57	**49.51**	11.82	57.50	9.43	3.77	36.07
	MTP	-	61.25	56.81	49.12	16.25	**70.00**	8.92	3.61	**37.99**
		Reverse	61.36	56.61	49.22	**16.31**	66.50	8.58	3.58	37.45
		Forward	61.23	**59.10**	49.36	11.91	55.50	**9.48**	**3.83**	35.77

Table 3: Zero-shot evaluation of models using different tokenizer vocabulary sizes after training for 10 epochs on the 10M BabyLM dataset. **Best** scores are highlighted.

except for the performance on Eye Tracking and Self-Paced Reading benchmarks. Now, the Static and Reverse Curriculum models show a worse performance than NTP and Forward Curriculum models on these tasks.

Since a smaller vocabulary means that words tend to be split into more tokens, MTP models more often encounter situations in which they have to predict 2 tokens belonging to the same word at a given prediction step. At evaluation time, when all models are doing next-token prediction, this training bias potentially shows up as lower probability assigned to the word onset token. As a result, the whole-word surprisal gets noisier and aligns less with human data.

As for the absolute scores, a smaller vocabulary size has led to better final zero-shot scores only on WUG and BLiMP Supplement. We therefore do not observe conclusive positive effects of reducing the vocabulary size for any of the models considered in our experiments. We also show how the performance of models with a vocabulary size of 8k develops over the epochs in Figure 5 of Appendix B.

5 Related Work

Curriculum learning. Curriculum learning (CL) structures the order of training examples so that models progress from simpler to more complex cases (Bengio et al., 2009). Inspired by staged human learning, CL has been applied in computer vision, speech recognition, and NLP (Soviany et al., 2022), including encoder-only pre-training (Xu et al., 2020; Nagatsuka et al., 2021; Ranaldi et al., 2023) and instruction-tuning of decoder-only models (Mukherjee et al., 2023; Lee et al., 2024).

Although rarely used in the large-scale pre-training of publicly available decoder-only foundation models, Feng et al. (2024) showed that a two-stage, quality-based curriculum can improve training outcomes. By contrast, CL is common in data-constrained scenarios such as the BabyLM challenge (Hu et al., 2024). A meta-analysis of the 2023 BabyLM submissions (Warstadt et al., 2023) concluded that difficulty-based data ordering often matched shuffled baselines, whereas objective-level curricula tended to produce more reliable improvements.

For instance, Salhan et al. (2024) explored acquisition-inspired, cross-lingual curricula derived from age-ordered child-directed speech, pairing them with different objective-level strategies, and found consistent gains in small-scale model training. Hong et al. (2024) proposed Active Curriculum Language Modeling, involving dynamically selecting examples based on model uncertainty, which improved common-sense and world-knowledge performance.

Multi-token prediction. Next-token prediction (NTP) remains the dominant language modeling objective, but several works have explored predicting multiple future tokens in parallel (MTP). ProphetNet (Qi et al., 2020) was an early large-scale implementation, introducing a future n-gram objective with n-stream self-attention to attend to and predict multiple tokens at once, albeit with additional computational cost. Pal et al. (2023) found that NTP-trained models implicitly encode information about several future tokens in their hidden states, which can be partially recovered through probing.

Gloeckle et al. (2024) proposed a compute-matched MTP architecture using full transformer layers as separate language modeling heads, preserving efficiency while matching or exceeding the

performance of NTP models and enabling faster inference through parallel decoding. However, they reported that MTP objective can lead to performance degradation in models with less than 7B parameters. Aynetdinov and Akbik (2025) addressed this issue by proposing pre-training curricula that allow SLMs to recover some of the performance gains enjoyed by larger LMs.

Cai et al. (2024), on the other hand, showed that it is possible to enable multi-token prediction in larger models pre-trained on the next-token prediction task only. This allows to speed up the inference speed of already trained models by enabling self-speculative decoding (Stern et al., 2018).

6 Conclusion

In this paper we explored the viability of using the multi-token prediction objective for training very small language models in a data-constrained setting posed by the BabyLM challenge. We tested both static and curriculum-based training strategies for the MTP objective against a model trained using a regular next token prediction objective. Our experimental results show that the MTP objective has its merit even at a scale of 130M model parameters, when evaluated using the BabyLM pipeline. In fact, the model trained under a reverse MTP curriculum outperformed the NTP baseline on all zero-shot evaluation tasks except for BLiMP.

The analysis of model performances throughout the training process revealed that the MTP objective functions as an early-phase regularizer on repeated, small corpora: it slows the erosion of non-local language patterns learned in the first epochs. The difference in the pre-training showed a very limited effect on downstream classification performance on SuperGLUE after fine-tuning, and the available BabyLM data mixture does not support meaningful world-knowledge acquisition via causal language modeling regardless of objective. Reducing the subword vocabulary largely preserved the same qualitative picture and offered no meaningful advantage neither to MTP-based, nor to NTP-based models.

In data- and parameter-constrained settings such as the one considered in this work, employing a reverse MTP curriculum during pre-training yields better downstream performance while maintaining the same final inference speed as using only the NTP objective. In contrast, the forward curriculum produced the lowest average zero-shot performance. We attribute this to the model becoming trapped in a local minimum caused by overfitting during the early training stages, with the subsequent increase in task difficulty further reinforcing rather than alleviating the suboptimal performance. Thus, if the goal is to increase inference speed, using a static MTP objective is more preferable in settings similar to the one considered in this work.

In the future we would like to use the MTP objective for pre-training slightly larger models, but still under 1B parameters, on somewhat larger datasets, such as the one used in the Strict track of the BabyLM challenge. We also see value in extending the evaluation of MTP-based models to include generative tasks, such as abstractive summarization, which could provide a richer assessment of their capabilities.

Limitations

One limitation of our experimental setup is the fact that we used MTP curricula that were pre-defined in advance. The decision to progressively add or remove a token to or from a 2-token objective in the middle of the training is arbitrary, since it does not rely on any metrics about the models themselves or the training loss. This means that dropping or adding the additional token from or to the objective was done perhaps at a suboptimal point in the training process, leaving additional performance improvements at the table. However, the goal of this paper was to establish that the MTP objective has any merit in a data- and parameter-constrained setting of a BabyLM challenge. We plan to improve on this aspect of our experiments in the future iterations of the BabyLM challenge.

Furthermore, models capable of multi-token prediction can also support self-speculative decoding, which has potential both for efficiency gains and for deeper analysis of model behavior. In this work, we did not explore this aspect, focusing instead on controlled comparisons within the BabyLM evaluation pipeline. Future work could incorporate such decoding strategies to examine how MTP-trained models differ from NTP-trained ones in real generation settings, potentially revealing qualitative differences that are not captured by the current benchmarks.

References

Ansar Aynetdinov and Alan Akbik. 2025. Pre-training curriculum for multi-token prediction in language

models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 25573–25588, Vienna, Austria. Association for Computational Linguistics.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 41–48, New York, NY, USA. Association for Computing Machinery.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D. Lee, Deming Chen, and Tri Dao. 2024. Medusa: Simple llm inference acceleration framework with multiple decoding heads. *Preprint*, arXiv:2401.10774.

Lucas Charpentier, Leshem Choshen, Ryan Cotterell, Mustafa Omer Gul, Michael Hu, Jaap Jumelet, Tal Linzen, Jing Liu, Aaron Mueller, Candace Ross, Raj Sanjay Shah, Alex Warstadt, Ethan Wilcox, and Adina Williams. 2025. Babylm turns 3: Call for papers for the 2025 babylm workshop. *Preprint*, arXiv:2502.10645.

Andrea Gregor de Varda, Marco Marelli, and Simona Amenta. 2024. Cloze probability, predictability ratings, and computational estimates for 205 english sentences, aligned with existing eeg and reading time data. *Behavior Research Methods*, 56(5):5190–5213.

Steven Feng, Shrimai Prabhumoye, Kezhi Kong, Dan Su, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Maximize your data's potential: Enhancing llm accuracy with two-phase pretraining. *Preprint*, arXiv:2412.15285.

Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Rozière, David Lopez-Paz, and Gabriel Synnaeve. 2024. Better & faster large language models via multi-token prediction. *Preprint*, arXiv:2404.19737.

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, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, and Damien Allonsius et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, and Jin Chen et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Valentin Hofmann, Leonie Weissweiler, David Mortensen, Hinrich Schütze, and Janet Pierrehumbert. 2024. Derivational morphology reveals analogical generalization in large language models. *Preprint*, arXiv:2411.07990.

Xudong Hong, Sharid Loáiciga, and Asad Sayeed. 2024. A surprisal oracle for when every layer counts. In *The 2nd BabyLM Challenge at the 28th Conference on Computational Natural Language Learning*, pages 237–243, Miami, FL, USA. Association for Computational Linguistics.

Michael Y. Hu, Aaron Mueller, Candace Ross, Adina Williams, Tal Linzen, Chengxu Zhuang, Ryan Cotterell, Leshem Choshen, Alex Warstadt, and Ethan Gotlieb Wilcox. 2024. Findings of the second babylm challenge: Sample-efficient pretraining on developmentally plausible corpora. *Preprint*, arXiv:2412.05149.

Anna A. Ivanova, Aalok Sathe, Benjamin Lipkin, Unnathi Kumar, Setayesh Radkani, Thomas H. Clark, Carina Kauf, Jennifer Hu, R. T. Pramod, Gabriel Grand, Vivian Paulun, Maria Ryskina, Ekin Akyurek, Ethan Wilcox, Nafisa Rashid, Leshem Choshen, Roger Levy, Evelina Fedorenko, Joshua Tenenbaum, and Jacob Andreas. 2024. Elements of world knowledge (ewok): A cognition-inspired framework for evaluating basic world knowledge in language models. *arXiv preprint arXiv:2405.09605*.

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.

Bruce W Lee, Hyunsoo Cho, and Kang Min Yoo. 2024. Instruction tuning with human curriculum. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1281–1309, Mexico City, Mexico. Association for Computational Linguistics.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, and Kai Hu et al. 2024. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. 2023. Orca: Progressive learning from complex explanation traces of gpt-4. *Preprint*, arXiv:2306.02707.

Koichi Nagatsuka, Clifford Broni-Bediako, and Masayasu Atsumi. 2021. Pre-training a BERT with curriculum learning by increasing block-size of input text. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 989–996, Held Online. INCOMA Ltd.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, and Chester Cho et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Koyena Pal, Jiuding Sun, Andrew Yuan, Byron Wallace, and David Bau. 2023. Future lens: Anticipating subsequent tokens from a single hidden state. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pages 548–560, Singapore. Association for Computational Linguistics.

Mihir Prabhudesai, Mengning Wu, Amir Zadeh, Katerina Fragkiadaki, and Deepak Pathak. 2025. Diffusion beats autoregressive in data-constrained settings. *Preprint*, arXiv:2507.15857.

Weizhen Qi, Yu Yan, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. ProphetNet: Predicting future n-gram for sequence-to-SequencePre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2401–2410, Online. Association for Computational Linguistics.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Leonardo Ranaldi, Giulia Pucci, and Fabio Massimo Zanzotto. 2023. Modeling easiness for training transformers with curriculum learning. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 937–948, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Suchir Salhan, Richard Diehl Martinez, Zébulon Goriely, and Paula Buttery. 2024. Less is more: Pre-training cross-lingual small-scale language models with cognitively-plausible curriculum learning strategies. In *The 2nd BabyLM Challenge at the 28th Conference on Computational Natural Language Learning*, pages 174–188, Miami, FL, USA. Association for Computational Linguistics.

Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. 2022. Curriculum learning: A survey. *Preprint*, arXiv:2101.10382.

Mitchell Stern, Noam Shazeer, and Jakob Uszkoreit. 2018. Blockwise parallel decoding for deep autoregressive models. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti

Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, and Pushkar Mishra et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2020. Superglue: A stickier benchmark for general-purpose language understanding systems. *Preprint*, arXiv:1905.00537.

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: The benchmark of linguistic minimal pairs for english. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Benfeng Xu, Licheng Zhang, Zhendong Mao, Quan Wang, Hongtao Xie, and Yongdong Zhang. 2020. Curriculum learning for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6095–6104, Online. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

A Further Training Details

Following the setup of the baseline models of the BabyLM challenge, we trained 130M-sized GPT-2 models with at most 16M additional trainable parameters (auxiliary language modeling head for the MTP task). We explored various learning rates schedules and values, as well as batch sizes, but found that the batch size of 16 and the maximum learning rate of 5e-5 with 1% of warmup steps and cosine decay to 10% of the maximum learning rate worked best across all training objectives based on zero-shot benchmark performances. All experiments were done using the AdamW optimizer (Loshchilov and Hutter, 2019) with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1e-8$. The dropout rates in the model were kept at 0.1. Each model was trained at full fp32 precision on a single RTX A6000 GPU. Each training run lasted roughly 1.1 GPU hours on average.

Regarding the computational costs introduced by the MTP objective during pre-training, using a naive implementation approach without any dedicated optimizations, the 2-token MTP objective increases pre-training time in our setup by around 10% in terms of GPU hours. The memory requirements increased proportionate to the increase in trainable parameters, dictated by the vocabulary and hidden layer sizes. However Gloeckle et al. (2024) proposed a memory-efficient implementation of MTP pre-training that keeps the VRAM requirements the same as in NTP-based pre-training.

B Additional Evaluation Results

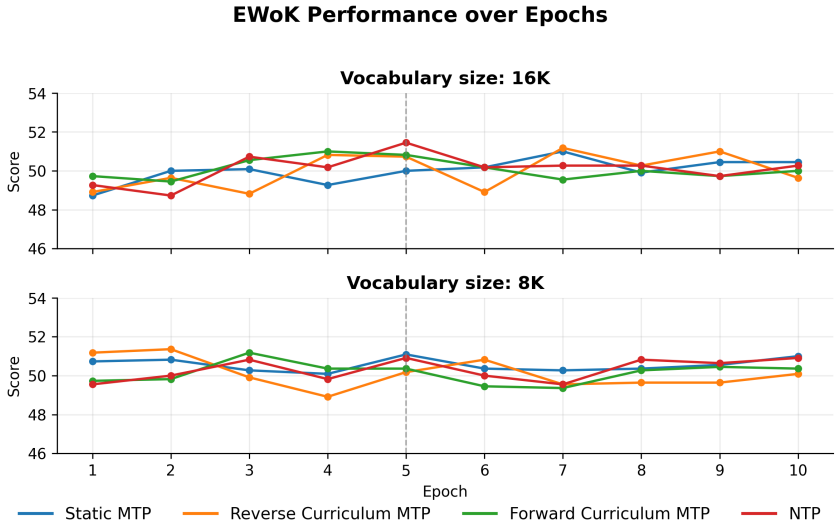

Figure 4: Zero-shot performance on EWoK over epochs. The performances are listed for models with both vocabulary sizes. The dotted line at epoch 5 indicates the switch in the training objective for models trained with either of the objective curricula.

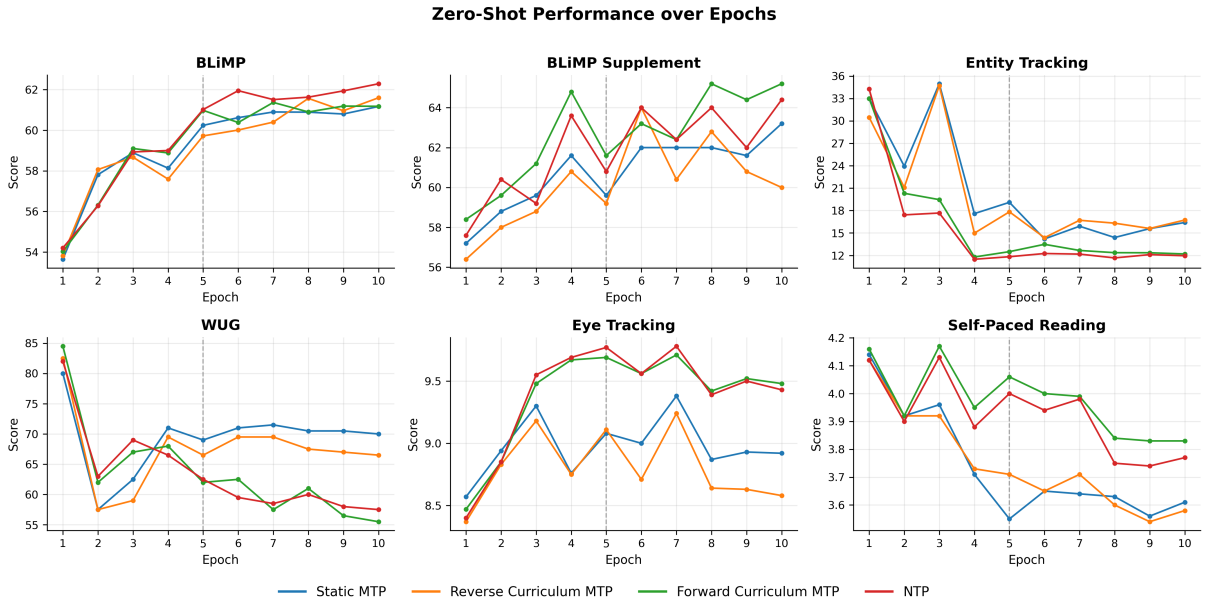

Figure 5: Zero-shot evaluation over epochs. The dotted line at epoch 5 indicates the switch in the training objective for models trained with either of the objective curricula. Tokenizer vocabulary size: **8K**.

Model	BLiMP (Acc.)	BLiMP Suppl. (Acc.)	EWoK (Acc.)	Entity Tracking (Acc.)	WUG Adj. Nom. (Acc.)	Eye Tracking (ΔR^2)	Self-paced Reading (ΔR^2)	Avg.
BabyLM Baseline	**66.36**	57.07	49.90	13.90	52.50	8.66	4.34	36.10
Baseline Replication	62.17	**59.48**	49.79	13.74	59.50	10.59	4.13	37.06
Static MTP	61.37	56.90	49.46	<u>17.88</u>	<u>65.00</u>	<u>11.00</u>	**4.30**	**37.99**
Reverse MTP Curriculum	<u>61.93</u>	57.60	**50.22**	**18.60**	**66.00**	**11.17**	<u>4.28</u>	**38.54**
Forward MTP Curriculum	61.51	<u>58.29</u>	49.73	13.40	60.00	10.15	4.00	36.73

Table 4: Zero-shot performance comparison against the `strict-small` (10M) BabyLM baseline. Tokenizer vocabulary size: **16K**. **Best** and <u>second-best</u> scores are highlighted. The differences between the baseline model and our replication of it can be explained by potential differences in the learning rate scheduler and data preprocessing. We used the cosine scheduler that anneals to 10% of the maximum learning rate. Our NTP, Static NTP and Reverse MTP Curriculum models outperform the BabyLM baseline on all benchmarks, except for BLiMP.