Master Thesis

Learning with Less: Contrastive Weight Tying on the BabyLM Challenge

Ino van de Wouw

Supervisor	Prof. Dr. Antske Fokkens
2^{nd} reader	Dr. Pia Sommerauer

a thesis submitted in fulfillment of the requirements for the degree of

MA Linguistics

(Text Mining)

Vrije Universiteit Amsterdam

Computational Linguistics and Text-Mining Lab
Department of Language and Communication
Faculty of Humanities

Date	June 27, 2025
Student number	2819716
Word count	10876

Abstract

The exponential growth in computational requirements for training state-of-the-art language models has created significant accessibility barriers, limiting advanced AI capabilities to well-resourced institutions. This research investigates the effectiveness of headless language models using Contrastive Weight Tying (CWT) as an alternative to traditional pre-training objectives, evaluated on the developmentally constrained BabyLM challenge datasets. Unlike conventional approaches that optimize vocabulary prediction through cross-entropy loss, CWT eliminates the projection head and employs contrastive learning with in-batch negative sampling to learn representations.

We systematically compare headless and vanilla architectures across both Masked Language Model (MLM) and Generative Pre-trained Transformer (GPT) configurations, training on 10-million and 100-million token datasets that approximate human child language exposure. Models are evaluated on the GLUE benchmark tasks requiring downstream fine-tuning and the BLiMP grammatical competence measures using zero-shot evaluation. All GLUE experiments are conducted across multiple random seeds to ensure statistical reliability.

The results demonstrate that headless models achieve substantial computational efficiency gains, delivering 32-34% faster training times for MLM architectures and 53% improvement for GPT models, while maintaining competitive downstream performance. On GLUE tasks, the vanilla MLM models show slight advantages on smaller datasets (59.9% vs 58.8%), but this gap narrows considerably with increased training data (71.51% vs 70.98%). Both the vanilla and headless GPT models failed to adequately perform the GLUE tasks. The MLM architectures struggle with grammatical knowledge on BLiMP (achieving near-random 50% performance), GPT models excel regardless of training objective (68% accuracy), suggesting autoregressive objectives may be more effective for grammatical competence acquisition from constrained corpora.

This work contributes to efficient pre-training literature by demonstrating that sophisticated linguistic representations can emerge through alternative training paradigms that prioritize embedding quality over explicit vocabulary prediction. The computational efficiency improvements support democratization of language model research by enabling training on resource-constrained hardware, while the scaling dynamics reveal important insights about data efficiency and architectural trade-offs. The findings challenge assumptions about the necessity of traditional prediction heads and provide practical guidance for researchers working under computational constraints, ultimately advancing toward more sustainable and accessible AI development.

Declaration of Authorship

I, author, declare that this thesis, titled *Learning with Less: Contrastive Weight Tying on the BabyLM Challenge* and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a degree degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: 27-6-2025

Signed: Ino van de Wouw

Acknowledgments

I would like to express my sincere gratitude to all those who contributed to the completion of this master's thesis.

First and foremost, I am deeply grateful to Antske Fokkens for their invaluable guidance and expertise throughout this research journey. Their insightful feedback, constructive criticism, and encouragement have been instrumental in shaping both this work and my development as a researcher. The knowledge and skills I have gained under their supervision will undoubtedly serve me well in my future endeavours.

I would also like to extend my heartfelt appreciation to Sophie Arnoult for their significant contributions to this project. Their expertise of the ADA-HPC Cluster was instrumental in achieving multiple important advances in this work.

Special thanks go to Fleur, whose support and encouragement have been a constant source of motivation. The insights I gained from our conversations about these topics were instrumental in advancing my thinking and progress.

Finally, I would like to acknowledge the broader academic community and my fellow students who have provided feedback through the Research Seminar discussion, giving key insights in how to improve the explanation of the Contrastive Weight Tying objective.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

The exponential growth in language model capabilities has come at an increasingly prohibitive computational cost, creating a fundamental accessibility barrier that threatens to concentrate advanced AI capabilities within well-resourced institutions (Brown et al., 2020, Cottier et al., 2024, Hoffmann et al., 2022). While state-of-the-art models like GPT-4 and Claude demonstrate remarkable linguistic competence, their training requirements—measured in petaFLOPs and billions of parameters—place them beyond the reach of most academic institutions, independent researchers, and developing economies (Chowdhery et al., 2023, Team et al., 2023, Touvron et al., 2023).

This computational bottleneck has sparked intensive research into efficient pretraining methodologies that can achieve sophisticated language understanding without the massive resource requirements of traditional approaches (Bender et al., 2021, Strubell et al., 2020). Whilst this resource concentration not only limits scientific participation but also raises concerns about the environmental sustainability of AI development, as training large language models can emit carbon equivalent to hundreds of transatlantic flights (Dodge et al., 2022, Luccioni et al., 2023, Strubell et al., 2020).

The efficiency challenge extends beyond mere resource constraints to fundamental questions about learning optimality. Humans achieve sophisticated linguistic competence with exposure to vastly less data than current computational models require, a difference that suggests substantial room for improvement in how artificial systems learn language (Lake et al., 2015, Vong et al., 2024, Warstadt et al., 2023a). This observation has motivated research into sample-efficient learning approaches that can achieve competitive performance with dramatically reduced data and computational requirements (Ouyang et al., 2022, Wang et al., 2023).

The efficiency challenge has garnered significant attention in the field, with researchers developing diverse approaches to reduce computational requirements while maintaining performance. Notable advances include Mixture of Experts architectures that achieve 4-5x faster pretraining (Jiang et al., 2024), parameter-efficient fine-tuning methods like QLoRA that enable large model training on consumer hardware (Dettmers et al., 2023), and memory-efficient attention mechanisms such as FlashAttention that provide substantial memory reductions (Dao et al., 2022). This thesis combines two such approaches: Headless Language Models and BabyLM methodologies (Godey et al., 2024, Warstadt et al., 2023a). Headless language models eliminate the standard prediction head during pretraining, instead using contrastive weight tying to learn representations through embedding reconstruction rather than token prediction, achieving up to 20x reduction in training computational requirements (Godey et al., 2024). Godey et al. (2024) showed that his headless language model was able to achieve comparable scores on the GLUE (Wang et al., 2018) benchmark as the vanilla counterpart. The BabyLM approaches address data efficiency by training on developmentally plausible corpora that approximate human child language exposure, constraining models to learn from 10-100 million word tokens—comparable to human language acquisition at the age of 4 and 12 years old—promoting architectural innovations and curriculum learning strategies (Charpentier and Samuel, 2023, Warstadt et al., 2023a). They evaluate model performance by utilising both the GLUE and BLiMP benchmark, the latter tests grammatical knowledge by using sentence pairs, that contrast in acceptability, across syntax, morphology and semantics (Warstadt et al., 2020).

To investigate the effect of combining the BabyLM corpora with headless language modelling I pose the following two research question:

- How does a headless language model trained on the BabyLM 10-million and 100-million token dataset compare in performance to a standard prediction-headed model, and under what conditions might it outperform its traditional counterpart?

The following hypotheses explore the potential advantages and limitations of headless language models compared to their traditional counterparts. These hypotheses will collectively address the research question.

- Hypothesis 1: The Headless Language Model will demonstrate superior computational efficiency during training, resulting in less memory consumption and increased training speed.

- Hypothesis 2: The Headless Language Model will achieve improved performance on GLUE and BLiMP benchmarks.

The results demonstrate that headless language models using contrastive weight tying (CWT) achieve the predicted computational efficiency gains while maintaining competitive performance under certain conditions. Headless models consistently outperformed vanilla counterparts in training speed, achieving 32-34% faster training times for masked language models and an impressive 53% improvement for generative models. On downstream task performance (GLUE), vanilla models generally maintained slight advantages on GLUE benchmarks, particularly on smaller datasets (59.9% vs 58.8% on STRICT-small), but this gap narrowed substantially with increased training data (71.51% vs 70.98% on STRICT). Notably, headless models demonstrated superior scaling properties, benefiting more from additional training data than their vanilla counterparts.

The study reveals important insights about the relationship between training objectives, model architectures, and linguistic competence acquisition. While both headless and vanilla masked language models struggled with grammatical knowledge on BLiMP benchmarks (achieving near-random performance around 50%), generative pre-trained transformer architectures excelled at these tasks regardless of training objective (achieving 68% accuracy). The findings indicate that the limitations in absolute performance stem primarily from the constrained nature of the BabyLM datasets rather than inherent architectural weaknesses, challenging assumptions about the necessity of explicit vocabulary prediction for meaningful representation learning.

This work contributes to the efficient pre-training literature by investigating how alternative training methodologies can achieve competitive performance with dramatically reduced computational requirements. The research addresses fundamental questions about the necessity of traditional architectural components and training procedures, providing insights that can inform future model design decisions.

The broader significance of this research extends to making advanced language model capabilities more accessible to diverse research communities worldwide. By demonstrating that sophisticated linguistic competence can emerge from constrained computational budgets, this work supports the democratization of AI research and development and is accessible through GitHub[1]. The environmental implications of such efficiency improvements align with growing recognition that sustainable AI development requires more intelligent approaches to learning and inference rather than simply scaling existing methods.

Through systematic evaluation of efficiency-performance trade-offs and comprehensive analysis of computational requirements, this research provides practical insights for researchers and practitioners working under resource constraints.

[1] https://github.com/IkNOw57/CWT-BabyLM-Thesis/

Chapter 2

Background and Related Work

In this chapter, first I provide a global overview of Language Models and the main approaches for creating efficient language models, before diving deeper into the two specific methodologies that form the core of this study. Then, we examine Headless Language Modelling, an approach that focusses on leveraging the power of the transformer body while reconsidering the traditional role of prediction heads. Subsequently, we examine the BabyLM shared task, which offers a compelling framework for investigating how language models can learn more effectively from limited data.

2.1 Language Models and Transformers

Modern language models are neural networks designed to understand and generate human language by learning statistical patterns from vast amounts of text data. At their core, these models are trained on unlabelled text through self-supervised learning by predicting masked or next words without requiring explicit human annotations. This pre-training process enables them to develop rich representations of language that can then be applied to diverse natural language processing (NLP) tasks through fine-tuning or direct application.

The transformer architecture, introduced by Vaswani et al. (2017), revolutionized natural language processing by processing information simultaneously rather than step-by-step, through the use of attention mechanisms. The architecture consists of stacked layers containing self-attention mechanisms and feed-forward neural networks.

Self-attention is a mechanism that enables each element in a sequence to directly compute weighted relationships with every other element in that sequence through learned transformations, allowing efficient capture of long-range dependencies. The feed-forward neural network takes the output of the self-attention and sends it through layers of connected nodes, with each layer performing mathematical operations that help the network learn to recognize patterns and make predictions.

This design has proven highly scalable and adaptable, forming the foundation for modern language models like BERT and GPT that have transformed the field through their ability to learn from vast amounts of text and transfer knowledge across diverse language tasks.

2.2 Research on Efficiency and Language Models

To position my approach within the current research on efficient Language Models, we examine existing studies to understand what has already been investigated and identify where my contribution fits within the broader field.

2.2.1 Knowledge Distillation and Parameter Reduction Approaches

Knowledge distillation has emerged as one of the most successful paradigms for creating efficient language models. The foundational work of Sanh et al. (2020) demonstrated that smaller models could retain 97% of BERT's performance with 40% fewer parameters through careful teacher-student training regimes. DistilBERT's innovation lay in pre-training-phase distillation rather than task-specific compression, using a triple loss function combining language modelling, distillation, and cosine embedding losses. This approach achieved 60% faster inference while maintaining near-teacher performance across downstream tasks, establishing distillation as a viable path to practical efficiency gains.

Building upon this foundation, Jiao et al. (2020) advanced transformer-specific distillation through layer-wise knowledge transfer. TinyBERT achieved 96.8% of BERT-base performance with 28% fewer parameters and 31% faster inference through a sophisticated two-stage training approach. The methodology involved general distillation followed by task-specific distillation, incorporating attention matrix distillation and hidden state transfer via linear mapping layers. This work demonstrated that careful attention to the distillation process could achieve extreme compression ratios while preserving model capabilities.

The distillation paradigm has continued to evolve with increasingly sophisticated approaches. MobileBERT (Sun et al., 2020) introduced bottleneck architectures specifically designed for mobile deployment, achieving 99.2% GLUE performance with 4× compression through deep-narrow design principles. Self-distillation approaches (Zhang et al., 2021) have shown that models can improve their own representations through iterative refinement, while progressive distillation enables gradual compression without catastrophic performance degradation.

2.2.2 Parameter-Efficient Fine-Tuning Methodologies

The development of parameter-efficient fine-tuning methods has revolutionized how practitioners adapt large language models for specific tasks. Low-Rank Adaptation (LoRA) (Hu et al., 2021) represents a breakthrough in this domain, freezing pre-trained weights while injecting trainable rank decomposition matrices. This approach reduces trainable parameters by 10,000× for GPT-3 175B while maintaining performance, demonstrating that most adaptation can occur in low-dimensional subspaces.

The subsequent development of QLoRA (Dettmers et al., 2023) combined 4-bit quantization with LoRA. This innovation enabled 65B parameter model fine-tuning on single 48GB GPUs with preserved 16-bit performance, dramatically improving access to large model adaptation. The approach uses double quantization for additional memory savings and paged optimizers to handle memory spikes during training.

Recent advances in parameter-efficient methods have expanded beyond LoRA to include diverse approaches. AdaLoRA (Zhang et al., 2023) adapts the rank allocation across weight matrices during training, while DoRA (Liu et al., 2024) decomposes pre-trained weights into magnitude and direction components for more effective adaptation.

2.2.3 Alternative Training Objectives and Architectural Innovations

Research into alternative pre-training objectives has revealed that traditional masked language modelling may not represent the optimal learning paradigm. Yamaguchi et al. (2021) demonstrated that token replacement detection, random vs. shuffled prediction, and frequency-based classification achieve comparable performance to MLM while running significantly faster. These simplified objectives particularly benefit smaller models, with BERT-MEDIUM showing minimal performance degradation despite reduced computational requirements.

Cheng et al. (2024) showed that instruction pre-training has emerged as a powerful alternative to traditional unsupervised pre-training. Using 200M synthetic instruction-response pairs covering 40+ task categories, models can achieve comparable performance to much larger counterparts, demonstrating superior sample efficiency and faster convergence.

The success of contrastive learning in computer vision has inspired analogous approaches for language models. SimCSE (Gao et al., 2021) demonstrated breakthrough potential by using dropout as minimal data augmentation, achieving 76.3% unsupervised and 81.6% supervised Spearman correlation on semantic textual similarity tasks.

DiffCSE (Chuang et al., 2022) extended contrastive learning through approaches sensitive to semantic differences, achieving 2.3 absolute point improvements over SimCSE by contrasting original sentences with edited versions from stochastic masking and MLM sampling. This work demonstrated how contrastive methods can balance insensitivity to certain augmentations while remaining sensitive to meaningful modifications.

2.2.4 Architectural Efficiency and Hardware-Aware Design

Fundamental architectural innovations have demonstrated that moving beyond traditional transformer designs can yield substantial efficiency improvements. The Mamba architecture (Gu and Dao, 2024) represents a revolutionary departure from attention-based designs, achieving 5× higher throughput than transformers with linear complexity through selective state space models.

RetNet (Sun et al., 2024) introduces a triple paradigm architecture enabling parallel training, a constant inference complexity, and linear complexity for long sequences. The approach delivers 8.4× faster inference than transformers using Key-Vector cache, 70% memory reduction for 7B models with 8K sequences, and 25-50% memory savings during training. These innovations demonstrate that alternative sequence modelling approaches can provide substantial computational advantages without sacrificing capabilities.

Hardware-aware optimization has become increasingly important as model scales have grown. FlashAttention (Dao et al., 2022) optimizes memory hierarchy usage between HBM and SRAM through tiling strategies, achieving 15% speedup on BERT-large and 3× on GPT-2 while providing exact computation without approximation. FlashAttention-2 delivers additional 2× improvement through enhanced parallelization and specialized GPU kernels, demonstrating how algorithm-hardware co-design can unlock significant performance improvements.

Recent architectural trends have moved toward specialized designs optimized for specific efficiency requirements. Mixture-of-experts (MoE) architectures enable scaling to massive parameter counts while maintaining manageable computational costs through sparse activation. Models like GLaM (Du et al., 2022) and Switch Transformer (Fedus et al., 2022) demonstrate how selective parameter activation can provide the benefits of large models whilst maintaining a consistent computational cost, equivalent to smaller models.

2.2.5 Sample-Efficient Learning and Data Quality

Research into sample-efficient learning has revealed that data quality often matters more than quantity for language model training. Careful data curation, deduplication, and filtering can achieve superior results with dramatically reduced dataset sizes (Lee et al., 2022, Xie et al., 2023). The success of models trained on carefully curated datasets demonstrates that intelligent data selection can overcome many computational limitations (Soldaini et al., 2024).

Curriculum learning approaches attempt to improve sample efficiency through careful ordering of training examples (Xu et al., 2020). While results have been mixed (Campos, 2021, Wu et al., 2021), some approaches have shown benefits for specific domains and tasks (Ye et al., 2021). Multi-stage training procedures that begin with broad pre-training before specializing on domain-specific or task-specific data have demonstrated consistent improvements over single-stage approaches (Devlin et al., 2019, Raffel et al., 2020).

Data augmentation techniques for language models have evolved beyond simple word substitution to more sophisticated approaches. Back-translation, paraphrasing, and synthetic data generation through large language models have all shown promise for improving sample efficiency (Sennrich et al., 2016, Li et al., 2023). The key insight is that increasing effective dataset size through high-quality augmentation can be more beneficial than simply collecting more raw data (Xie et al., 2023).

2.3 Background: Headless Language Modelling

The Contrastive Weight Tying (CWT) pre-training objective introduced by Godey et al. (2024) represents a significant deviation from more classical language model training approaches such as Masked Language Modelling (Devlin et al., 2019) or Next Token Prediction (Radford et al., 2018). In these traditional language modelling objectives models are trained to predict the probability distributions over an entire vocabulary for masked or next token predictions.

Traditional language modelling objectives operate under a singular focus: maximizing predictive accuracy on next-token or masked-token objectives through minimizing cross-entropy loss. This approach has proven remarkably effective, but remains fundamentally oriented toward prediction rather than representation quality. Models trained with these standard objectives may develop internal representations that serve predictive purposes without capturing meaningful linguistic structure that generalises consistently (McCoy et al., 2020).

An improvement of the classical learning objectives is Weight Tying, which reworks the approach to training language models. Clark et al. (2020) proposed to share the weights between the input embedding and output projections, which are traditionally treated independently. The fundamental insight behind weight tying lies in the similarity between the input embeddings that map semantically similar tokens to nearby vectors, while output projections assign similar probability scores to semantically interchangeable words. In standard transformer architectures two separate matrices handle these operations, one matrix that converts token IDs to dense vectors, one matrix that converts dense vectors (the hidden layer) to token probabilities. Since both operations require understanding which words are semantically similar, it enables the replacement of both matrices with one shared matrix, as long as the embedding dimensions equal the hidden dimension size. The weight tying approach essentially halves the amount of embedding weights in the model and thus reduces the compute necessary, whilst simultaneously increasing model performance.

CWT follows the idea of Weight Tying and introduces an objective that explicitly enforces representational consistency between input embeddings and output projections. This consistency constraint encourages the model to develop more coherent and semantically structured internal representations that preserve meaningful relationships across different model layers (Godey et al., 2024). Godey's Contrastive Weight Tying is similar to Weight Tying, as it creates a connection between two states, but differs in that the projection head is removed for the CWT learning objective. The name Headless Language Model stems from this property. Traditionally, the projection head transforms the hidden state to a token, by predicting which token has the highest probability to be the final hidden state. This step is individually lightweight, but becomes computationally intensive when the token vocabulary grows, as the model must compute a dot product between the hidden state and every token embedding in the vocabulary to generate output logits. For modern language models with vocabularies of 30K-100K tokens, this linear scaling relationship means the projection step can dominate computational costs, especially for long sequences. Removing this head gets rid of this costly last step and therefore has the potential of significantly speeding up the process of pre-training. In addition to providing probabilities, the projection head's softmax function serves as a regularisation mechanism that prevents weight explosion. Godey et al. (2024) propose in-batch negative sampling to carry out the regularisation, as it naturally embeds the required regularisation within the loss function.

A critical aspect of CWT, the in-batch negative sampling, is its implementation of a contrastive framework that treats the correct embedding-projection pairs as positive examples while considering all other potential pairings as negative examples. This approach shares conceptual similarities with other contrastive learning methods that have proven successful in computer vision and more recently in language understanding tasks. However, unlike methods such as SimCLR (Chen et al., 2020) or CLIP (Radford et al., 2021) that require explicit construction of positive and negative pairs, CWT leverages the inherent structure of language models to define these contrasts organically.

Intuitively this feels not too different from Word2Vec by Mikolov et al. (2013) as explained by Goldberg and Levy (2014) with the only difference the context words are the in-batch negatives examples. Word2Vec's negative sampling mechanism maximizes the similarity between observed word-context pairs while minimizing similarity with randomly sampled "negative" words drawn from a noise distribution. CWT employs a parallel contrastive framework but replaces Word2Vec's external negative sampling with in-batch negative sampling, where all other embeddings within the same training batch serve as negative examples for each target embedding. Given the examples in Figure 2.1, the masked-token's embedding becomes less similar to the other masked-token's embeddings in the same batch, moving the embeddings of *Tigers*, *cars* and *look* away from each other.

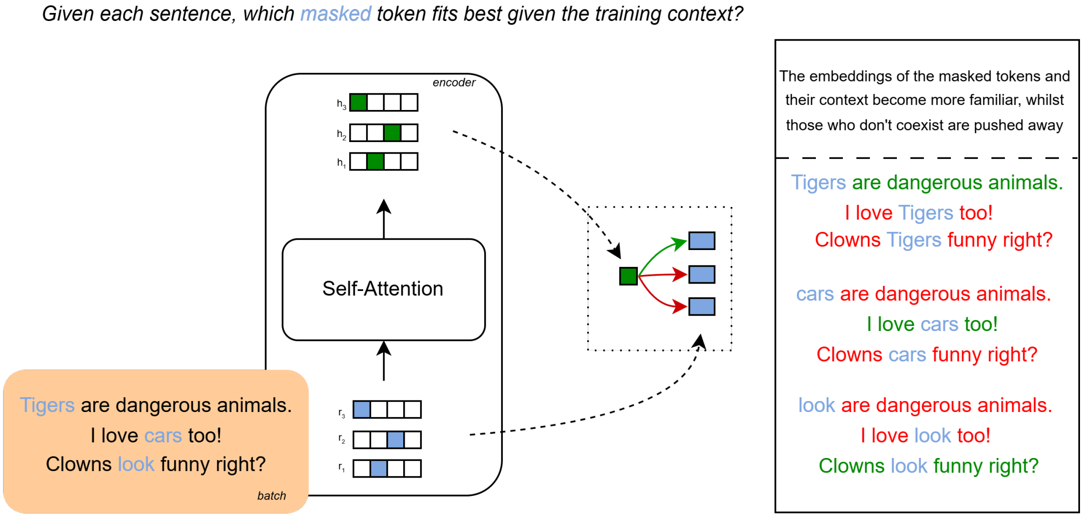
Given each sentence, which masked token fits best given the training context?
encoder
h₃
h₂
h₁
Self-Attention
r₃
r₂
r₁
Tigers are dangerous animals.
I love cars too!
Clowns look funny right?
batch
The embeddings of the masked tokens and their context become more familiar, whilst those who don't coexist are pushed away
Tigers are dangerous animals.
I love Tigers too!
Clowns Tigers funny right?
cars are dangerous animals.
I love cars too!
Clowns cars funny right?
look are dangerous animals.
I love look too!
Clowns look funny right?

Figure 2.1: Masked Headless Language Modelling use of in-batch negatives, based on Figure 1 in Godey's Arxiv Preprint (Godey et al., 2023) available here: `https://arxiv.org/pdf/2309.08351`

2.4 Background: BabyLM Shared Task

The BabyLM Challenge, introduced in 2023, represents a significant initiative in language model research that focusses on sample-efficient pre-training using developmentally plausible corpora. This shared task created a deliberate shift from the prevailing trend in language model development—where success has been predominantly associated with massive datasets—to a more constrained and cognitively relevant approach to model training.

2.4.1 Dataset Description

The BabyLM dataset was curated to reflect two key properties inspired by human language acquisition: a limited size deliberately constrained to under 100M words, with the STRICT track having approximately 98M words and the STRICT-SMALL track having approximately 10M words, and a modality representation recognizing that children primarily experience language through speech rather than written text. The dataset draws from ten diverse sources spanning multiple domains, including child-directed speech from CHILDES (5% of the corpus), British National Corpus dialogue portion (8%), children's books from the Children's Book Test and Children's Stories Text Corpus (9% combined), standard written English from Project Gutenberg (10%), movie subtitles from OpenSubtitles (31%), educational video subtitles from QCRI Educational Domain Corpus (11%), Wikipedia and Simple Wikipedia (25% combined), and dialogue from the Switchboard Dialog Act Corpus (1%). The STRICT-SMALL dataset was created as an approximately 10% uniform subsample of the STRICT dataset, maintaining the same proportional distribution across these sources. The distribution between transcribed speech and written texts is almost evenly distributed (56% vs. 44%), differentiating from traditional Language Model training data. This combined with a high percentage of child-directed speech (40%) creates an interesting dataset which differentiates itself not merely in its limited size, but also in its composition (Warstadt et al., 2023a).

The 10M word STRICT-SMALL dataset enables faster iteration on architectures and hyper-parameters, optimising training pipelines before scaling to larger datasets. Moreover, it lowers the barrier to entry for language model research, empowering academic and independent researchers to explore foundational questions in language learning without the need for highly funded industry resources (Warstadt et al., 2023a). Warstadt et al. (2023a) claim the STRICT-SMALL dataset closely approximates the linguistic exposure of a child during formative language acquisition years. The presence of child-directed speech in the STRICT-SMALL dataset may limit its generalisability, as half of the input is simplified and stylistically distinct from adult-directed language. This could impact experiments by skewing model behaviour toward patterns not representative of broader language use, potentially reducing performance on tasks requiring more complex or diverse input.

The STRICT track of the BabyLM Challenge presents a interesting intermediate case between extremely limited data (STRICT-SMALL) and the massive datasets common in modern LLM development. With approximately 98M words, this track represents roughly the amount of linguistic input a child might encounter by the age of 12, according to Gilkerson et al. (2017)'s findings on language exposure. What makes the STRICT track particularly interesting is that it offers a balance between cognitive plausibility and model capacity. At this scale, more complex architectural innovations become viable while still maintaining strong constraints that prevent simply scaling up as a solution. The 98M word dataset allows researchers to investigate how performance improves with an order of magnitude more data than the STRICT-SMALL track, potentially revealing important scaling laws and efficiency thresholds in language acquisition.

Chapter 3

Methodology

This research investigates the comparative performance of headless language models against their vanilla counterparts across specialized evaluation tracks. The methodology involves training both headless and conventional architectures of Masked Language Models (MLMs) and Generative Pre-trained Transformers (GPTs) on the STRICT track, with an additional exploration of the STRICT track by the MLM architecture. MLMs, such as BERT, are trained to predict masked tokens within a sequence, encouraging deep bidirectional understanding of context. In contrast, GPT-style models are trained to predict the next token in a sequence, learning language through autoregressive generation. This chapter explains the comprehensive experimental procedures, training protocols, evaluation metrics, and computational infrastructure employed in this study.

3.1 Model Architectures

3.1.1 Headless Model Configurations

Two headless variants are implemented in this study, featuring modified architectures in which the traditional prediction head is removed. For the MLM configuration, we remove the standard token prediction layer while maintaining the core transformer encoder blocks. Similarly, for the GPT architecture, the final linear projection layer that typically maps hidden states to vocabulary-sized logits is eliminated, creating a model that is not optimised for language generation. This is solved by adding a prediction head and fine-tuning it, but only when using the model for generation itself. Since this is done after pre-training, the advantages of the headless setup still hold. The headless MLM configuration maintains the bidirectional attention characteristic of BERT-like models, while the headless GPT preserves the autoregressive, left-to-right generation pattern fundamental to generative models.

3.1.2 Vanilla Model Configurations

For comparative purposes, we implement conventional “vanilla” versions of both model families. The vanilla MLM follows the standard BERT-base architecture with a complete prediction head, while the vanilla GPT adheres to the traditional GPT architecture with its parameters based on the 70M parameter Pythia model by EleutherAI (Biderman et al., 2023), in order to stay close to Godey et al. (2024). These models serve as baselines against which the performance of the headless variants is evaluated.

3.2 Tokenizer

The BERT-base-uncased tokenizer was employed consistently across all experiments to maintain methodological uniformity. This tokenizer, which processes text with case insensitivity and contains a vocabulary of 30,522 tokens, offers a suitable compromise between computational requirements and representational capabilities for our language tasks. By standardising the tokenisation approach between all models, we eliminated potential variability in tokenisation that could affect experimental outcomes. This choice also aligns with common practices in the field, allowing for meaningful comparison with existing research and positioning our results within the current body of knowledge on transformer-based language models.

3.3 Training Procedure

In this experiment, four distinct model configurations were evaluated to investigate the impact of different training setups on masked language model performance. Two configurations utilized a batch size of 12: the Headless Masked Language Model (HMLM) following Godey et al. (2024)'s training objective trained on a single GPU, and the Vanilla Masked Language Model (VMLM) with the traditional pretraining objective trained across two GPUs. The other two configurations employed a batch size of 32: the HMLM variant trained on two GPUs, and the Vanilla MLM distributed across four GPUs for more efficient computation. All four models were structurally based on the google-bert/bert-base-uncased architecture: 110M parameters, 768 hidden dimensions, 12 layers and 12 attention heads, with the critical distinction being the specialized headless training objective implemented in the HMLM variants as proposed by Godey et al. (2024), which removes the traditional prediction head while maintaining effective representation learning capabilities.

3.4 Evaluation Benchmarks: GLUE & BLiMP

In order to evaluate the trained models comprehensively, I look at Godey et al. (2024) and Warstadt et al. (2023a). Godey et al. (2024) uses GLUE as a means to score his models, whilst Warstadt et al. (2023a) utilises GLUE and BLiMP. GLUE and BLiMP offer complementary measures of broad language comprehension abilities and detailed grammatical knowledge.

The General Language Understanding Evaluation (GLUE) benchmark has become a foundational metric to evaluate Natural Language Understanding (NLU) capabilities in AI systems since its introduction (Wang et al., 2018). GLUE comprises nine diverse tasks: CoLA judges whether sentences are deemed grammatical English. SST-2 performs binary sentiment classification on movie reviews. MRPC determines if two sentences are semantically similar. QQP investigates semantic similarity between question pairs. STS-B measures similarity between two sentences on a continuous scale. MNLI investigates the model's ability to predict whether a premise entails, contradicts or is unrelated to a hypothesis. QNLI investigates how well the model is at retrieving answers to questions from context sentences. RTE contains sentences that investigate whether one sentence entails another. Each task evaluates different aspects of language comprehension. While GLUE revolutionised model evaluation by providing a standardised framework, it eventually suffered from ceiling effects as models like BERT and its successors began achieving near-human performance (Wang et al., 2019).

In contrast, the Benchmark of Linguistic Minimal Pairs (BLiMP), introduced in 2020, takes a fundamentally different approach by focussing on targeted linguistic phenomena through minimal pairs of (un)grammatical sentence constructions that differ by just one element (Warstadt et al., 2020). BLiMP's 12 categories evaluate anaphor agreement, argument structure, binding, control, determiner-noun agreement, ellipsis, filler-gap, irregular forms, island effects, NPI licensing, quantifiers and subject-verb agreement without requiring task-specific fine-tuning, instead measuring a model's raw linguistic knowledge through forced-choice accuracy.

Both GLUE and BLiMP are used as benchmarks. While GLUE primarily assesses task performance on downstream applications, BLiMP provides deeper insights into a model's underlying grammatical competence and linguistic representations, making these complementary benchmarks essential tools for understanding the strengths and limitations of language models across different dimensions of language understanding. GLUE requires prior fine-tuning on every downstream task, where BLiMP is a form of zero-shot evaluation, requiring no additional fine-tuning or task-specific training.

Chapter 4

Results

This chapter presents a comprehensive empirical comparison between headless and vanilla masked language models (MLM) and generative pre-trained transformers (GPT) across two datasets of varying sizes: STRICT-small and STRICT. We start with the results of the MLM on the STRICT-SMALL dataset, after which we move to the STRICT dataset on which we report the results for both the MLM and GPT architectures.

4.1 STRICT-small

4.1.1 Headless- vs Vanilla MLM: GLUE

We evaluated both headless and vanilla masked language model (MLM) architectures on the STRICT-small dataset using the GLUE benchmark validation sets. All experiments were conducted with a batch size of 12, and results represent averages across 3 random seeds to ensure statistical reliability.

Overall Performance Trends

Both architectures demonstrated consistent improvement in performance as training progressed from 100 million to 600 million tokens, where the number of tokens indicates the total amount of text the model was exposed to during pre-training. Since our dataset contained 10 million tokens, reaching 600 million tokens requires 60 complete passes through the dataset (epochs), with each training step processing a batch of tokens until the full dataset had been seen multiple times. The headless MLM achieves an average performance of 50.0% after 100M tokens, steadily increasing to 58.8% after 600M tokens (Table 4.1). Similarly, the vanilla MLM shows an improvement from 49.8% at 100M tokens to 59.9% after 600M tokens (Table 4.2), ultimately outperforming the headless variant by 1.1 percentage points. When looking at the task specific results, we see a pattern that follows the results of the first BabyLM task (Warstadt et al., 2023b). In which the COLA, STS-B and RTE tasks consistently receive a lower score than SST-2, MRPC, QQP, MNLI and QNLI.

Task-Specific Performance Analysis

Examination of individual GLUE tasks reveals distinct patterns in how the two architectures handle different linguistic phenomena. On sentiment analysis (SST-2), both architectures demonstrate strong performance with consistent improvement across

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE	Avg.
100M	**0**	78.4	**69.6**/57.1	9.1/9.5	**77.8**/74.3	**61.4**	60.2	52.5	50.0
200M	**5.0**	78.5	67.9/**59.8**	27.4/26.3	**80.9/78.6**	**61.5**	**63.9**	51.4	54.7
400M	**8.1**	80.0	69.1/**60.4**	37.3/36.5	81.7/78.9	**63.6**	65.3	50.2	57.4
600M	7.5	**80.7**	**70.7**/63.7	41.1/40.1	81.5/78.7	63.8	**67.1**	51.5	58.8

Table 4.1: Results of the Headless Masked Language Models on the validation sets of the GLUE benchmark. Matthews correlation coefficient is reported for COLA, Pearson and Spearman correlation for STS-B, accuracy and F1-score for all other tasks. Results represent averages across 3 random seeds. The scores shown in bold fall within the standard deviation range of the Vanilla variant's performance on the same task.

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE	Avg.
100M	**0**	76.4	70.6/**55.6**	13.1/14.1	75.7/69.4	**61.1**	61.2	**51.3**	49.8
200M	1.6	**79.5**	70.5/**57.6**	14.9/16.1	79.7/77.3	**61.4**	65.0	54.1	52.5
400M	**8.5**	80.6	62.9/**60.2**	41.1/40.3	81.0/77.8	**63.7**	66.1	53.1	57.8
600M	**9.8**	**80.8**	71.9/**62.1**	44.4/43.5	82.2/79.4	64.7	**66.8**	**53.1**	59.9

Table 4.2: Results of the vanilla Masked Language Models, trained on the STRICT-SMALL dataset, on the validation sets of the GLUE benchmark. Matthews correlation coefficient is reported for COLA, Pearson and Spearman correlation for STS-B, accuracy and F1-score for all other tasks. Results represent averages across 3 random seeds. The scores shown in bold fall within the standard deviation range of the Headless variant's performance on the same task

training, ultimately achieving nearly identical results with the vanilla MLM reaching 80.7% compared to the headless MLM's 80.6% at 600M tokens. This suggests both architectures effectively learn sentiment-related features with similar efficiency. Natural language inference tasks (MNLI) shows comparable performance between the two approaches, with both models reaching approximately 64% accuracy at the final checkpoint. The vanilla MLM maintaines a slight advantage throughout training, achieving 64.7% versus the headless MLM's 63.8% at 600M tokens. For paraphrase detection (MRPC), the results reveal interesting architectural differences in how precision and recall are balanced. While the vanilla MLM achieve superior accuracy (71.9% versus 70.7% at 600M tokens), the headless MLM demonstrates better F1-score performance (63.7% versus 62.1%), suggesting the two architectures make different precision-recall trade-offs when identifying paraphrases. The semantic textual similarity task (STS-B) shows substantial improvement over training steps for both architectures, with the vanilla MLM achieving superior performance in both correlation measures. At 600M tokens, the vanilla MLM reaches 44.4% Pearson correlation and 43.5% Spearman correlation, compared to the headless MLM's 41.1% and 40.1% respectively. Question answering performance (QNLI) was remarkably similar between architectures, with both achieving approximately 67% accuracy at the final checkpoint. Linguistic acceptability judgments (COLA) proved challenging for both approaches, though the vanilla

MLM shows marginally better performance with a Matthews correlation coefficient of 9.7% compared to the headless MLM's 7.4% at 600M tokens. Both vanilla and headless MLM architectures demonstrate comparable performance across GLUE tasks, with the vanilla model showing slight advantages in most tasks (particularly STS-B and COLA) while the headless model achieves better F1-scores in paraphrase detection, suggesting minimal but consistent architectural differences in handling various linguistic phenomena.

Convergence Patterns

The results indicate that both architectures benefit from extended training, with most tasks showing continued improvement from 400M to 600M tokens. However, some tasks exhibited plateauing behaviour, particularly RTE, where performance remained relatively stable across later checkpoints for both architectures. The vanilla MLM architecture demonstrated more consistent improvement across checkpoints, achieving better final performance on 6 out of 8 GLUE tasks, while the Headless MLM shows superior performance on MRPC accuracy and comparable results on SST-2.

The evaluation results presented reflect averaged performance across three random seeds to ensure statistical reliability, with some individual scores naturally affected by this averaging process. In the accompanying tables, scores displayed in bold indicate instances where a model's performance falls within one standard deviation (average ± standard deviation) of its corresponding counterpart model's performance range. This formatting helps identify cases where apparent performance differences between models may be affected by underperforming seeds, providing a more nuanced interpretation of the comparative results between vanilla MLM and headless MLM architectures. An example of such an instance is the COLA score for the Headless 600M checkpoint, where the standard deviation is 2.8, indicating a big discrepancy between evaluation runs. All standard deviation scores are reported in Appendix A.

4.1.2 Headless- vs Vanilla MLM: BLiMP

The BLiMP (Benchmark of Linguistic Minimal Pairs) evaluation provides crucial insights into the grammatical competence and linguistic knowledge acquired by both model architectures during pre-training. Unlike GLUE tasks that require task-specific fine-tuning, BLiMP measures raw linguistic understanding without prior fine-tuning. All experiments were conducted with a batch size of 12 and 600 million token checkpoint, as it performed best on the GLUE benchmark.

Overall Performance Comparison

The vanilla MLM demonstrated superior overall performance on the BLiMP benchmark, achieving 53.67% accuracy compared to the headless MLM at 50.24%—a difference of 3.43 percentage points. This represents the most substantial performance gap observed between the two architectures across all evaluation metrics, suggesting that the traditional prediction head architecture provides advantages for certain aspects of grammatical knowledge acquisition.

Interestingly, both architectures performed just above random chance (50%) on the aggregate BLiMP filtered scores, indicating that both training objectives are unable to induce meaningful grammatical representations. Performance on the BLiMP supplement tasks shows the same pattern, with the headless model achieving 51.72% compared to the vanilla model's 50.60%.

Detailed Linguistic Phenomena Analysis

This analysis presents preliminary findings from a comparative evaluation of headless and vanilla architectures, examining their performance across various tasks. While the results offer initial insights into potential architectural differences, it is important to emphasize that these observations require more extensive research to verify their robustness and generalizability.

Agreement Phenomena. The results reveal striking differences in how the two architectures handle various agreement tasks. For anaphor agreement, the vanilla model significantly outperformed the headless model, achieving 68.80% accuracy on gender agreement compared to 44.90% for the headless model, and 81.20% versus 48.23% on number agreement. This substantial gap suggests that the traditional prediction head architecture may be particularly beneficial for learning pronoun-antecedent relationships and the associated morphological constraints.

Subject-verb agreement patterns shows a more nuanced differences. For regular plural subject-verb agreement, both models performed similarly (vanilla: 49.21% and 53.86%; headless: 49.44% and 52.59% across the two variants), while irregular plural agreement also shows less comparable performance between architectures (vanilla: 51.74% & 54.93%; headless: 48.88% & 46.97%). This suggests that irregular plural subject verb agreement is more difficult for the headless training approach.

Syntactic Island Effects and Constraints. The models show interesting patterns on syntactic island violations and extraction constraints. For complex NP islands, the vanilla model achieves 54.37% accuracy compared to the headless model's 47.04%, indicating better sensitivity to structural constraints on long-distance dependencies. However, some extraction phenomena show different patterns. On wh-island effects, the headless model outperformed the vanilla model (69.37% vs 72.92%), suggesting that the headless architecture better captures the structural limitations on question formation and extraction. Left-branch island effects reveal an interesting reversal, with the vanilla model achieving 58.82% on echo questions compared to the headless model's 40.65%, while simple questions show the opposite pattern (vanilla: 49.21% vs headless: 51.21%).

Ellipsis and Argument Structure. The two architectures show dramatically different performance on ellipsis phenomena. For N-bar ellipsis, the vanilla model achieves exceptional performance on the second variant (86.23% accuracy) compared to the headless model's 45.89%, the first variant followed this pattern (vanilla: 57.98% vs headless: 43.14%).

Argument structure phenomena, including causative constructions and transitivity alternations, show mixed patterns. The headless model performed better on drop argument constructions (52.50% vs 45.33%), and at inchoative constructions (56.49% vs 42.34%).

Negative Polarity Items (NPIs) and Quantification. The treatment of negative polarity items reveal some of the most dramatic performance differences between architectures. The vanilla model achieves near-perfect performance on several NPI licensing tasks: 97.17% accuracy on sentential negation NPI licensor presence and 99.56% on Principle A case licensing. In contrast, the headless model achieves 61.04%

and 71.93% respectively on these tasks, representing some of the largest performance gaps observed.

However, the pattern was not consistently in favour of the vanilla model. For "only" NPI licensing, the vanilla model achieves 84.01% compared to the headless model's 42.74%, but NPI scope relationships show different patterns, with both models performing around 42% accuracy.

Morphological and Lexical Phenomena. Irregular morphological forms presented interesting challenges for both architectures. For irregular past participles used as adjectives, the vanilla model performed slightly better (49.53% vs 43.60%), while irregular past participles in verbal contexts had a bigger difference (63.27% vs 40.76%).

Implications for Linguistic Competence The BLiMP results provide several important insights into the linguistic competence developed by headless versus vanilla training approaches. The vanilla model's superior performance on anaphor agreement, NPI licensing, and certain syntactic island effects suggests that the traditional prediction head architecture may be particularly beneficial for learning complex long-distance dependencies and licensing relationships that require precise grammatical computation.

Conversely, the headless model's competitive or superior performance on certain phenomena (left-branch island echo questions, drop arguments, some morphological tasks) indicates that the contrastive weight tying objective successfully captures important aspects of grammatical structure, even without explicit vocabulary prediction training.

It is crucial to acknowledge that both models achieve relatively low absolute performance scores across most BLiMP tasks, with many individual task accuracies hovering close to random chance (50%). The majority of specific linguistic phenomena show performance between 40% and 60%, with only a few notable exceptions achieving substantially higher accuracy. This generally poor performance suggests that both architectures struggled to acquire robust grammatical competence under the constrained training conditions of the STRICT-small dataset (10M words). These low scores indicate that the observed differences between architectures, while statistically meaningful, may not be practically significant for real-world applications requiring reliable grammatical knowledge. The limited training data appears insufficient for either model to develop strong linguistic representations, making the comparative analysis more indicative of relative learning efficiency rather than absolute grammatical competence.

4.1.3 Training Efficiency Analysis

A notable advantage of the headless MLM architecture emerges in computational efficiency during training. The headless model completed its full 600M token training regimen in 3 hours and 15 minutes, while the vanilla MLM required 4 hours and 45 minutes to reach the same checkpoint, representing a 32% decrease in training time for the headless architecture. The fine-tuning runs for the GLUE benchmark saw a similar speed up, 16 hours for the headless models vs 24 hours for the vanilla models. This efficiency difference is particularly striking when considering that by the time the headless MLM had completed its entire training run, the vanilla MLM had only reached 400M tokens. At this intermediate checkpoint, the headless model (58.5% average performance at 600M tokens) had already surpassed the vanilla model's performance at the 400M token mark (57.8% average performance), suggesting that the headless architecture not only trains faster but also achieves superior performance in equivalent wall-clock time. Additionally, the headless architecture enables training on less powerful devices. During the experimental phase a headless MLM could be trained on the STRICT-SMALL dataset with a batch size of 16 on my personal computer, whilst this was not possible for the vanilla counterpart. My personal computer is equipped with an RTX3060 GPU containing 12GB of memory, while the ADA cluster used for all experiments utilizes a Nvidia A30 GPUs with 24GB of memory.

Architecture	Pre-training Time	Performance after 3h 15min (GLUE)
Headless MLM	3h 15m	58.5%
Vanilla MLM	4h 45m	57.8%
Efficiency Gain Headless	32% faster	0.7 increase

Table 4.3: Training efficiency comparison between headless and vanilla MLM architectures for 600M pre-training tokens

4.2 STRICT

4.2.1 Headless- vs Vanilla MLM: GLUE

We conducted a comprehensive evaluation of both headless and vanilla masked language model architectures trained on varying amounts of the STRICT dataset to assess performance scaling with increased training data. Models were trained on 1 billion and 2 billion tokens respectively, with all experiments maintaining consistent hyper-parameters and evaluation protocols across the GLUE benchmark tasks. Results represent averages across 3 random seeds to ensure statistical robustness.

Overall Performance Trends

Both architectures demonstrated substantial performance improvements when scaling from 1 billion to 2 billion training tokens. The headless MLM shows a remarkable improvement from 59.63% average performance at 1B tokens to 70.98% at 2B tokens, representing an 11.35 percentage point gain, as can bee seen in Table 4.4. The vanilla MLM exhibited similar scaling behaviour, improving from 62.84% at 1B tokens to 71.51% at 2B tokens, with a gain of 8.67 percentage points, as can bee seen in Table 4.5. Particularly, while the vanilla architecture maintained a consistent advantage at both scales, the performance gap narrowed from 3.21 percentage points at 1B tokens to just 0.53 percentage points at 2B tokens, suggesting that the headless architecture benefits more substantially from increased training data. Just as for the STRICT-SMALL models our task specific results follow the pattern in the results of the first BabyLM task (Warstadt et al., 2023b). In which the COLA, STS-B and RTE tasks consistently receive a lower score than SST-2, MRPC, QQP, MNLI and QNLI.

Task-Specific Performance Analysis

The scaling analysis reveals distinct patterns across different linguistic tasks and architectural choices. For sentiment analysis (SST-2), both architectures show strong performance improvements with increased training data, with the vanilla MLM achieving superior results at both scales (79.69% vs 81.73% at 1B tokens, and 86.20% vs

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE	Avg.
1B	5,29	81,73	**68,72/61,71**	42,07/41,52	82,88/80,69	**67,41**	70,53	**53,33**	59,63
2B	**22,09**	84,56	72,12/58,32	**77,27/77,28**	88,11/86,73	74,99	83,22	56,08	70,98

Table 4.4: Results of the Headless Masked Language Models, trained on the STRICT dataset, on the validation sets of the GLUE benchmark. Matthews correlation coefficient is reported for COLA, Pearson and Spearman correlation for STS-B, accuracy and F1-score for all other tasks. Results represent averages across 3 random seeds. The scores shown in bold fall within the standard deviation range of the Vanilla variant's performance on the same task.

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE	Avg.
1B	**8,42**	79,69	**68,43/61,75**	54,26/53,22	84,85/83,14	67,91	75,92	**53,62**	62,84
2B	**20,99**	86,20	75,00/68,08	**76,61/77,22**	86,91/85,48	73,65	82,57	53,92	71,51

Table 4.5: Results of the Vanilla Masked Language Models, trained on the STRICT dataset, on the validation sets of the GLUE benchmark. Matthews correlation coefficient is reported for COLA, Pearson and Spearman correlation for STS-B, accuracy and F1-score for all other tasks. Results represent averages across 3 random seeds. The scores shown in bold fall within the standard deviation range of the Headless variant's performance on the same task.

84.56% at 2B tokens). Interestingly, the headless architecture's performance on this task plateaued or slightly decreased at 2B tokens, while the vanilla model continued to improve substantially.

Linguistic acceptability judgments (COLA) proved to be the most challenging task for both architectures at 1B tokens, but show dramatic improvements with additional pre-training. The headless MLM demonstrated exceptional scaling on this task, improving from 5.29% to 22.09% Matthews correlation coefficient, while the vanilla MLM show similar gains from 8.42% to 20.99%. Both architectures achieve statistically equivalent performance on COLA at both scales, as indicated by the bold formatting.

Natural language inference tasks reveal interesting architectural differences. For MNLI, both models show similar performance at 1B tokens (vanilla: 67.91% vs headless: 67.41%), but the headless architecture shows better scaling properties, achieving 74.99% compared to the vanilla model's 73.65% at 2B tokens. Question answering performance (QNLI) followed a similar pattern, with both architectures showing substantial improvements between the 1B- and 2B token checkpoints. Even though the headless model's performance was worse after 1B tokens (headless: 70.53% vs vanilla: 75.92%), it managed to get a slight advantage at the larger scale (headless: 83.22% vs vanilla: 82.57%).

The semantic textual similarity task (STS-B) exhibited the most dramatic scaling effects for both architectures. At 1B tokens, performance was moderate for both models (headless: 42.07 & 41.52 vs vanilla: 54.26 & 53.22), but at 2B tokens, both architectures achiev remarkable improvements, with correlation coefficients exceeding 76 for both Pearson and Spearman measures. The headless model marginally outperformed the vanilla model (headless: 77,27& 77,28 vs vanilla: 76,61& 77,22).

Paraphrase detection (MRPC) shows interesting precision-recall trade-offs that persisted across scales. At 1B tokens, both architectures achieve statistically equivalent performance on both accuracy and F1-score metrics (68% accuracy and 61% F1). However, at 2B tokens, the vanilla MLM demonstrated superior performance on both measures (vanilla: 75.00%/68.08% vs headless: 72.12%/58.32%), suggesting different scaling behaviours in how the architectures balance precision and recall for this task. Most notable is the consistent collapse of the F1 score for the headless model, as indicated by the standard deviation.

A notable exception to the general scaling trend emerges with the Recognising Textual Entailment (RTE) task, where both architectures show minimal improvement despite doubling the training data. The headless MLM improved marginally from 53.33% to 56.08%, while the vanilla MLM remained essentially static at 53.62% and 53.92% respectively. This plateau suggests that RTE may require architectural modifications or different training strategies beyond simply increasing data scale, as both models appear to have reached a performance ceiling on this textual entailment task.

Both vanilla and headless MLM architectures demonstrate comparable performance across GLUE tasks, with the headless model showing slight advantages in most tasks.

Convergence Patterns

Statistical analysis of the results reveals that many performance differences between architectures fall within overlapping confidence intervals, as indicated by the bolded scores in the tables. At 1B tokens, several tasks show equivalent performance between architectures, including MRPC accuracy/F1-score and RTE. However, at 2B tokens, the architectures show more differentiated performance profiles per specific task, whilst average performance over all task was comparable. The headless version averaged 70.98% while the vanilla version averaged 71.51%.

The convergence of average performance at 2B tokens, despite different scaling trajectories, suggests that both architectures reach similar representational capacity limits with increased training data. However, the task-specific performance patterns indicate that architectural choices continue to influence how effectively different linguistic phenomena are captured, even at scale.

4.2.2 Headless- vs Vanilla MLM: BLiMP

All experiments were conducted with a batch size of 12 and 2 billion token checkpoint, as it performed best on the GLUE benchmark.

Overall Performance Summary

The vanilla MLM demonstrated significantly superior overall performance on the BLiMP benchmark, achieving 54.84% accuracy compared to the headless MLM at 49.20% -a difference of 5.64 percentage points on the filtered tasks. This performance gap is even more pronounced on the BLiMP supplement tasks, where the vanilla model achieve 61.60% compared to the headless model's 50.93%, representing a substantial 10.67 percentage point advantage. Unlike the models trained on the STRICT-SMALL dataset, both headless and vanilla architectures demonstrate meaningfully above-random performance on many individual tasks, with several achieving accuracies above 70% and some exceeding 90%. This suggests that the increased model size and training data (2B tokens vs 600M tokens) enabled both architectures to develop more robust grammatical competence.

Detailed Linguistic Phenomena Analysis

Anaphor Agreement The vanilla model substantially outperformed the headless model on both gender and number agreement: achieving 78.78% accuracy on gender agreement compared to 59.94% for the headless model, and 84.21% versus 56.18% on number agreement. This massive gap confirms that the traditional prediction head architecture provides significant advantages for learning pronoun-antecedent relationships.

Subject-Verb Agreement The pattern was mixed but generally favoured the vanilla model. For regular plural subject-verb agreement, the vanilla model shows contrasting performance across variants (44.72% vs 82.01%), while the headless model performed more consistently (52.81% vs 48.47%). For irregular plural agreement, the vanilla model achieve strong performance on the second variant (79.04% vs 51.35% for headless), indicating better handling of morphologically complex agreement patterns.

Syntactic Islands On the Complex NP Islands, interestingly, the headless model slightly outperformed the vanilla model (50.24% vs 42.20%). Where at the Wh-Island Effects the headless model maintained its advantage (55.42% vs 46.98%). The Left-Branch Islands shows a reversal in this pattern: the vanilla model dramatically outperformed on simple questions (58.46% vs 43.64%), while the headless model was superior on echo questions (44.35% vs 24.82%), suggesting different strategies for handling syntactic extraction.

Ellipsis and Argument Structure The vanilla model achieve exceptional performance on both ellipsis variants, with particularly remarkable results on the second variant 96.14% compared to the headless model's 33.09%, a massive 63.05 percentage point gap. The first variant also shows substantial vanilla superiority (71.57% vs 45.89%). Whilst argument structure shows a mixed pattern: the headless model performed slightly better on drop argument constructions (53.70% vs 49.13%) and causative constructions (44.01% vs 39.73%), while the vanilla model displays advantage on inchoative constructions (54.97% vs 49.12%).

Negative Polarity Items and Quantification The vanilla model achieve exceptional performance on core NPI licensing tasks: 94.23% accuracy on sentential negation NPI licensor presence compared to the headless model's 82.26%, and 98.79% on Principle A case licensing versus 87.61% Interestingly, the pattern reversed for "only" NPI tasks, where the headless model outperformed: achieving 26.19% vs 12.47% on licensor presence, and 71.92% vs 55.91% on scope relationships, suggesting different mechanisms for handling various types of NPI licensors.

Morphological and Lexical Phenomena The results show contrasting patterns: for irregular past participles used as adjectives, the headless model performed substantially better (55.15% vs 28.41%), while in verbal contexts, the vanilla model achieves superior performance (71.87% vs 52.44%). This suggests different strengths in handling morphological complexity across syntactic contexts.

Implications for Linguistic Competence

The BLiMP results from these 2 Billion token models provide several crucial insights. The 2B token vanilla model marginally outperformed the 600M vanilla model on total average, whilst the headless model performed slightly worse than its 600M token counterpart.

The vanilla model's substantial advantages on anaphor agreement, NPI licensing, ellipsis, and many morphological tasks confirm that the traditional prediction head architecture is better at complex grammatical dependencies and licensing relationships.

The headless model's competitive or superior performance on certain island effects and specific morphological tasks indicates that contrastive weight tying can successfully capture important structural aspects of grammar, particularly those involving syntactic constraints.

The consistent patterns across linguistic phenomena suggest that the observed differences reflect genuine architectural differences rather than random variation, making these comparative results significant.

4.2.3 Headless- vs Vanilla GPT: GLUE

The performance of both Headless and Vanilla Generative Pre-trained Transformers trained on the STRICT dataset was evaluated on the GLUE benchmark validation sets. Results are presented across two model versions, 1- and 2 billion tokens, and averaged over three random seeds.

Overall Performance

Both model variants demonstrated limited performance across the GLUE tasks, with average scores of 36.00 and 35.83 for the 1B and 2B Headless GPT models respectively (Table 4.6), and 37.94 and 36.23 for the corresponding Vanilla GPT variants (Table 4.7). The modest differences between model sizes and architectures suggest that scaling from 1B to 2B parameters provided minimal performance gains under the current training configuration.

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE	Avg.
1B	0,00	50,89	68,59/40,61	0/0	63,16/38,54	31,82	49,47	52,88	36,00
2B	0,00	50,89	68,59/40,61	0,03/2,01	63,16/38,54	31,82	49,47	49,04	35,83

Table 4.6: Results of the Headless Generative Pre-trained Transformers, trained on the STRICT dataset, on the validation sets of the GLUE benchmark. Matthews correlation coefficient is reported for COLA, Pearson and Spearman correlation for STS-B, accuracy and F1-score for all other tasks. Results represent averages across 3 random seeds.

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE	Avg.
1B	0,00	50,89	68,59/40,61	0,00/21,40	63,16/38,54	31,82	49,47	52,88	37,94
2B	0,00	50,89	68,59/40,61	0,03/6,34	63,16/38,54	31,82	49,47	49,04	36,23

Table 4.7: Results of the Vanilla Generative Pre-trained Transformers, trained on the STRICT dataset, on the validation sets of the GLUE benchmark. Matthews correlation coefficient is reported for COLA, Pearson and Spearman correlation for STS-B, accuracy and F1-score for all other tasks. Results represent averages across 3 random seeds.

Task-Specific Results

Performance varied considerably across individual tasks. Both model variants achieve zero performance on the CoLA task (Matthews correlation coefficient = 0.00), indicating an inability to capture linguistic acceptability judgments. On SST-2, performance remained at chance level (50.89% accuracy) across all configurations.

For tasks involving sentence pairs, results were mixed. On MRPC, both variants achieve 68.59% accuracy but substantially lower F1 scores 40.61%, suggesting difficulty with the precision-recall balance in paraphrase detection. The QQP task shows similar patterns with 63.16% accuracy but 38.54% F1 score.

The STS-B task reveal peculiar differences between model variants. The vanilla models show minimal correlation scores (Pearson: 0.00-0.03, Spearman: 6.34-21.40), while headless models performed comparably poorly (Pearson: 0.00-0.03, Spearman: 0-2.01).

Natural language inference tasks (MNLI, QNLI, RTE) yielded below-chance performance, with MNLI accuracy at 31.82%, QNLI at 49.47%, and RTE ranging from 49.04% to 52.88%.

Architecture Comparison

The comparison between Headless and Vanilla architectures reveal minimal systematic differences, with neither architecture demonstrated clear superiority across tasks. Strikingly, the scores show that both models failed to successfully carry out the assigned tasks. The absence of substantial performance gaps suggests that the architectural modifications did not significantly impact downstream task performance under the current experimental conditions.

Fine-tuning Dynamics

An analysis of the training progression uncovered troubling trends in model optimization. Regardless of architecture performance metrics show little improvement over fine-tuning epochs, pointing to potential issues with the learning dynamics of the 70M parameter GPT model on the STRICT dataset. This is in line with the findings of Steuer et al. (2023), who found a positive correlation between Language Model size and their performance on both the GLUE and BLiMP benchmark. The early plateau in validation performance suggests that models may be converging to suboptimal solutions, possibly due to limited data diversity, suboptimal learning rates, or architectural shortcomings in capturing the necessary patterns for GLUE task success. This stagnation during training likely contributes to the consistently poor performance observed across all model variants on downstream tasks.

4.2.4 Headless- vs Vanilla GPT: BLiMP

All experiments were conducted with a batch size of 12 and 2 billion token checkpoint.

Overall Performance Summary

The headless GPT model demonstrated slightly superior overall performance on the BLiMP benchmark, achieving 68.27% accuracy compared to the vanilla GPT model at 67.99% - a marginal difference of 0.28 percentage points on the filtered tasks. However, the vanilla model shows better performance on the BLiMP supplement tasks, achieving 63.14% compared to the headless model's 61.39%, representing a 1.75 percentage point advantage.

Detailed Linguistic Phenomena Analysis

Anaphor Agreement Contrary to previous patterns, the headless model slightly outperformed the vanilla model on both gender and number agreement: achieving 93.31% accuracy on gender agreement compared to 91.86% for the vanilla model, and 95.92% versus 93.34% on number agreement.

Subject-Verb Agreement For regular plural subject-verb agreement, performance varied significantly across variants: the headless model achieves 70.79% vs 81.27% across variants, while the vanilla model shows 81.35% vs 72.38%. For irregular plural agreement, both models performed strongly, with the headless model achieving 75.12% and 87.78% across variants compared to the vanilla model's 77.61% and 84.08%.

Syntactic Islands For the Complex NP Islands tasks the headless model marginally outperformed the vanilla model (49.17% vs 48.11%), maintaining competitive performance on structural constraint detection. On Wh-Island Effects the headless model continued to show better performance (47.92% vs 44.79%), indicating better sensitivity to question formation constraints. The Left-Branch Islands shows a striking pattern difference: while the vanilla model substantially outperformed on simple questions (45.43% vs 36.59%), and dramatically excelled on echo questions (58.71% vs 32.52%), suggesting that the vanilla architecture maintained advantages in certain types of syntactic extraction after extended training.

Ellipsis and Argument Structure The ellipsis results shows task-specific patterns: the vanilla model performs better on the first variant (64.21% vs 54.99%), while the headless model excelled on the second variant (91.79% vs 78.50%), indicating different strengths across ellipsis contexts.

Argument Structure shows mixed patterns: the headless and vanilla model perform similarly on drop argument constructions (73.59% vs 74.78%). Whilst inchoative constructions (headless: 49.47% vs vanilla: 57.31%) and causative constructions (vanilla: 65.77% vs headless: 57.21%) are dominated by the vanilla model.

Negative Polarity Items and Quantification Both models achieve exceptional performance on core NPI licensing tasks: the headless model reaches 96.63% accuracy on sentential negation NPI licensor presence compared to the vanilla model's 99.02%, and both achieve 100% and 80.66%/82.30% respectively on Principle A case licensing tasks. For "only" NPI tasks, the vanilla model shows modest advantages: achieving 42.74% vs 37.19% on licensor presence, and nearly identical performance on scope relationships (44.44% vs 44.56%).

Morphological and Lexical Phenomena The headless model demonstrates substantial advantages in morphological processing: for irregular past participles used as adjectives, the headless model performs significantly better (74.09% vs 61.39%), and also shows superior performance in verbal contexts (86.73% vs 82.06%).

Implications for Linguistic Competence The BLiMP results from these two GPT models provide several important insights into the effects of extended training on different architectures. The overall convergence in performance between the two architectures after 2 billion tokens (68.27% vs 67.99%) suggests that both approaches can achieve similar levels of grammatical competence with sufficient training, though they may develop different internal representations and processing strategies for specific linguistic phenomena.

4.2.5 Training Efficiency Analysis

The efficiency increase observed during pre-training on the STRICT-SMALL dataset was also present on the STRICT dataset. The headless MLM model completed its full 2 billion token training regimen in 9 hours and 52 minutes, while the vanilla MLM required 15 hours to reach the same checkpoint, representing a 34% decrease in headless training time (Table 4.8). The comparable performance on GLUE tasks after equivalent pre-training tokens is particularly noteworthy.

Architecture	Performance GLUE	Performance BLiMP	Pre-training Time
Headless	70.98%	49.20%	9h 52m
Vanilla	71.51%	54.84%	15h
Efficiency Gain Headless			34% faster

Table 4.8: Training efficiency comparison between headless and vanilla MLM architectures for 2 billion pre-training tokens

Whilst both headless MLM models saw a decrease in training time of around 34%, the headless GPT managed a decrease in training time of 53%. The headless GPT model completed its full 2 billion token training procedure in 7 hours, while the vanilla GPT required 14 hours and 40 minutes to reach the same number of pre-training tokens (Table 4.9). Contrary to all of the MLM models, both the headless and vanilla GPT models demonstrated terrible performance on the GLUE benchmark whilst simultaneously noting the highest BLiMP score's of the study.

Architecture	Performance GLUE	Performance BLiMP	Pre-training Time
Headless	35.83%	68.27%	7h
Vanilla	36.23%	67.99%	14h 40m
Efficiency Gain Headless			53% faster

Table 4.9: Training efficiency comparison between headless and vanilla GPT architectures for 2 billion pre-training tokens

Chapter 5

Discussion

The discussion section will evaluate the research hypotheses regarding computational efficiency and performance improvements, examining the trade-offs between headless and vanilla architectures and their scaling dynamics under different training conditions. It will explore the broader implications for democratizing language model research in resource-constrained environments, analyse task-specific performance patterns across different linguistic phenomena, and address key limitations including computational constraints and methodological considerations around data sufficiency.

5.1 Key Findings and Interpretation

This study investigated the effectiveness of headless language models using contrastive weight tying (CWT) compared to traditional vanilla architectures on the BabyLM challenge datasets. The results provide nuanced insights into the trade-offs between computational efficiency and model performance under data-constrained conditions.

5.1.1 Hypothesis 1: Computational Efficiency

The first hypothesis regarding superior computational efficiency was strongly supported. Headless models demonstrated substantial training speed improvements across all configurations: 32% faster for MLM architectures on the STRICT-small dataset, 34% faster for MLM on STRICT, and an impressive 53% faster for GPT architectures on the STRICT dataset. These efficiency gains validate (Godey et al., 2024)'s theoretical predictions about the computational benefits of removing the vocabulary projection head. The efficiency improvements were particularly pronounced for GPT models, likely due to the autoregressive nature of generation requiring repeated projection head computations during training.

5.1.2 Hypothesis 2: Performance Improvements

The second hypothesis regarding improved performance on GLUE and BLiMP benchmarks received mixed support. On GLUE tasks, vanilla models generally outperformed headless variants, though the differences were often within statistical significance ranges. The performance gap was most pronounced on the STRICT-small dataset (59.9% vs 58.8% average GLUE performance for the 600M token models) but narrowed considerably on the larger STRICT dataset (71.51% vs 70.98% for the 2B token models), suggesting that headless models may benefit more from increased training data.

Performance on the BLiMP benchmark proved to be rather difficult for all MLM models, rarely outperforming a random baseline. The BLiMP Benchmark the GPT models were significantly better (headless: 68.27% vs. vanilla: 67.99%), than the MLM models. However, neither architecture —MLM nor GPT— produced a headless model that significantly outperformed their vanilla counterpart and vice versa.

This allows us to answer the research question: 'How does a headless language model trained on the BabyLM 10-million and 100-million token dataset compare in performance to a standard prediction-headed model, and under what conditions might it outperform its traditional counterpart?'. The headless language model achieves scores on the GLUE and BLiMP benchmarks comparable to the prediction headed model's performance. This is true for both subsets of the BabyLM dataset. The headless language model outperforms the prediction-headed model on training speed, showing a significant speed up during pre-training.

5.2 Scaling Dynamics and Data Efficiency

A particularly intriguing finding was the differential scaling behaviour between architectures. While vanilla models maintained consistent advantages at smaller scales, headless models demonstrated superior scaling properties, nearly matching vanilla performance when trained on the full STRICT dataset. This suggests that the contrastive weight tying objective may require more data to reach its full potential, but ultimately achieves comparable results.

The poor absolute performance of both MLM architectures on BLiMP tasks (often near 50% accuracy) raises fundamental questions about the sufficiency of the training data. Even the larger STRICT dataset (100M tokens) appears insufficient for developing robust grammatical competence. This discrepancy between human learning efficiency and current computational approaches highlights a significant opportunity to further study language acquisition mechanisms.

5.3 Architectural Considerations and Linguistic Competence

The task-specific performance patterns reveal important insights about how different architectures capture linguistic phenomena. Vanilla models consistently outperformed on tasks requiring complex long-distance dependencies and licensing relationships (e.g., anaphor agreement, NPI licensing), suggesting that the traditional Cross Entropy objective has an advantage for certain grammatical computations. Conversely, headless models showed competitive or superior performance on specific syntactic phenomena, indicating that contrastive weight tying successfully captures important structural aspects of grammar. The dramatic differences in performance between MLM and GPT architectures on different benchmarks (GPT models achieving 68% on BLiMP but 36% on GLUE) suggest that architectural choice significantly influences the type of linguistic knowledge acquired.

5.4 Infrastructure- and Time Constraints

Initial experiments were conducted on a personal PC to test the feasibility of the approach, however memory constraints limited the system to running only the headless model variant. Due to these hardware limitations preventing the execution of more computationally demanding model configurations (the vanilla MLM model), the experimental setup was migrated to the ADA cluster to access sufficient computational resources. This transition to the cluster, while necessary for comprehensive testing, introduced additional complexity in terms of job scheduling, resource allocation, and debugging remote execution issues. This created major bottlenecks that went well beyond the anticipated setup time.

This experience shows how infrastructure challenges can seriously impact research timelines, especially when working with custom architectures like headless models that require specialized implementations. The time spent on technical setup forced reductions in the planned experimental scope—originally intended comprehensive evaluations across multiple datasets and longer training runs had to be scaled back to deliver meaningful results within the available time-frame.

The biggest constraint was being unable to train models across multiple GPUs due to memory limitations. The vanilla MLM architecture's memory requirements, mainly from the traditional prediction head and vocabulary-sized output layers, were too demanding for the available resources when using larger batch sizes. This forced a reduction to batch size 12, which allowed comparison across both dataset sizes and architectural variants but was not the original plan.

Interestingly, this computational bottleneck actually demonstrated one of the main advantages of the headless approach. The vanilla model's high memory demands made it difficult to run comprehensive experiments, while the headless model's lower memory footprint allowed more flexible configurations. This practical experience validated the accessibility benefits of removing computationally expensive components—the efficiency improvements weren't just theoretical but enabled broader experimental exploration despite resource constraints.

5.5 Future Research Directions

Several promising avenues for future research emerge from this work. First, combining the CWT training objective with established efficiency-focused approaches warrants investigation. Integrating CWT with knowledge distillation methods such as TinyBERT (Jiao et al., 2020) or DistilBERT (Sanh et al., 2020) could yield benefits for model compression while maintaining performance.

Second, given the specialized nature of the child-directed speech training dataset, examining the influence of tokenizer selection on model performance represents a valuable research direction. The unique linguistic characteristics of child-directed speech may benefit from domain-specific tokenization strategies that could enhance model effectiveness.

Third, a systematic investigation of batch size effects on headless model performance could provide crucial insights into optimal training dynamics for the Contrastive Weight Tying objective. This investigation is particularly relevant given Godey et al. (2024)'s assertion that increasing batch size in contrastive learning frameworks may not consistently improve performance, a finding supported by Awasthi et al. (2022).

Understanding these dynamics could inform best practices for CWT implementation across different model architectures and training scenarios.

Chapter 6

Conclusion

This research investigated the effectiveness of headless language models using contrastive weight tying (CWT) compared to traditional vanilla architectures on the BabyLM challenge datasets. Through systematic evaluation across multiple model configurations, training scales, and linguistic benchmarks, this study provides important insights into the viability of alternative pre-training objectives and the challenges of learning from developmentally constrained corpora.

The primary contribution of this work lies in demonstrating that the Contrastive Weight Tying objective represents a fundamentally viable alternative to traditional masked language modelling and next-token prediction paradigms. Rather than optimizing for vocabulary prediction accuracy, CWT employs contrastive learning with in-batch negative sampling to learn representations, eliminating the computationally expensive projection head while maintaining effective representation learning.

The success of this approach challenges the assumption that explicit vocabulary prediction is necessary for acquiring meaningful linguistic representations. The headless architecture's ability to achieve comparable performance to vanilla model demonstrates that language models can develop sophisticated internal representations through alternative learning objectives that focus on embedding quality rather than prediction accuracy.

The CWT objective achieved the predicted computational benefits, delivering substantial training speed improvements across all configurations. More significantly, the CWT objective demonstrated to have more benefits of scaling compared to traditional objectives. Headless models showed larger performance improvements when training data increased.

A critical finding that emerged from this research concerns the limitations imposed by the developmentally constrained training corpora rather than the architectural choices themselves. The poor absolute performance of both headless and vanilla Masked Language Modelling architectures on BLiMP tasks suggests that the constrained BabyLM datasets may be fundamentally insufficient for developing robust grammatical competence, regardless of the training objective employed. However, the GPT models achieved substantially better performance on BLiMP, indicating that autoregressive training objectives may be more effective at acquiring grammatical knowledge from developmentally constrained corpora.

The results of this study highlight the importance of distinguishing between architectural innovations and dataset constraints when evaluating model performance. The CWT objective's success in achieving comparable performance to traditional approaches, despite using a fundamentally different learning paradigm, suggests that the limitations observed in absolute performance scores stem primarily from the constrained nature of the BabyLM corpora rather than inherent weaknesses in the headless architecture.

Appendix A

Standard Deviation scores per GLUE task.

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE
100M	0,00	1,70	0,66/1,60	1,04/0,93	0,23/1,99	1,08	0,08	1,61
200M	1,24	1,10	1,16/2,53	3,29/3,84	0,05/0,56	0,66	0,02	2,05
400M	2,54	0,38	1,02/4,40	1,95/2,31	0,09/0,44	0,88	0,33	1,03
600M	2,83	0,95	0,57/2,07	1,09/1,14	0,13/0,31	0,09	0,65	3,13

Table A.1: Standard Deviation of the Headless Masked Language Models on the validation sets of the GLUE benchmark.

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE
100M	0,00	4,53	0,47/8,03	0,87/0,89	2,54/4,93	1,01	0,36	1,07
200M	1,69	1,28	1,72/1,42	2,61/2,61	0,20/0,40	0,59	0,29	0,93
400M	5,44	0,19	0,57/6,42	3,19/3,16	1,85/1,87	0,58	1,15	0,43
600M	2,96	0,43	1,91/0,69	0,83/0,86	0,21/0,52	0,29	0,19	1,41

Table A.2: Standard Deviation of the Vanilla Masked Language Models on the validation sets of the GLUE benchmark.

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE
1B	3,92	0,29	2,74/0,84	0,09/0,46	0,08/0,09	0,45	1,71	0,54
2B	9,00	0,32	1,14/1,68	0,81/0,78	0,14/0,21	0,49	0,33	1,87

Table A.3: Standard Deviation of the Headless Masked Language Models on the validation sets of the GLUE benchmark.

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE
1B	1,18	0,90	1,94/1,11	1,65/1,77	0,12/0,17	1,02	0,52	2,19
2B	14,89	0,50	1,46/4,59	2,61/1,65	0,69/0,74	0,21	0,21	0,66

Table A.4: Standard Deviation of the Vanilla Masked Language Models on the validation sets of the GLUE benchmark.

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE
1B	0,00	0,00	0,00/0,00	0,00/0,00	0,00/0,00	0,00	0,00	0,00
2B	0,00	0,00	0,00/0,00	0,02/0,00	0,00/0,00	0,00	0,00	2,71

Table A.5: Standard Deviation of the Headless GPT on the validation sets of the GLUE benchmark.

#tokens	COLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE
1B	0,00	0,00	0,00/0,00	0,00/0,00	0,00/0,00	0,00	0,00	0,00
2B	0,00	0,00	0,00/0,00	0,02/6,11	0,00/0,00	0,00	0,00	0,00

Table A.6: Standard Deviation of the Vanilla GPT on the validation sets of the GLUE benchmark.

References

P. Awasthi, N. Dikkala, and P. Kamath. Do More Negative Samples Necessarily Hurt In Contrastive Learning? In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, 2022. URL `https://proceedings.mlr.press/v162/awasthi22b.html`.

E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? . In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623, Virtual Event Canada, Mar. 2021. ACM. ISBN 978-1-4503-8309-7. doi: 10.1145/3442188.3445922. URL `https://dl.acm.org/doi/10.1145/3442188.3445922`.

S. Biderman, H. Schoelkopf, Q. Anthony, H. Bradley, K. O'Brien, E. Hallahan, M. A. Khan, S. Purohit, U. S. Prashanth, E. Raff, A. Skowron, and L. Sutawika. Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling. *Proceedings of the 40th International Conference on Machine Learning*, 202:2397–2430, 2023. URL `https://proceedings.mlr.press/v202/biderman23a.html`.

T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, and T. Henighan. Language Models are Few-Shot Learners. *Advances in neural information processing systems*, 33, 2020.

D. Campos. Curriculum learning for language modeling, Aug. 2021. URL `http://arxiv.org/abs/2108.02170`. arXiv:2108.02170 [cs].

L. G. G. Charpentier and D. Samuel. Not all layers are equally as important: Every Layer Counts BERT. In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 210–224, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.conll-babylm.20. URL `https://aclanthology.org/2023.conll-babylm.20`.

T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A Simple Framework for Contrastive Learning of Visual Representations. *Proceedings of the 37th International Conference on Machine Learning*, 2020.

D. Cheng, Y. Gu, S. Huang, J. Bi, M. Huang, and F. Wei. Instruction Pre-Training: Language Models are Supervised Multitask Learners. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2529–2550, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.148. URL `https://aclanthology.org/2024.emnlp-main.148`.

A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Z. X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, N. Fiedel, and R. Salakhutdinov. PaLM: Scaling Language Modeling with Pathways. *Journal of Machine Learning Research*, 24(240), 2023.

Y.-S. Chuang, R. Dangovski, H. Luo, Y. Zhang, S. Chang, M. Soljacic, S.-W. Li, S. Yih, Y. Kim, and J. Glass. DiffCSE: Difference-based Contrastive Learning for Sentence Embeddings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4207–4218, Seattle, United States, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.311. URL https://aclanthology.org/2022.naacl-main.311.

K. Clark, M.-T. Luong, and Q. V. Le. ELECTRA: PRE-TRAINING TEXT ENCODERS AS DISCRIMINATORS RATHER THAN GENERATORS. *ELECTRA*, 85, 2020.

B. Cottier, R. Rahman, L. Fattorini, N. Maslej, T. Besiroglu, and D. Owen. The rising costs of training frontier AI models, 2024. URL http://arxiv.org/abs/2405.21015. arXiv:2405.21015 [cs].

T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré. FLASHATTENTION: Fast and Memory-Efficient Exact Attention with IO-Awareness. *Advances in neural information processing systems*, 35, 2022.

T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer. QLoRA: Efficient Finetuning of Quantized LLMs. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1feb87871436031bdc0f2beaa62a049b-Paper-Conference.pdf.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 2019.

J. Dodge, T. Prewitt, R. Tachet Des Combes, E. Odmark, R. Schwartz, E. Strubell, A. S. Luccioni, N. A. Smith, N. DeCario, and W. Buchanan. Measuring the Carbon Intensity of AI in Cloud Instances. In *2022 ACM Conference on Fairness Accountability and Transparency*, pages 1877–1894, Seoul Republic of Korea, 2022. ACM. ISBN 978-1-4503-9352-2. doi: 10.1145/3531146.3533234. URL https://dl.acm.org/doi/10.1145/3531146.3533234.

N. Du, Y. Huang, A. M. Dai, S. Tong, D. Lepikhin, Y. Xu, M. Krikun, Y. Zhou, A. W. Yu, O. Firat, B. Zoph, L. Fedus, M. Bosma, Z. Zhou, T. Wang, Y. E. Wang, K. Webster, M. Pellat, K. Robinson, K. Meier-Hellstern, T. Duke, L. Dixon, K. Zhang, Q. V. Le, Y. Wu, Z. Chen, and C. Cui. GLaM: Efficient Scaling of Language Models with Mixture-of-Experts. In *International conference on machine learning*, 2022.

W. Fedus, B. Zoph, and N. Shazeer. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Journal of Machine Learning Research*, 23, 2022.

T. Gao, X. Yao, and D. Chen. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.552. URL `https://aclanthology.org/2021.emnlp-main.552`.

J. Gilkerson, J. A. Richards, S. F. Warren, J. K. Montgomery, C. R. Greenwood, D. Kimbrough Oller, J. H. L. Hansen, and T. D. Paul. Mapping the Early Language Environment Using All-Day Recordings and Automated Analysis. *American Journal of Speech-Language Pathology*, 26(2):248–265, 2017. ISSN 1058-0360, 1558-9110. doi: 10.1044/2016_AJSLP-15-0169. URL `http://pubs.asha.org/doi/10.1044/2016_AJSLP-15-0169`.

N. Godey, d. l. Clergerie, and B. Sagot. Headless Language Models: Learning without Predicting with Contrastive Weight Tying, 2023. URL `http://arxiv.org/abs/2309.08351`. arXiv:2309.08351 [cs].

N. Godey, V. d. l. Clergerie, and B. Sagot. Headless Language Models: Learning without Predicting with Contrastive Weight Tying. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=ONPECq0Rk7`.

Y. Goldberg and O. Levy. word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method, Feb. 2014. URL `http://arxiv.org/abs/1402.3722`. arXiv:1402.3722 [cs].

A. Gu and T. Dao. Linear-Time Sequence Modeling with Selective State Spaces. In *First Conference on Language Modeling*, 2024. URL `https://openreview.net/forum?id=tEYskw1VY2`.

J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L. Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, and K. Millican. Training Compute-Optimal Large Language Models. In *36th Conference on Neural Information Processing Systems*, 2022.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-Rank Adaptation of Large Language Models, Oct. 2021. URL `http://arxiv.org/abs/2106.09685`. arXiv:2106.09685 [cs].

A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. d. l. Casas, E. B. Hanna, F. Bressand, G. Lengyel, G. Bour, G. Lample, L. R. Lavaud, L. Saulnier, M.-A. Lachaux, P. Stock, S. Subramanian, S. Yang, S. Antoniak,

T. L. Scao, T. Gervet, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed. Mixtral of Experts, 2024. URL http://arxiv.org/abs/2401.04088. arXiv:2401.04088 [cs].

X. Jiao, Y. Yin, L. Shang, X. Jiang, X. Chen, L. Li, F. Wang, and Q. Liu. TinyBERT: Distilling BERT for Natural Language Understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.372. URL https://www.aclweb.org/anthology/2020.findings-emnlp.372.

B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, Dec. 2015. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.aab3050. URL https://www.science.org/doi/10.1126/science.aab3050.

K. Lee, D. Ippolito, A. Nystrom, C. Zhang, D. Eck, C. Callison-Burch, and N. Carlini. Deduplicating Training Data Makes Language Models Better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8424–8445, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.577. URL https://aclanthology.org/2022.acl-long.577.

Z. Li, H. Zhu, Z. Lu, and M. Yin. Synthetic Data Generation with Large Language Models for Text Classification: Potential and Limitations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10443–10461, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.647. URL https://aclanthology.org/2023.emnlp-main.647.

S.-Y. Liu, C.-Y. Wang, H. Yin, P. Molchanov, Y.-C. F. Wang, K.-T. Cheng, and M.-H. Chen. DoRA: Weight-Decomposed Low-Rank Adaptation. In *Forty-first International Conference on Machine Learning*, 2024.

A. S. Luccioni, S. Viguier, and A.-L. Ligozat. Estimating the Carbon Footprint of BLOOM, a 176B Parameter Language Model. *Journal of Machine Learning Research*, 24, 2023.

R. T. McCoy, J. Min, and T. Linzen. BERTs of a feather do not generalize together: Large variability in generalization across models with similar test set performance. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 217–227, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.21. URL https://www.aclweb.org/anthology/2020.blackboxnlp-1.21.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space, Sept. 2013. URL http://arxiv.org/abs/1301.3781. arXiv:1301.3781 [cs].

L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35, 2022.

A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving Language Understanding by Generative Pre-Training. 2018.

A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *roceedings of Machine Learning Research*, 2021. URL `https://proceedings.mlr.press/v139/radford21a.html`.

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21, 2020.

V. Sanh, L. Debut, J. Chaumond, and T. Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter, 2020.

R. Sennrich, B. Haddow, and A. Birch. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany, 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1009. URL `http://aclweb.org/anthology/P16-1009`.

L. Soldaini, R. Kinney, A. Bhagia, D. Schwenk, D. Atkinson, R. Authur, B. Bogin, K. Chandu, J. Dumas, Y. Elazar, V. Hofmann, A. Jha, S. Kumar, L. Lucy, X. Lyu, N. Lambert, I. Magnusson, J. Morrison, N. Muennighoff, A. Naik, C. Nam, M. Peters, A. Ravichander, K. Richardson, Z. Shen, E. Strubell, N. Subramani, O. Tafjord, E. Walsh, L. Zettlemoyer, N. Smith, H. Hajishirzi, I. Beltagy, D. Groeneveld, J. Dodge, and K. Lo. Dolma: an Open Corpus of Three Trillion Tokens for Language Model Pretraining Research. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15725–15788, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.840. URL `https://aclanthology.org/2024.acl-long.840`.

J. Steuer, M. Mosbach, and D. Klakow. Large GPT-like Models are Bad Babies: A Closer Look at the Relationship between Linguistic Competence and Psycholinguistic Measures. In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 114–129, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.conll-babylm.12. URL `https://aclanthology.org/2023.conll-babylm.12`.

E. Strubell, A. Ganesh, and A. McCallum. Energy and Policy Considerations for Modern Deep Learning Research. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(09):13693–13696, 2020. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v34i09.7123. URL `https://ojs.aaai.org/index.php/AAAI/article/view/7123`.

Y. Sun, L. Dong, S. Huang, S. Ma, Y. Xia, J. Xue, J. Wang, and F. Wei. Retentive Network: A Successor to Transformer for Large Language Models, 2024. URL `https://openreview.net/forum?id=UU9Icwbhin`.

Z. Sun, H. Yu, X. Song, R. Liu, Y. Yang, and D. Zhou. MobileBERT: a Compact Task-Agnostic BERT for Resource-Limited Devices. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2158–2170, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.195. URL `https://www.aclweb.org/anthology/2020.acl-main.195`.

G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican, and others. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom. Llama 2: Open Foundation and Fine-Tuned Chat Models, 2023. URL `http://arxiv.org/abs/2307.09288`. arXiv:2307.09288 [cs].

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Kaiser, and I. Polosukhin. Attention is All you Need. *Advances in neural information processing systems*, 30, 2017.

W. K. Vong, W. Wang, A. E. Orhan, and B. M. Lake. Grounded language acquisition through the eyes and ears of a single child. *Science*, 383(6682):504–511, 2024. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.adi1374. URL `https://www.science.org/doi/10.1126/science.adi1374`.

A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL `http://aclweb.org/anthology/W18-5446`.

A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems. *Advances in neural information processing systems*, 32, 2019.

Y. Wang, Y. Kordi, S. Mishra, A. Liu, N. A. Smith, D. Khashabi, and H. Hajishirzi. Self-Instruct: Aligning Language Models with Self-Generated Instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.754. URL `https://aclanthology.org/2023.acl-long.754`.

A. Warstadt, A. Parrish, H. Liu, A. Mohananey, W. Peng, S.-F. Wang, and S. R. Bowman. BLiMP: The Benchmark of Linguistic Minimal Pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392, 2020. ISSN 2307-387X. doi: 10.1162/tacl_a_00321. URL `https://direct.mit.edu/tacl/article/96452`.

A. Warstadt, L. Choshen, A. Mueller, A. Williams, E. Wilcox, and C. Zhuang. Call for Papers - The BabyLM Challenge: Sample-efficient pretraining on a developmentally plausible corpus, 2023a. URL `http://arxiv.org/abs/2301.11796`. arXiv:2301.11796 [cs].

A. Warstadt, A. Mueller, L. Choshen, E. Wilcox, C. Zhuang, J. Ciro, R. Mosquera, B. Paranjabe, A. Williams, T. Linzen, and R. Cotterell. 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 1–6, Singapore, 2023b. Association for Computational Linguistics. doi: 10.18653/v1/2023.conll-babylm.1. URL `https://aclanthology.org/2023.conll-babylm.1`.

X. Wu, E. Dyer, and B. Neyshabur. WHEN DO CURRICULA WORK? In *International Conference on Learning Representations*, 2021.

S. M. Xie, S. Santurkar, T. Ma, and P. Liang. Data Selection for Language Models via Importance Resampling. *Advances in Neural Information Processing Systems*, 36, 2023.

B. Xu, L. Zhang, Z. Mao, Q. Wang, H. Xie, and Y. Zhang. Curriculum Learning for Natural Language Understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6095–6104, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.542. URL `https://www.aclweb.org/anthology/2020.acl-main.542`.

A. Yamaguchi, G. Chrysostomou, K. Margatina, and N. Aletras. Frustratingly Simple Pretraining Alternatives to Masked Language Modeling. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3116–3125, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.249. URL `https://aclanthology.org/2021.emnlp-main.249`.

S. Ye, J. Kim, and A. Oh. Efficient Contrastive Learning via Novel Data Augmentation and Curriculum Learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1832–1838, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.138. URL `https://aclanthology.org/2021.emnlp-main.138`.

L. Zhang, C. Bao, and K. Ma. Self-Distillation: Towards Efficient and Compact Neural Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021. ISSN 0162-8828, 2160-9292, 1939-3539. doi: 10.1109/TPAMI.2021.3067100. URL `https://ieeexplore.ieee.org/document/9381661/`.

Q. Zhang, M. Chen, A. Bukharin, P. He, Y. Cheng, W. Chen, and T. Zhao. Adaptive Budget Allocation for Parameter-Efficient Fine-Tuning. In *The Eleventh International COnference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=lq62uWRJjiY`.