

Contrastive Decoding for Synthetic Data Generation in Low-Resource Language Modeling

Jannek Ulm¹ Kevin Du¹ Vésteinn Snæbjarnarson^{1,2}

¹ETH Zürich ²University of Copenhagen

jannek.ulm@gmail.com kevin.du@inf.ethz.ch vest.snae@gmail.com

Abstract

Large language models (LLMs) are trained on huge amounts of textual data, and concerns have been raised that the limits of such data may soon be reached. A potential solution is to train on synthetic data sampled from LLMs. In this work, we build on this idea and investigate the benefits of *contrastive decoding* for generating synthetic corpora. In a controlled setting, we experiment with sampling corpora using the relative difference between a GOOD and BAD model trained on the same original corpus of 100 million words. By amplifying the signal from a model that has better performance, we create a synthetic corpus and mix it with the original training data. Our findings show that training on a mixture of synthesized and real data improves performance on the language modeling objective and a range of downstream tasks. In particular, we see that training with a mix of synthetic data from contrastive decoding benefits tasks that require more *reasoning skills*, while synthetic data from traditional sampling helps more on tasks dependent on surface-level *linguistic capabilities*.

<https://github.com/janulm/CD-for-Synthetic-Data-Generation>

1 Introduction

Large language models (LLMs) require enormous amounts of text to achieve strong performance (Kaplan et al., 2020; Hoffmann et al., 2022). For the largest models, it has even been claimed that current training regimes already consume the vast majority of publicly available text on the internet (Villalobos et al., 2024; Dubey et al., 2024). The BabyLM Challenge (Charpentier et al., 2025) emphasizes this point by asking what can be learned under a strict budget of 100M words, prioritizing data efficiency over raw scale, mimicking the far more efficient language learning capabilities of humans. Furthermore, not all training data is equally beneficial (Eldan and Li,

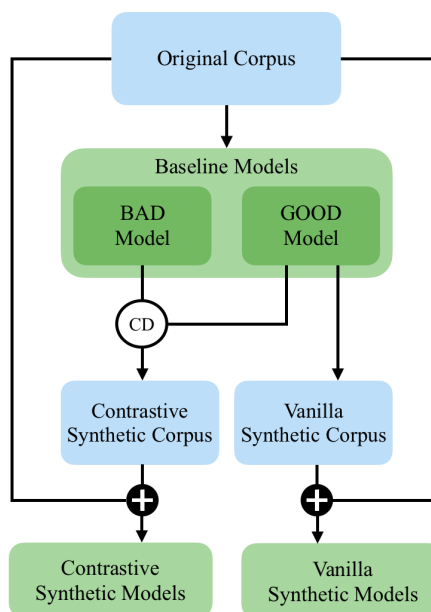


Figure 1: Our synthetic data generation and training pipeline: Start by training baseline LMs on a “real” corpus (*TinyBabyLM*: human-written text + *TinyStories*). The GOOD model is the best checkpoint; the BAD model is a weaker variant, e.g., an earlier checkpoint. We generate synthetic corpora via (i) *contrastive decoding* (CD), and (ii) non-contrastive ancestral (*vanilla*) sampling. We then train new models on a mixture of the original and synthetic corpora. We find that contrastive models improve the most over the BASELINE in evaluations on reasoning-oriented benchmarks, such as entity tracking.

2023; Gunasekar et al., 2023). The question thus arises: How can we get more high-quality data in a constrained setting? One proposed solution is to generate synthetic data using existing pre-trained models, thereby expanding the available corpus without collecting more human-written text (Wang et al., 2023; Eldan and Li, 2023; Gunasekar et al., 2023; Abdin et al., 2024).

Generating synthetic data is non-trivial, however. The quality of synthetic text may be hindered by

noise, factual errors, or stylistic artifacts (Lin et al., 2022; Huang et al., 2025). Models may also replicate or even amplify biases from their training data (Gallegos et al., 2024; Bender et al., 2021), and generated text may diverge from the target distribution, leading to potential degradation in downstream performance or model collapse (Dohmatob et al., 2025; Gerstgrasser et al., 2024; Shumailov et al., 2024). Moreover, producing high-quality synthetic data is particularly difficult because language models often hallucinate facts or repeat memorized content from their original training corpus (Bender et al., 2021; Lin et al., 2022).

This work explores the use of contrastive decoding (CD) (Li et al., 2023) to generate synthetic data in a controlled setting. CD is a decoding strategy that takes advantage of the differences between a GOOD model and a BAD model to produce more coherent and informative text. In prior work, CD has been largely restricted to improving the quality of responses generated for inference-time tasks (Li et al., 2023; O’Brien and Lewis, 2023; Chang et al., 2024). In contrast, we use CD to synthesize corpora to train new models from scratch. Our goal is to know whether these inference-time benefits of CD translate into gains when generating synthetic data for training language models.

The high-level experimental approach is illustrated in Figure 1 and goes as follows.

1. Start with an original corpus (100M tokens, BabyLM setting (Charpentier et al., 2025)).
2. Train BASELINE models (100M-parameter models based on the Llama 2 architecture (Touvron et al., 2023)) on the original corpus.
3. Generate synthetic corpora (100M tokens each) using CD and standard sampling.
4. Train models on the original and synthetic corpora.
5. Evaluate models on downstream tasks and compare to BASELINE.

We find that synthetic data improves performance on the language-modeling objective and downstream tasks. Moreover, tasks that emphasize reasoning benefit most from CD-generated data, whereas tasks emphasizing linguistic competence gain more from standard (non-contrastive) sampling.

2 Synthetic Data Generation for Pre-training Language Models

Recent work shows that *curated, high-quality* synthetic corpora can substantially boost data efficiency for small or low-resource LMs (Eldan and Li, 2023). Carefully constructed “textbook”-style corpora improve generalization (Gunasekar et al., 2023), and iterative pipelines that generate, critique, and revise synthetic content have been shown to boost reasoning-oriented capabilities (Abdin et al., 2024). Domain-targeted corpora can be especially effective: *TinyStories* demonstrates that fully synthetic, child-directed narratives enable 1–10M-parameter models to produce multi-paragraph coherent and grammatical text (Eldan and Li, 2023). For instruction following, Self-Instruct bootstraps instruction-response pairs from a seed set, leading to gains without additional human annotation (Wang et al., 2023). These results collectively suggest that synthetic data can significantly increase downstream performance.

However, naive reuse of model-generated text across generations can severely harm performance, resulting in “model collapse” (Shumailov et al., 2024; Gerstgrasser et al., 2024; Dohmatob et al., 2025). Empirically, careful filtering, diversification, and sustained mixing with real data mitigate such risks while preserving gains (Gerstgrasser et al., 2024). In this work, we explore an orthogonal axis: *decoding-control* for synthetic corpora. Specifically, we study whether CD can produce higher-signal synthetic corpora for pre-training under a tight data budget, compared to non-contrastive approaches.

3 Contrastive Decoding

Language-models. Following Cotterell et al. (2024), let Σ be a set of tokens we call the vocabulary, the Kleene closure Σ^* be the set of all strings built from Σ , if p is a probability distribution over Σ^* we say it is a **language model**. Then, $p(x_i | \mathbf{x}_{<i})$ represents the model’s *next-token* probability, i.e., the probability that the next token is x_i given the preceding context $\mathbf{x}_{<i} \stackrel{\text{def}}{=} x_0 x_1 \dots x_{i-1}$.

Contrastive Decoding We now describe the CD approach in detail. Let p_G be a GOOD (better performing) language model, and p_B be a BAD (worse performing) language model. Following Li et al. (2023), we define $\mathcal{V}_{\text{head}}$ as the set of likely

tokens under p_G :

$$\mathcal{V}_{\text{head}}(\mathbf{x}_{<i}) \stackrel{\text{def}}{=} \{x_i \in \Sigma : p_G(x_i | \mathbf{x}_{<i}) \geq \alpha \max_{w \in \Sigma} p_G(w | \mathbf{x}_{<i})\}. \quad (1)$$

Where α is a scalar hyper-parameter. The contrastive score CD for $x_i \in \mathcal{V}_{\text{head}}(\mathbf{x}_{<i})$ is then defined as follows:

$$\text{CD}(x_i | \mathbf{x}_{<i}) \stackrel{\text{def}}{=} \log p_G(x_i | \mathbf{x}_{<i}) - \lambda \log p_B(x_i | \mathbf{x}_{<i}), \quad (2)$$

where contrast strength is controlled by a scalar λ . Further, if $x_i \notin \mathcal{V}_{\text{head}}(\mathbf{x}_{<i})$ then $\text{CD}(x_i | \mathbf{x}_{<i}) \stackrel{\text{def}}{=} -\infty$. Typically, the contrastive scores $\text{CD}(\cdot | \mathbf{x}_{<i})$ are treated as logits giving rise to a new probability distribution over Σ from which we can decode the next token.

Background and variants. CD biases generation toward tokens preferred by a stronger GOOD model while down-weighting those preferred by a weaker BAD model, under the plausibility mask $\mathcal{V}_{\text{head}}$ (Li et al., 2023). Empirically, CD reduces repetition and topic drift in open-ended generation and, without additional training, improves reasoning-focused decoding compared to greedy or nucleus (top- p) sampling (Li et al., 2023; O’Brien and Lewis, 2023).

Several works adapt CD to lower its compute and memory cost or to strengthen specific capabilities. Phan et al. (2024) replace an explicit bad model with a distilled proxy (e.g., via dropout or quantization), retaining most of CD’s gains while reducing memory. In retrieval or context-heavy settings, Zhao et al. (2024) integrate CD with adversarial negatives so that decoding remains grounded in relevant passages. These methods focus on evaluating the CD-like inference performance, rather than on generating pre-training corpora.

Relation to synthetic-data generation. A related approach is STEER, which performs contrastive expert guidance by subtracting a base model from a fine-tuned domain expert and combining it with negative prompting to generate synthetic corpora for downstream fine-tuning (O’Neill et al., 2023). In contrast, we use CD with a general GOOD/BAD pair trained on the same base corpus and treat CD as a data generator for pre-training: we synthesize full corpora and then train new models from scratch on mixtures of real and synthetic text. This lets us test whether CD’s inference-time

benefits translate into better pre-training signals, and how they compare to vanilla sampling under a fixed data budget.

4 Training on Synthetic Data

Given the success of CD in generating higher scoring text for evaluations, we ask whether it can also be employed to generate higher-quality text for pre-training. This section describes our procedure for generating synthetic corpora using CD and training models on them.

4.1 Synthetic Corpus Generation

General Procedure. To ensure independence from the training data, following (Wang et al., 2023) we generate synthetic corpora from *prefix seeds* that are held out from all training and evaluation data. The prefix seeds are evenly sampled across the four data sources to preserve balance, we describe this in more detail in Section 5.1. For each prefix seed, we fix the first 20 tokens as a context prefix, and then we sample continuations from the target model. To ensure sufficient diversity and corpus size, we produce eight completions of up to 400 tokens per seed. To sample each next token, we use the decoding strategies described below. Using $\sim 30.4\text{K}$ generation seeds we produce approximately 100M tokens for each decoding strategy.

Decoding Strategies. We mainly compare two decoding settings that differ only in how candidate tokens are scored before sampling. Let $\mathcal{V}_{\text{head}}(\mathbf{x}_{<i})$ be the set of α -likely tokens of the GOOD distribution p_G as defined in Eq. (1); If $\mathcal{V}_{\text{head}}(\mathbf{x}_{<i})$ is applied, tokens outside $\mathcal{V}_{\text{head}}$ are assigned score $-\infty$ (Li et al., 2023). Let the contrastive score $\text{CD}(x_i | \mathbf{x}_{<i})$ be as in Eq. (2). For CD we treat $\text{CD}(\cdot | \mathbf{x}_{<i})$ as a logit over $\mathcal{V}_{\text{head}}(\mathbf{x}_{<i})$, i.e., we sample with probabilities proportional to $\exp(\text{CD}(x_i | \mathbf{x}_{<i}))$.

1. **NO-CONTRAST:** Ancestral sampling from $p_G(\cdot | \mathbf{x}_{<i})$.
2. **CONTRASTIVE DECODING (CD):** Ancestral sampling within $\mathcal{V}_{\text{head}}(\mathbf{x}_{<i})$ using logits $\text{CD}(x_i | \mathbf{x}_{<i})$ (Eq. (2)), which promote tokens preferred by p_G over p_B .

We also study the effect of truncating the sampling support to further suppress low-probability continuations as follows:

3. **NO-CONTRAST + $\mathcal{V}_{\text{head}}$:** Ancestral sampling from $p_G(\cdot | \mathbf{x}_{<i})$ restricted to $\mathcal{V}_{\text{head}}(\mathbf{x}_{<i})$.

4. **NO-CONTRAST + top- p** : Ancestral sampling from $p_G(\cdot | \mathbf{x}_{<i})$ restricted to top- p selection (Holtzman et al., 2020).
5. **NO-CONTRAST + top- k** : Ancestral sampling from $p_G(\cdot | \mathbf{x}_{<i})$ restricted to top- k selection (Fan et al., 2018).
6. **CD with top- p** : Ancestral sampling restricted to the top- p after already restricting to the $\mathcal{V}_{\text{head}}(\mathbf{x}_{<i})$ using logits $\text{CD}(x_i | \mathbf{x}_{<i})$ (Eq. (2)).
7. **CD with top- k** : Ancestral sampling restricted to the top- k after already restricting to the $\mathcal{V}_{\text{head}}(\mathbf{x}_{<i})$ using logits $\text{CD}(x_i | \mathbf{x}_{<i})$ (Eq. (2)).

We sweep $k \in \{50, 100, 200\}$ and $p \in \{0.90, 0.95, 0.97\}$ and report effects on performance in Section 6.4 and Table 4.

4.2 The BAD and GOOD Models

We consider three approaches to instantiate a BAD model p_B (details in Appendix A):

- i) **Smaller models** that are $10\times$, $20\times$, $50\times$, and $100\times$ smaller than the GOOD model, and, following (Li et al., 2023), selecting the checkpoint with the best evaluation perplexity.
- ii) **Earlier checkpoints**, e.g., if a GOOD checkpoint is taken at step 2500, we test BAD checkpoints at steps 2000, 1500, 1000 and 500.
- iii) **Attention dropout**, where the BAD model is the GOOD model, but run with attention dropout rates $\{0.1, 0.3, 0.5, 0.7\}$ at inference time (Phan et al., 2024).

Note on scale. Prior evaluations of CD use billion-parameter GOOD models paired with much smaller BAD models (e.g., OPT-13B vs. OPT-125M; GPT-2-XL vs. GPT-2-small), and report that performance improves as the GOOD-BAD *scale gap* increases (Li et al., 2023, §7.1; Fig. 2). CD is not limited to GOOD models that are several billion parameters or larger, e.g., as Li et al. (2023) also show gains with GPT-2-XL ($\sim 1.5\text{B}$). However, the observed size-gap effect suggests that a strong contrast may be harder to elicit at our $\sim 100\text{M}$ -parameter scale. Consistent with this, O’Brien and Lewis (2023) find that smaller BAD models help more than larger ones and that gains tend to

be stronger for larger GOOD models on reasoning tasks. We therefore investigate multiple BAD model instantiations to identify how we can elicit a sufficient contrastive signal at this scale (Li et al., 2023; O’Brien and Lewis, 2023; Phan et al., 2024).

Hyperparameters. Following Li et al. (2023), we use $\alpha = 0.1$ for $\mathcal{V}_{\text{head}}$ and the contrast strength is set to $\lambda = 1$.

The GOOD models. We describe how the better models, p_G , are selected in Section 5.2.

4.3 Training with Mixed Corpora

All models are trained from scratch to isolate the effect of the synthetic corpora. For each decoding method, we mix its 100M-token synthetic corpus with the same 100M-token TinyBabyLM corpus (see Section 5.1) used for the baselines, while keeping initialization seeds, training length, and optimization hyperparameters identical to the baseline runs. Batches contain 256 sequences of 1024 tokens with a fixed 70/30 mixture at the sequence level (70% real, 30% synthetic) and are repeatedly regrouped and re-tokenized to act as a data regularizer (see Section 5.2 and Appendix A).

We ablate the original/synthetic mixture and report its effect on performance in Section 6.5. Since initial testing indicated that the 70/30 mixture achieved the strongest average performance across tasks, we report results under this fixed ratio in the main experiments.

5 Experimental Details

5.1 TinyBabyLM Corpus

We start from the BabyLM 100M corpus and construct a modified variant by replacing the CHILDES, BNC and SWITCHBOARD portions with the synthetic *TinyStories* (Eldan and Li, 2023). We add a portion of *TinyStories* because Eldan and Li (2023) show that their corpus, a constrained, child-directed synthetic corpus enables very small models (1–10M parameters) to learn fluent, grammatical multi-paragraph stories, making it a high-signal, data-efficient addition for low-resource pretraining. Concretely, we substitute $\sim 39.7\text{M}$ words of *TinyStories* for the removed words, yielding the following composition: Gutenberg (27.4M), SimpleWiki (14.9M), OpenSubtitles (17.7M) and *TinyStories* (39.7M) (Eldan and Li, 2023; Gerlach and Font-Clos, 2020; Lison and Tiedemann, 2016). We refer

to this modified corpus as *TinyBabyLM*. The total amount of human-written+TinyStories text is held at ≈ 100 M words; note that “words” here denote whitespace-delimited tokens, so totals differ from BPE token counts used during training (see Section A). We partition TinyBabyLM into three disjoint splits: *train* (90.5M words), *eval* (8.9M words), and *seeds* (600K words). The generation seeds (a selection of ~ 30 K paragraph start prefixes) for synthetic generation are sampled exclusively from the *seeds* split and are strictly disjoint from all *train* and *eval* text. To maintain balance across domains, the splits, *seed*, *train* and *eval*, are distributed evenly across the four data sources.

5.2 Model Architecture & Training Setup

We use a decoder-only Transformer LLaMA-2 architecture with ~ 100 M parameters (Touvron et al., 2023): 12 layers, hidden size 768, 12 attention heads, MLP intermediate size 3072, and a maximum context length of 1024 tokens. All models are trained from scratch with the same initialization scheme.

Tokenization is performed with a SentencePiece BPE tokenizer (vocabulary size 32k) trained on the TinyBabyLM corpus; the same tokenizer is used for all experiments to ensure comparability (see Appendix A for details).

Training uses a global batch of 256 sequences \times 1024 tokens, AdamW with weight decay 0.1, and a cosine learning-rate schedule: peak 1×10^{-3} , 150 warm-up steps, and decay to zero by step 8000. The training duration is fixed to 8000 steps for every run and checkpoints are saved every 500 steps. Each experimental condition is repeated with $n = 10$ distinct random seeds.

Data pipeline (applies to all runs). Real and synthetic corpora are stored as rows of text and, at the start of training, are independently shuffled, tokenized, and split into fixed-length sequences. Sampling proceeds until a corpus is exhausted, at which point that corpus is reshuffled, and re-segmented before resuming. This periodic resegmentation acts as a regularizer and is applied identically to baseline and mixed-data runs.

Good checkpoint selection. From each of the $n = 10$ BASELINE seeds, we first select the saved checkpoint with the lowest perplexity, forming the candidate set \mathcal{X} . We then evaluate only \mathcal{X} on the full suite of tasks, convert scores to percentiles within the task, average percentile across tasks, and

choose as the GOOD model the checkpoint with the highest average percentile.

5.3 Evaluation & Statistical Analysis

Benchmarks. We evaluate on the zero-shot BabyLM evaluation suite¹ and report Perplexity on the TinyBabyLM *eval*-split (see 5.1). The tasks considered are:

- **BLiMP:** Benchmark of Linguistic Minimal Pairs testing core English grammar linguistic competence (Warstadt et al., 2020).
- **BLiMP Supplement:** BLiMP-style suite, extending to dialogue and question answering, focused on reasoning, syntax and semantics (Hu et al., 2024; Warstadt et al., 2023).
- **EWoK:** Checks for social/physical/world knowledge and semantic understanding (Ivanova et al., 2024).
- **Entity Tracking:** Requires maintaining and updating entity states across text to test memory and state reasoning (Kim and Schuster, 2023).
- **WUG:** Evaluates morphology, evaluating on adjective nominalization to estimate linguistic generalization (Hofmann et al., 2025).
- **Reading:** Compares model surprisal to human word-by-word reading times to assess processing alignment (De Varda et al., 2023).
- **Eye-Tracking:** Tests whether model predictability tracks human eye-movement measures during reading (De Varda et al., 2023).

The metric used for the **Reading** and **Eye-tracking** tasks is the partial change (%) in the coefficient of determination, that is, the additional proportion of variance explained. For the other tasks, accuracy is used.

Per-task mean-max over checkpoints. For each training method² m , benchmark task t , and initialization seed s , we save checkpoints every 500 steps and select the best checkpoint independently per (m, t, s) . Let $\mathcal{C}_{m,t,s}$ denote the set of saved checkpoints over the steps, and $S_{m,t,s}(c)$ the task score at checkpoint c . For higher-is-better tasks, we set

$$c_{m,t,s}^* \stackrel{\text{def}}{=} \arg \max_{c \in \mathcal{C}_{m,t,s}} S_{m,t,s}(c),$$

¹<https://github.com/babylm/evaluation-pipeline-2025>

²Either BASELINE, or a pair of decoding strategy from 4.1 and bad model setting from 4.2

Name	Perplexity↓	BLiMP↑	BLiMP Supp.↑	Entity Tracking↑	EWoK↑	WUG↑	Reading↑	Eye Tracking↑
GOOD	24.62	71.22	63.50	27.01	53.64	57.50	1.44	3.51
BASELINE	24.46±0.10	71.03±0.27	64.10±0.60	27.82±1.18	53.18±0.28	66.90±2.47	1.76±0.22	3.85±0.31

Table 1: Reference performance of the BASELINE (mean \pm s.e., $n=10$ independent runs; per-task mean-max checkpointing per Section 5.3) versus the single fixed GOOD checkpoint. Because it is a single checkpoint chosen once across seeds rather than per task, it can sit below the BASELINE mean on some tasks.

while for perplexity we take $\arg \min$. The selected checkpoint $c_{m,t,s}^*$ is then evaluated. This procedure estimates the best attainable performance per task under the fixed training budget and avoids coupling to a single global checkpoint.

Paired bootstrap for statistical significance.

Evaluation of checkpoint $c_{m,t,s}^*$ for task t yields per-example outcomes $y_{m,t,s,i}$ for examples $i = 1, \dots, N_t$. We use paired bootstrap with $B = 1000$ resamples to calculate confidence intervals. For each (t, s) and bootstrap draw b , sample the index-set $I^{(b)}$ of size N_t with replacement from $\{1, \dots, N_t\}$ and apply the *same* $I^{(b)}$ to all methods (pairing). We average out uncertainty over the seeds:

$$\bar{y}_{m,t,s}^{(b)} = \frac{1}{N_t} \sum_{i \in I^{(b)}} y_{m,t,s,i} \quad (3)$$

$$\mu_{m,t}^{(b)} = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \bar{y}_{m,t,s}^{(b)}. \quad (4)$$

As in (3), $\bar{y}_{m,t,s}^{(b)}$ is the mean for tasks with per-example scalar scores (e.g., BLiMP, EWoK). For metrics with task-specific aggregations (e.g., Perplexity or Reading), we substitute the appropriate aggregation function and proceed identically. For a comparison of two methods m_1 and m_2 , we form the bootstrap difference distribution

$$\Delta_t^{(b)} = \mu_{m_1,t}^{(b)} - \mu_{m_2,t}^{(b)} \quad (5)$$

We compute 95% confidence intervals via the percentile method, $CI_{95} = [\text{pct}_{2.5}, \text{pct}_{97.5}]$ of $\{\Delta_t^{(b)}\}_{b=1}^B$. A difference is deemed significant if $0 \notin CI_{95}$. We compute one-sided p -values in the direction of the observed effect using the estimator on the bootstrap differences $\{\Delta_t^{(b)}\}_{b=1}^B$: for higher-is-better tasks with $\hat{\Delta}_t > 0$,

$$p = \frac{1 + \sum_{b=1}^B \mathbb{I}\{\Delta_t^{(b)} \leq 0\}}{B + 1} \quad (6)$$

and if $\bar{\Delta}_t < 0$ use \geq instead. For lower-is-better metrics we swap the inequality accordingly.

Aggregate reporting. For tables and figures, we bold the best method per benchmark and mark significant improvements/degradations relative to the BASELINE. We report, for each method m and task t , the bootstrap mean $\bar{\mu}_{m,t}$ and standard-error.

$$\bar{\mu}_{m,t} = \frac{1}{B} \sum_{b=1}^B \mu_{m,t}^{(b)}, \quad \widehat{SE}_{m,t} = \frac{\sigma_{m,t}}{\sqrt{B}} \quad (7)$$

This analysis serves to estimate the maximum achievable performance for each method, on each task, given the training setup. Our aggregating metric $\mu_{\Delta\text{REL}}$ is the mean relative performance, across all tasks except Perplexity, vs. the BASELINE—i.e., it is the average proportional change given in percentages.

6 Results

6.1 BASELINE Performance

Table 1 summarizes the performance of our reference points, the GOOD and BASELINE results. Recall that the BASELINE row reports the mean \pm s.e. over $n=10$ independent runs under our per-task bootstrapped mean-max evaluation (Section 5.3). In contrast, the GOOD model is a *single* checkpoint selected once, across seeds, using the selection procedure described in Section 5.2. As such GOOD is broadly representative of a strong model but sits slightly below the BASELINE mean on some tasks (e.g., Perplexity, BLiMP Supplement) because it cannot adapt per task. We use this fixed checkpoint as the GOOD model in all subsequent synthetic corpora and comparisons.

6.2 Contrastive vs. non-contrastive generation

Setup. Recall from Section 4.1 that we compare the three generation settings for synthesizing the 100M-token corpora: (i) NO-CONTRAST, (ii) NO-CONTRAST- $\mathcal{V}_{\text{HEAD}}$, and (iii) CD. Among all contrastive instantiations, using the early checkpoint at 500 steps (**CD-Early-500**) emerged as the strongest (see Section 6.3), and we use it as our CD representative in this section. Results are summarized in Table 2.

Name	$\mu_{\Delta\text{REL}} \uparrow$	Perplexity \downarrow	BLiMP \uparrow	BLiMP Supp. \uparrow	Entity Tracking \uparrow	EWoK \uparrow	WUG \uparrow	Reading \uparrow	Eye Tracking \uparrow
Baseline	-	24.46 \pm 0.10	71.03 \pm 0.27	64.10 \pm 0.60	27.82 \pm 1.18	53.18 \pm 0.28	66.90 \pm 2.47	1.76 \pm 0.22	3.85 \pm 0.31
No-Contrast	2.96%	23.56\pm0.11*	72.09\pm0.17*	64.83 \pm 0.73	28.14 \pm 1.75	53.17 \pm 0.30	64.67 \pm 1.66*	1.91\pm0.25	4.31 \pm 0.33*
No-Contrast-V-Head	0.66%	24.33 \pm 0.10*	71.67 \pm 0.24*	64.86 \pm 0.74	25.47 \pm 1.40*	53.03 \pm 0.31	66.67 \pm 1.58	1.76 \pm 0.23	4.32 \pm 0.33*
CD-Early-500	4.90%	23.73 \pm 0.10*	71.72 \pm 0.19*	65.10\pm0.60*	30.38\pm0.65*	53.80\pm0.29*	70.55\pm2.32*	1.79 \pm 0.22	4.42\pm0.32*

Table 2: Task-by-task results for synthetic-data regimes. Entries are mean \pm s.e.; * denotes a significant difference vs. BASELINE. CD-Early-500 attains the best overall $\mu_{\Delta\text{REL}}$ (+4.90%) and leads on BLiMP Supplement, Entity Tracking, EWoK, WUG, and Eye Tracking, while NO-CONTRAST yields the lowest Perplexity, the best BLiMP and Reading. Find relative change vs. BASELINE at Table 6

Aggregate performance. All synthetic regimes beat BASELINE. CD delivers the strongest overall gains ($\mu_{\Delta\text{REL}}$ +4.90%), with the non-contrastive variants lacking, see Table 2.

Language modeling (Perplexity). Perplexity drops for every method. NO-CONTRAST attains the lowest value (23.56), with CD close behind, so non-contrastive sampling edges out CD slightly on the LM objective, while CD still clearly improves over BASELINE; see Table 2.

Metric	CD vs NO-CONTRAST	Significance
$\mu_{\Delta\text{REL}}^{\text{CD}} - \mu_{\Delta\text{REL}}^{\text{NO-CONTRAST}}$	+1.94pp	
Perplexity \downarrow	-0.7%	***
BLiMP \uparrow	-0.5%	***
BLiMP Supp. \uparrow	+0.4%	
Entity Tracking \uparrow	+7.3%	***
EWoK \uparrow	+1.2%	*
WUG \uparrow	+8.2%	***
Reading \uparrow	-6.2%	
Eye Tracking \uparrow	+2.5%	

Table 3: Statistical significance and relative change of CD-EARLY-500 vs. NO-CONTRAST by metric. Entries are percentage changes; for Perplexity (\downarrow), more negative is better, while for all others (\uparrow), more positive is better. The ‘‘Significance’’ column reports paired-bootstrap one-sided p -values per Section 5.3: * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$ (blank = not significant). $\mu_{\Delta\text{REL}}$ is shown as an absolute difference in percentage points (pp).

Task-level pattern and head-to-head. CD performs best on five tasks and shows significant gains on five, notably on reasoning-/tracking-oriented evaluations like BLiMP Supplement, Entity Tracking, and EWoK (see Table 2). In contrast, NO-CONTRAST is best on three tasks with significant effects on four, and it leads on core linguistic competence with Perplexity and BLiMP. In direct statistical comparisons (CD vs. NO-CONTRAST), as displayed in Table 3, NO-CONTRAST has a small but significant edge on Perplexity and BLiMP, whereas CD achieves significant, and generally larger, gains on Entity Tracking, EWoK, and WUG. The remaining tasks show no reliable difference.

Is it the $\mathcal{V}_{\text{head}}$ mask or the contrastive logits? NO-CONTRAST- $\mathcal{V}_{\text{HEAD}}$ serves as a control that isolates the effect of restricting to the α -head without any contrastive subtraction. If head-masking alone explained CD’s gains, NO-CONTRAST- $\mathcal{V}_{\text{HEAD}}$ would mirror CD. It does not: while NO-CONTRAST- $\mathcal{V}_{\text{HEAD}}$ modestly helps Perplexity (-0.51%) and Eye Tracking (+12.12%), it significantly hurts Entity Tracking (-8.45%) and yields small/neutral changes elsewhere (Table 2). This suggests that the improvements are driven by the contrastive logits and not the $\mathcal{V}_{\text{head}}$ constraint.

Takeaway. Mixing synthetic data consistently helps. Among generation strategies, CD delivers the strongest overall improvements and a clear advantage on reasoning-oriented benchmarks, while NO-CONTRAST remains best for the LM objective and BLiMP. The NO-CONTRAST- $\mathcal{V}_{\text{HEAD}}$ control suggests that contrastive scoring, not head-masking, is the key to CD’s benefits.

Name	$\mu_{\Delta\text{REL}} \uparrow$	Perplexity \downarrow
BASELINE	-	24.46 \pm 0.10
NO-CONTRAST	2.96%	23.56\pm0.11* (3.68%)
NO-CONTRAST-Top-k-200	3.65%	23.65 \pm 0.10* (3.29%)
CD-Small-20	3.55%	23.73 \pm 0.14* (2.96%)
CD-Drop-0.7	3.29%	24.06 \pm 0.13* (1.65%)
CD-Early-500	4.90%	23.73 \pm 0.10* (2.98%)
CD-Early-500-Top-k-200	5.69%	23.77 \pm 0.10* (2.80%)

Table 4: Comparison of CD variants (early checkpoint, smaller model, dropout) against non-contrastive baselines, including the best truncation configurations. The best truncation for both regimes is Top-k=200; CD-Early-500-Top-k-200 achieves the highest overall task improvement at unchanged perplexity.

6.3 Searching for Effective CD Settings

We instantiate the amateur for contrastive decoding using three settings (Section 4.2): (i) earlier checkpoints, (ii) smaller models, and (iii) inference-time attention dropout. We report the best setting from each setting in Table 4 and the full results in Table 6

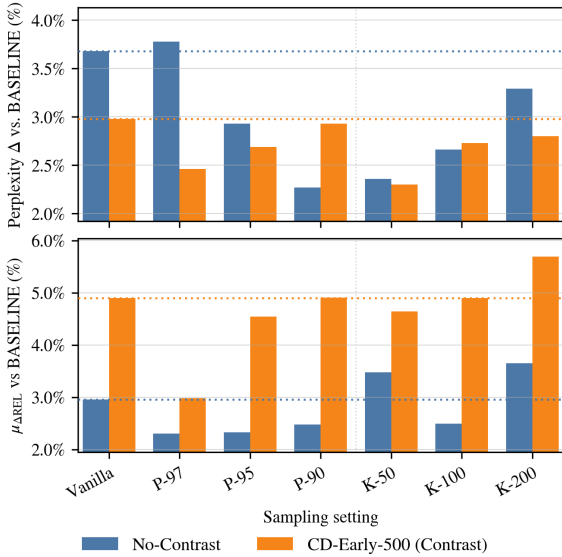


Figure 2: Top- k and top- p truncation under ancestral decoding. “Vanilla” denotes ancestral sampling from unmodified logits after CD or NO-CONTRAST. On downstream tasks, $k=200$ is the strongest setting; perplexity exhibits no single optimum. Full results in Table 6.

in the Appendix. While all CD versions give some boost in performance, using an earlier checkpoint gives the strongest signal.

6.4 Effect of truncation

Across both non-contrastive and contrastive generators, truncation yields at most modest gains, see Figure 2 and Table 4, for the full sweep Table 6. Benefits are largest for Top- k with $k = 200$; nucleus truncation is less reliable.

CD-EARLY-500-TOP-K-200 attains the best aggregate improvement, increasing $\mu_{\Delta REL}$ to 5.69% (vs. 4.90% for CD-EARLY-500) at essentially unchanged perplexity (23.77 vs. 23.73). Slightly tighter truncation with CD-EARLY-500-TOP-K-100 delivers the strongest *Entity Tracking* (+19.02%) and the best *EWoK* (+1.44%), indicating that modest tail pruning can amplify the contrastive signal, with small trade-offs on *Reading Alignment* and *WUG*.

For NO-CONTRAST, nucleus truncation marginally improves perplexity but reduces $\mu_{\Delta REL}$. In contrast, NO-CONTRAST-TOP-K-200 raises $\mu_{\Delta REL}$ to 3.65% while reducing perplexity.

Within our (limited) sweep, truncation can provide additional headroom, especially for contrastive decoding. Light Top- k ($k \in [100, 200]$) appears to preserve diversity while reinforcing preferences for higher-signal tokens.

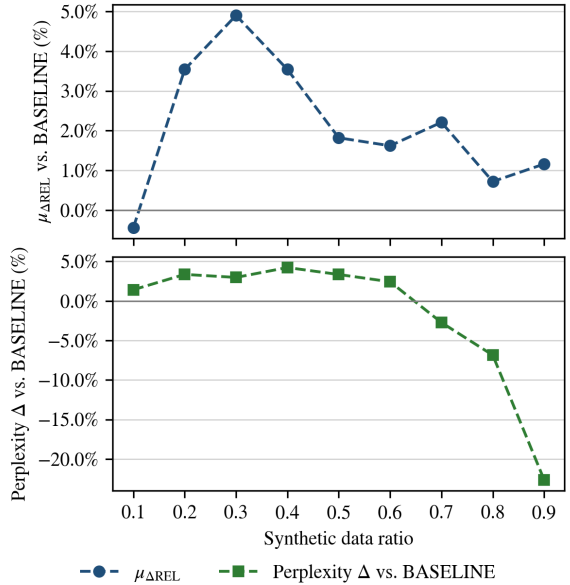


Figure 3: Mixing ratio ablation for CD-generated synthetic corpora (CD-Early-500), also see in Table 6. The ratio indicates the fraction of synthetic data in training batches. $\mu_{\Delta REL}$ is the mean relative improvement over BASELINE across non-perplexity tasks; Perplexity shows relative change vs. BASELINE; A 30% mix yields the best overall $\mu_{\Delta REL}$ (+4.90%), while 40% attains the lowest perplexity (23.42).

6.5 Mixing Ratio Ablation

We analyze what proportion of the original and CD-generated data is most beneficial by varying their ratio. The results can be seen in Figure 3. Note that all corpora were generated with the CD-Early-500 setting. The full result are shown in the Appendix in Table 6. A ratio of 30% synthetic data performs best. Interestingly, similar ratios have shown to perform well when including semi-synthetic data in machine translation using back-translations (Fadaee and Monz, 2018; Simonarson et al., 2021).

7 Discussion

This work asks whether inference-time CD can be repurposed as a *corpus generator* for improving pre-training of language-models. Three findings stand out.

Mixing synthetic data helps; CD helps most where reasoning is required. Across the BabyLM suite, adding any synthetic corpus to TinyBabyLM improves over the BASELINE trained only on real text (Table 2). Among generators, CD delivers the strongest aggregate gains ($\mu_{\Delta REL}$

+4.90% for standard sampling and +5.69% using top- k) and the clearest advantages on reasoning- and tracking-oriented tasks (BLiMP Supplement, Entity Tracking, EWoK, WUG). By contrast, non-contrastive sampling yields the lowest Perplexity and leads on BLiMP, suggesting it better reinforces core grammatical regularities. Together, these results support a practical division of labor: use CD when downstream targets emphasize multi-step inference, state maintenance, or world knowledge; use vanilla sampling when the objective is to minimize perplexity or to improve core grammaticality. A combined approach could also be considered.

Contrastive scoring, not head masking, is the key ingredient. The NO-CONTRAST- $\mathcal{V}_{\text{HEAD}}$ control, which applies only the α -head mask from the good model, does not replicate CD’s benefits and can even hurt Entity Tracking. This indicates that the subtraction against a worse model is doing the heavy lifting. Intuitively, CD preserves high-plausibility tokens while attenuating those over-predicted by the amateur, reducing topical drift and shallow heuristics that smaller or earlier checkpoints tend to prefer effects that plausibly matter most for reasoning-heavy benchmarks.

A practical amateur: earlier checkpoints are a strong and simple choice. Among amateur families, an earlier checkpoint of the same architecture (CD-EARLY-500) performs best in our sweep (Table 6). This choice is attractive operationally: it requires no additional model training, and produced a non-trivial contrast. Smaller-model amateurs and dropout-only amateurs also work but did not perform as well.

Broader implications. These results suggest that inference-time guidance can be re-purposed into *corpus-level* signal shaping: by subtracting the preferences of a systematically weaker model, the generator appears to skew synthetic text toward trajectories that contain constraints that more relevant for reasoning tasks.

8 Limitations

Scale and budget. All experiments use $\sim 100\text{M}$ -parameter models, a fixed 8k-step budget, and an English-only, curated TinyBabyLM corpus. Findings may not transfer to larger scales, non-English data, or web-scale pre-training.

Amateur choice and hyperparameters. Although multiple amateur families were explored, the sweep is not exhaustive. The strongest setting (EARLY-500) may depend on save frequency, optimizer dynamics, or data order. We kept $\alpha=0.1$ and $\lambda=1$ fixed.

Compute and memory overhead. CD generation requires concurrent access to both expert and amateur models at inference time, roughly doubling activation memory and increasing generation latency. While dropout-based amateurs reduce memory pressure, they did not consistently match the early-checkpoint amateur in our setting.

Distributional narrowing. Head masking constrains support and can reduce lexical diversity; while CD outperformed the head-only control, the mask remains part of the procedure, which may under-represent rare constructions. Effects on long-tail generalization and stylistic diversity were not directly measured.

Safety, bias, and factuality. No human evaluation of safety or factual correctness was conducted, and no targeted bias audits were performed. Although CD can downweight some amateur-preferred artifacts, it may also amplify biases present in the expert. More rigorous filtering and auditing are needed for deployment-facing settings.

Single iteration. We only consider a single iteration of CD, in follow-up work we plan to consider how repeated application of CD scales.

Acknowledgments

Vésteinn Snæbjarnarson is supported by the Pioneer Centre for AI, DNRF grant number P1.

References

- Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, and 8 others. 2024. [Phi-4 Technical Report](#). *arXiv preprint arXiv:2412.08905*.
- Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. [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. ACM.
- Haw-Shiuan Chang, Nanyun Peng, Mohit Bansal, Anil Ramakrishna, and Tagyoung Chung. 2024. [Explaining and Improving Contrastive Decoding by Extrapolating the Probabilities of a Huge and Hypothetical LM](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8503–8526, Miami, Florida, USA. Association for Computational Linguistics.
- Lucas Charpentier, Leshem Choshen, Ryan Cotterell, Mustafa Omer Gul, Michael Hu, Jaap Jumelet, Tal Linzen, Jing Liu, Aaron Mueller, Candace Ross, Raj Sanjay Shah, Alex Warstadt, Ethan Wilcox, and Adina Williams. 2025. [BabyLM Turns 3: Call for papers for the 2025 BabyLM workshop](#). *arXiv preprint arXiv:2502.10645*.
- Ryan Cotterell, Anej Svete, Clara Meister, Tianyu Liu, and Li Du. 2024. [Formal Aspects of Language Modeling](#). *arXiv preprint arXiv:2311.04329*.
- Andrea Gregor De Varda, Marco Marelli, and Simona Amenta. 2023. [Cloze probability, predictability ratings, and computational estimates for 205 English sentences, aligned with existing EEG and reading time data](#). *Behavior Research Methods*, 56(5):5190–5213. Publisher: Springer Science and Business Media LLC.
- Elvis Dohmatob, Yunzhen Feng, Arjun Subramonian, and Julia Kempe. 2025. [Strong Model Collapse](#). In *The Thirteenth International Conference on Learning Representations*.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, and 82 others. 2024. [The Llama 3 Herd of Models](#). *arXiv preprint arXiv:2407.21783*.
- Ronen Eldan and Yuanzhi Li. 2023. [TinyStories: How Small Can Language Models Be and Still Speak Coherent English?](#) *arXiv preprint arXiv:2305.07759*.
- Marzieh Fadaee and Christof Monz. 2018. [Back-Translation Sampling by Targeting Difficult Words in Neural Machine Translation](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 436–446, Brussels, Belgium. Association for Computational Linguistics.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2018. [Hierarchical Neural Story Generation](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.
- Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K. Ahmed. 2024. [Bias and Fairness in Large Language Models: A Survey](#). *Computational Linguistics*, 50(3):1097–1179.
- Martin Gerlach and Francesc Font-Clos. 2020. [A Standardized Project Gutenberg Corpus for Statistical Analysis of Natural Language and Quantitative Linguistics](#). *Entropy*, 22(1).
- Matthias Gerstgrasser, Rylan Schaeffer, Apratim Dey, Rafael Rafailov, Tomasz Korbak, Henry Sleight, Rajashree Agrawal, John Hughes, Dhruv Bhandarkar Pai, Andrey Gromov, Dan Roberts, Diyi Yang, David L. Donoho, and Sanmi Koyejo. 2024. [Is Model Collapse Inevitable? Breaking the Curse of Recursion by Accumulating Real and Synthetic Data](#). In *First Conference on Language Modeling*. Conference on Language Modeling (COLM).
- Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. 2023. [Textbooks Are All You Need](#). *arXiv preprint arXiv:2306.11644*.
- Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, and 3 others. 2022. [Training Compute-Optimal Large Language Models](#). In *Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22*, Red Hook, NY, USA. Curran Associates Inc.
- Valentin Hofmann, Leonie Weissweiler, David R. Mortensen, Hinrich Schütze, and Janet B. Pierrehumbert. 2025. [Derivational Morphology Reveals Analogical Generalization in Large Language Models](#). *Proceedings of the National Academy of Sciences*, 122(19):e2423232122.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. [The Curious Case of Neural Text Degeneration](#). In *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Michael Y. Hu, Aaron Mueller, Candace Ross, Adina Williams, Tal Linzen, Chengxu Zhuang, Ryan Cotterell, Leshem Choshen, Alex Warstadt, and Ethan Gotlieb Wilcox. 2024. [Findings of the second BabyLM challenge: Sample-efficient pretraining on developmentally plausible corpora](#). In *The 2nd BabyLM Challenge at the 28th Conference on Computational Natural Language Learning*, pages 1–21, Miami, FL, USA. Association for Computational Linguistics.

- Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. [A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions](#). *ACM Trans. Inf. Syst.*, 43(2).
- Anna A. Ivanova, Aalok Sathe, Benjamin Lipkin, Unnathi Kumar, Setayesh Radkani, Thomas Hikaru Clark, Carina Kauf, Jennifer Hu, R. T. Pramod, Gabriel Grand, Vivian C. Paulun, Maria Ryskina, Ekin Akyürek, Ethan Wilcox, Nafisa Rashid, Leshem Choshen, Roger Levy, Evelina Fedorenko, Joshua B. Tenenbaum, and Jacob Andreas. 2024. [Elements of World Knowledge \(EWOK\): A cognition-inspired framework for evaluating basic world knowledge in language models](#). *arXiv preprint arXiv:2405.09605*.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. [Scaling Laws for Neural Language Models](#). *arXiv preprint arXiv:2505.19371v1*.
- Najoung Kim and Sebastian Schuster. 2023. [Entity Tracking in Language Models](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3835–3855, Toronto, Canada. Association for Computational Linguistics.
- Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023. [Contrastive Decoding: Open-ended Text Generation as Optimization](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12286–12312, Toronto, Canada. Association for Computational Linguistics.
- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. [TruthfulQA: Measuring How Models Mimic Human Falsehoods](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.
- Pierre Lison and Jörg Tiedemann. 2016. [OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).
- Sean O’Brien and Mike Lewis. 2023. [Contrastive Decoding Improves Reasoning in Large Language Models](#). *arXiv preprint arXiv:2309.09117*.
- Charles O’Neill, Yuan-Sen Ting, Ioana Ciucu, Jack Miller, and Thang Bui. 2023. [Steering Language Generation: Harnessing Contrastive Expert Guidance and Negative Prompting for Coherent and Diverse Synthetic Data Generation](#). *arXiv preprint arXiv:2308.07645*.
- Phuc Phan, Hieu Tran, and Long Phan. 2024. [Distillation Contrastive Decoding: Improving LLMs Reasoning with Contrastive Decoding and Distillation](#). *arXiv preprint arXiv:2402.14874*.
- Iliia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. 2024. [AI models collapse when trained on recursively generated data](#). *Nature*, 631(8022):755–759.
- Haukur Barri Símonarson, Vésteinn Snæbjarnarson, Pétur Orri Ragnarson, Haukur Jónsson, and Vilhjálmur Thorsteinnsson. 2021. [Miðeind’s WMT 2021 submission](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 136–139, Online. Association for Computational Linguistics.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023. [Llama 2: Open Foundation and Fine-Tuned Chat Models](#). *arXiv preprint arXiv:2307.09288*.
- Pablo Villalobos, Anson Ho, Jaime Sevilla, Tamay Besiroglu, Lennart Heim, and Marius Hobbhahn. 2024. [Position: Will we run out of data? Limits of LLM scaling based on human-generated data](#). In *Forty-first International Conference on Machine Learning*.
- Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. [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. Association for Computational Linguistics.
- Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Bhargavi Paranjabe, Adina Williams, Tal Linzen, and Ryan Cotterell. 2023. [Findings of the BabyLM Challenge: Sample-Efficient Pretraining on Developmentally Plausible Corpora](#). In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 1–6, Singapore. Association for Computational Linguistics.
- Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. [BLiMP: The benchmark of linguistic minimal pairs for English](#). *Transactions of the Association for Computational Linguistics*, 8:377–392.
- Zheng Zhao, Emilio Monti, Jens Lehmann, and Haytham Assem. 2024. [Enhancing Contextual Understanding in Large Language Models through Contrastive Decoding](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Asso-*

ciation for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 4225–4237, Mexico City, Mexico. Association for Computational Linguistics.

Appendix

A Model & Tokenizer Training Details

Model Details The architecture we use is a LLaMA-2–style decoder-only Transformer from Touvron et al. (2023) with name=llama-12-768: 12 layers, hidden size 768, 12 attention heads, MLP intermediate size 3072, and maximum context length 2048 tokens. All models use dtype=float32 and the same tokenizer configuration.

Tokenizer. We use a SentencePiece BPE tokenizer (vocabulary size 32,000) trained on the Tiny-BabyLM corpus. Preprocessing follows SentencePiece defaults, including Unicode normalization, whitespace deduplication, and removal of control characters. The identical tokenizer is used across all experiments to ensure comparability.

Training Details, Data Mixing & Regrouping Regularizer. Per-device batches contain 16 sequences of length 1,024 tokens; with 4 GPUs and gradient accumulation of 4, the effective global batch is $256 \times 1,024$ tokens.

For training with a (70% real, 30% synthetic) mixture for each batch, the 256 sequences are sampled from the original/synthetic corpus accordingly to satisfy the required ratio at a sequence-ratio level. To implement this mixture, the real and synthetic corpora are stored as rows of text and, at the start of training, each corpus is independently shuffled, tokenized, and split into fixed-length sequences. Sequences are then sampled until one corpus is exhausted; the exhausted corpus is reshuffled, re-tokenized, and re-split before sampling resumes. This periodic resegmentation acts as a light regularizer by continually refreshing ordering and boundaries, and we apply the identical procedure in the BASELINE runs for parity

We train with the causal language modeling objective (next-token prediction), minimizing token-level cross-entropy (negative log-likelihood). Optimization uses AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay 0.1, and initial learning rate $1e-3$. The schedule is cosine with 150 warm-up steps, decaying to zero by step 8,000. All runs are executed on a multi-GPU cluster with NVIDIA RTX 3090 or RTX 4090 GPUs.

Table 5: Architectures used for the good and bad models. All models share the same tokenizer and max position embeddings (1024). The suffix in name (e.g., 5x, 10x) indicates the intended scale relative to the expert.

Name	Layers	Hidden	Heads	Intermediate	Max pos
llama-12-768 (GOOD)	12	768	12	3072	1024
llama-10-512-5x	10	512	8	2048	1024
llama-8-384-10x	8	384	6	1536	1024
llama-6-256-20x	6	256	4	1024	1024
llama-5-224-50x	5	224	4	896	1024
llama-4-192-100x	4	192	3	768	1024

B Synthetic Generation Details

Framework and hardware. Synthetic text is produced with a custom, PyTorch generation loop designed for efficiency and flexibility. The loop supports multi-GPU parallelization, per-token logit transforms (for contrastive decoding), and caching. All generation runs on the same multi-GPU cluster used for training, typically $4 \times$ NVIDIA RTX 3090/4090 GPUs.

C All Task Results

We give a comprehensive overview of model performance in Table 6.

Name	$\mu_{\Delta REL} \uparrow$	Perplexity \downarrow	BLiMP \uparrow	BLiMP Supp. \uparrow	Entity Tracking \uparrow
BASELINE	-	24.46 \pm 0.10	71.03 \pm 0.27	64.10 \pm 0.60	27.82 \pm 1.18
No-Contrast-MR-0.3	2.96%	23.56 \pm 0.11* (3.68%)	72.09 \pm 0.17* (1.50%)	64.83 \pm 0.73 (1.15%)	28.14 \pm 1.75 (1.16%)
No-Contrast-Top-K-100-MR-0.3	2.49%	23.81 \pm 0.11* (2.66%)	72.12 \pm 0.26* (1.55%)	64.22 \pm 0.69 (0.19%)	28.09 \pm 1.65 (0.98%)
No-Contrast-Top-K-200-MR-0.3	3.65%	23.65 \pm 0.10* (3.29%)	71.78 \pm 0.21* (1.06%)	63.98 \pm 0.69 (-0.19%)	26.96 \pm 1.23* (-3.08%)
No-Contrast-Top-K-50-MR-0.3	3.48%	23.88 \pm 0.10* (2.36%)	71.52 \pm 0.13* (0.69%)	64.45 \pm 0.83 (0.54%)	27.23 \pm 1.64* (-2.13%)
No-Contrast-Top-P-90-MR-0.3	2.73%	23.88 \pm 0.11* (2.37%)	71.96 \pm 0.14* (1.31%)	64.84 \pm 0.62 (1.16%)	26.12 \pm 0.89* (-6.09%)
No-Contrast-Top-P-95-MR-0.3	2.33%	23.74 \pm 0.12* (2.93%)	72.02 \pm 0.22* (1.40%)	64.50 \pm 0.63 (0.63%)	26.29 \pm 1.31* (-5.51%)
No-Contrast-Top-P-97-MR-0.3	2.11%	23.61 \pm 0.10* (3.47%)	71.62 \pm 0.11* (0.83%)	64.33 \pm 0.68 (0.36%)	27.29 \pm 1.48* (-1.91%)
No-Contrast-Top-V-Head-MR-0.3	0.66%	24.33 \pm 0.10* (0.51%)	71.67 \pm 0.24* (0.91%)	64.86 \pm 0.74 (1.20%)	25.47 \pm 1.40* (-8.45%)
CD-Early-100-MR-0.3	2.42%	24.02 \pm 0.11* (1.79%)	71.31 \pm 0.12* (0.40%)	63.54 \pm 0.63 (-0.87%)	26.19 \pm 1.51* (-5.87%)
CD-Early-1500-MR-0.3	4.26%	24.04 \pm 0.14* (1.70%)	71.69 \pm 0.26* (0.94%)	63.92 \pm 0.57 (-0.28%)	27.78 \pm 1.19 (-0.15%)
CD-Early-2000-MR-0.3	2.06%	24.28 \pm 0.16* (0.73%)	71.87 \pm 0.22* (1.19%)	63.82 \pm 0.55 (-0.44%)	27.55 \pm 1.47 (-0.98%)
CD-Drop-0.1-MR-0.3	-1.42%	24.02 \pm 0.10* (1.78%)	71.55 \pm 0.20* (0.74%)	64.39 \pm 0.60 (0.45%)	22.59 \pm 0.93* (-18.80%)
CD-Drop-0.3-MR-0.3	0.99%	24.09 \pm 0.19* (1.52%)	71.39 \pm 0.14* (0.52%)	64.86 \pm 0.64 (1.19%)	24.22 \pm 1.05* (-12.93%)
CD-Drop-0.5-MR-0.3	2.52%	23.94 \pm 0.10* (2.11%)	71.80 \pm 0.28* (1.09%)	64.91 \pm 0.60 (1.27%)	28.72 \pm 1.00* (3.23%)
CD-Drop-0.7-MR-0.3	3.29%	24.06 \pm 0.13* (1.65%)	71.79 \pm 0.31* (1.08%)	65.19 \pm 0.70 (1.71%)	28.91 \pm 1.64* (3.91%)
CD-Small-100-MR-0.3	1.65%	23.81 \pm 0.14* (2.65%)	71.97 \pm 0.27* (1.33%)	64.86 \pm 0.58 (1.19%)	29.59 \pm 1.14* (6.38%)
CD-Small-10-MR-0.3	3.66%	23.86 \pm 0.11* (2.44%)	71.95 \pm 0.22* (1.30%)	64.95 \pm 0.56 (1.33%)	27.68 \pm 1.21 (-0.49%)
CD-Small-20-MR-0.3	3.55%	23.73 \pm 0.14* (2.96%)	71.84 \pm 0.19* (1.15%)	64.09 \pm 0.66 (-0.01%)	29.25 \pm 1.32* (5.15%)
CD-Small-50-MR-0.3	2.30%	23.73 \pm 0.11* (2.97%)	71.97 \pm 0.23* (1.33%)	65.55\pm0.58* (2.27%)	29.28 \pm 1.46* (5.26%)
CD-Small-5-MR-0.3	2.97%	23.97 \pm 0.10* (1.99%)	71.46 \pm 0.11* (0.62%)	63.89 \pm 0.53 (-0.33%)	28.44 \pm 1.03* (2.25%)
CD-Early-500-MR-0.1	-0.44%	24.11 \pm 0.10* (1.42%)	72.11 \pm 0.21* (1.53%)	63.59 \pm 0.49 (-0.79%)	27.22 \pm 1.16* (-2.17%)
CD-Early-500-MR-0.2	3.54%	23.64 \pm 0.10* (3.36%)	72.49\pm0.18* (2.06%)	64.94 \pm 0.57 (1.31%)	31.25 \pm 1.12* (12.34%)
CD-Early-500-MR-0.3	4.90%	23.73 \pm 0.10* (2.98%)	71.72 \pm 0.19* (0.98%)	65.10 \pm 0.60* (1.56%)	30.38 \pm 0.65* (9.19%)
CD-Early-500-MR-0.4	3.54%	23.42\pm0.13* (4.23%)	70.90 \pm 0.21 (-0.17%)	63.69 \pm 0.55 (-0.63%)	33.30\pm0.84* (19.70%)
CD-Early-500-MR-0.5	1.82%	23.64 \pm 0.16* (3.35%)	69.46 \pm 0.20* (-2.21%)	62.84 \pm 0.61* (-1.96%)	28.68 \pm 1.31* (3.09%)
CD-Early-500-MR-0.6	1.62%	23.86 \pm 0.09* (2.43%)	68.91 \pm 0.21* (-2.98%)	62.30 \pm 0.58* (-2.81%)	30.45 \pm 1.09* (9.47%)
CD-Early-500-MR-0.7	2.21%	25.13 \pm 0.12* (-2.73%)	68.18 \pm 0.19* (-4.00%)	62.42 \pm 0.67* (-2.62%)	31.01 \pm 1.10* (11.48%)
CD-Early-500-MR-0.8	0.72%	26.14 \pm 0.11* (-6.86%)	67.42 \pm 0.25* (-5.07%)	61.30 \pm 0.82* (-4.36%)	30.57 \pm 0.60* (9.89%)
CD-Early-500-MR-0.9	1.16%	30.00 \pm 0.12* (-22.64%)	66.50 \pm 0.25* (-6.38%)	59.86 \pm 0.79* (-6.62%)	31.76 \pm 1.11* (14.18%)
CD-Early-500-Top-K-100-MR-0.3	4.90%	23.79 \pm 0.12* (2.73%)	71.49 \pm 0.18* (0.65%)	65.29 \pm 0.80* (1.87%)	33.11 \pm 0.62* (19.02%)
CD-Early-500-Top-K-200-MR-0.3	5.69%	23.77 \pm 0.10* (2.80%)	71.87 \pm 0.35* (1.19%)	64.23 \pm 0.59 (0.20%)	31.05 \pm 0.79* (11.61%)
CD-Early-500-Top-K-50-MR-0.3	4.64%	23.90 \pm 0.12* (2.30%)	71.90 \pm 0.21* (1.23%)	64.74 \pm 0.68 (1.01%)	30.29 \pm 1.49* (8.89%)
CD-Early-500-Top-P-90-MR-0.3	4.91%	23.74 \pm 0.10* (2.93%)	72.16 \pm 0.14* (1.60%)	64.69 \pm 0.65 (0.92%)	30.43 \pm 1.07* (9.37%)
CD-Early-500-Top-P-95-MR-0.3	4.54%	23.80 \pm 0.15* (2.69%)	71.36 \pm 0.27* (0.47%)	64.79 \pm 0.62 (1.09%)	32.56 \pm 0.74* (17.06%)
CD-Early-500-Top-P-97-MR-0.3	2.98%	23.86 \pm 0.13* (2.46%)	71.69 \pm 0.20* (0.94%)	64.54 \pm 0.56 (0.69%)	30.20 \pm 0.92* (8.56%)
Name	$\mu_{\Delta REL} \uparrow$	EWoK \uparrow	WUG \uparrow	Reading \uparrow	Eye Tracking \uparrow
BASELINE	-	53.18 \pm 0.28	66.90 \pm 2.47	1.76 \pm 0.22	3.85 \pm 0.31
No-Contrast-MR-0.3	2.96%	53.17 \pm 0.30 (-0.01%)	64.67 \pm 1.66* (-3.34%)	1.91 \pm 0.25 (8.34%)	4.31 \pm 0.33* (11.92%)
No-Contrast-Top-K-100-MR-0.3	2.49%	53.43 \pm 0.32 (0.48%)	66.71 \pm 2.15 (-0.28%)	1.85 \pm 0.27 (4.65%)	4.23 \pm 0.38 (9.86%)
No-Contrast-Top-K-200-MR-0.3	3.65%	53.52 \pm 0.32 (0.64%)	67.81 \pm 1.53 (1.36%)	1.96 \pm 0.26 (10.76%)	4.43 \pm 0.35* (15.01%)
No-Contrast-Top-K-50-MR-0.3	3.48%	53.37 \pm 0.30 (0.37%)	67.38 \pm 1.82 (0.71%)	1.97 \pm 0.27 (11.83%)	4.33 \pm 0.36* (12.38%)
No-Contrast-Top-P-90-MR-0.3	2.73%	53.36 \pm 0.27 (0.35%)	66.25 \pm 2.05 (-0.97%)	1.94 \pm 0.23 (9.92%)	4.37 \pm 0.32* (13.44%)
No-Contrast-Top-P-95-MR-0.3	2.33%	53.41 \pm 0.32 (0.45%)	66.44 \pm 1.52 (-0.69%)	1.90 \pm 0.26 (7.51%)	4.33 \pm 0.35* (12.51%)
No-Contrast-Top-P-97-MR-0.3	2.11%	53.24 \pm 0.28 (0.12%)	66.00 \pm 1.63 (-1.35%)	1.87 \pm 0.24 (6.20%)	4.26 \pm 0.32* (10.51%)
No-Contrast-Top-V-Head-MR-0.3	0.66%	53.03 \pm 0.31 (-0.27%)	66.67 \pm 1.58 (-0.35%)	1.76 \pm 0.23 (-0.54%)	4.32 \pm 0.33* (12.12%)
CD-Early-100-MR-0.3	2.42%	53.19 \pm 0.30 (0.03%)	66.83 \pm 1.58 (-0.10%)	1.89 \pm 0.25 (7.02%)	4.48 \pm 0.34* (16.30%)
CD-Early-1500-MR-0.3	4.26%	53.61 \pm 0.31 (0.81%)	67.89 \pm 2.26 (1.48%)	2.03\pm0.26 (14.95%)	4.32 \pm 0.34* (12.09%)
CD-Early-2000-MR-0.3	2.06%	53.30 \pm 0.29 (0.23%)	68.67 \pm 1.67 (2.64%)	1.80 \pm 0.24 (2.23%)	4.22 \pm 0.35 (9.55%)
CD-Drop-0.1-MR-0.3	-1.42%	53.11 \pm 0.29 (-0.12%)	65.33 \pm 2.11 (-2.34%)	1.81 \pm 0.24 (2.80%)	4.14 \pm 0.32 (7.36%)
CD-Drop-0.3-MR-0.3	0.99%	53.43 \pm 0.31 (0.48%)	67.28 \pm 1.37 (0.56%)	1.91 \pm 0.26 (8.15%)	4.20 \pm 0.33 (8.95%)
CD-Drop-0.5-MR-0.3	2.52%	53.17 \pm 0.33 (-0.00%)	68.28 \pm 1.74 (2.06%)	1.75 \pm 0.24 (-0.72%)	4.27 \pm 0.33 (10.74%)
CD-Drop-0.7-MR-0.3	3.29%	53.62 \pm 0.40 (0.83%)	66.80 \pm 1.72 (-0.15%)	1.90 \pm 0.35 (7.76%)	4.16 \pm 0.44 (7.92%)
CD-Small-100-MR-0.3	1.65%	53.36 \pm 0.28 (0.34%)	66.20 \pm 1.61 (-1.05%)	1.68 \pm 0.21 (-4.65%)	4.16 \pm 0.31 (7.99%)
CD-Small-10-MR-0.3	3.66%	53.50 \pm 0.32 (0.61%)	68.80 \pm 2.24 (2.84%)	1.93 \pm 0.24 (9.12%)	4.27 \pm 0.31* (10.87%)
CD-Small-20-MR-0.3	3.55%	53.45 \pm 0.27 (0.50%)	69.05 \pm 2.53 (3.21%)	1.79 \pm 0.22 (1.59%)	4.37 \pm 0.31* (13.29%)
CD-Small-50-MR-0.3	2.30%	53.29 \pm 0.28 (0.20%)	66.45 \pm 1.54 (-0.67%)	1.78 \pm 0.23 (1.13%)	4.10 \pm 0.31 (6.54%)
CD-Small-5-MR-0.3	2.97%	53.23 \pm 0.30 (0.09%)	67.40 \pm 1.37 (0.75%)	1.83 \pm 0.22 (3.74%)	4.38 \pm 0.32* (13.68%)
CD-Early-500-MR-0.1	-0.44%	53.45 \pm 0.33 (0.51%)	66.10 \pm 1.44 (-1.20%)	1.69 \pm 0.22 (-4.08%)	3.97 \pm 0.32 (3.09%)
CD-Early-500-MR-0.2	3.54%	53.53 \pm 0.27 (0.66%)	66.35 \pm 1.48 (-0.82%)	1.78 \pm 0.23 (0.79%)	4.18 \pm 0.31 (8.41%)
CD-Early-500-MR-0.3	4.90%	53.80 \pm 0.29* (1.18%)	70.55\pm2.32* (5.46%)	1.79 \pm 0.22 (1.30%)	4.42 \pm 0.32* (14.64%)
CD-Early-500-MR-0.4	3.54%	53.41 \pm 0.28 (0.44%)	66.50 \pm 1.87 (-0.60%)	1.74 \pm 0.23 (-1.19%)	4.13 \pm 0.31 (7.27%)
CD-Early-500-MR-0.5	1.82%	53.36 \pm 0.29 (0.34%)	67.00 \pm 1.76 (0.15%)	1.77 \pm 0.22 (0.34%)	4.35 \pm 0.32 (12.98%)
CD-Early-500-MR-0.6	1.62%	53.04 \pm 0.31 (-0.25%)	68.35 \pm 1.89 (2.17%)	1.69 \pm 0.22 (-4.36%)	4.24 \pm 0.32 (10.10%)
CD-Early-500-MR-0.7	2.21%	52.91 \pm 0.28 (-0.50%)	64.75 \pm 1.85 (-3.21%)	1.82 \pm 0.23 (2.95%)	4.29 \pm 0.34 (11.37%)
CD-Early-500-MR-0.8	0.72%	52.73 \pm 0.31 (-0.85%)	64.28 \pm 1.89* (-3.92%)	1.80 \pm 0.24 (1.79%)	4.15 \pm 0.32 (7.59%)
CD-Early-500-MR-0.9	1.16%	52.57 \pm 0.31 (-1.13%)	65.06 \pm 1.23 (-2.76%)	1.79 \pm 0.25 (1.35%)	4.22 \pm 0.34 (9.50%)
CD-Early-500-Top-K-100-MR-0.3	4.90%	53.94\pm0.36* (1.44%)	67.44 \pm 2.13 (0.80%)	1.73 \pm 0.26 (-2.20%)	4.34 \pm 0.35* (12.74%)
CD-Early-500-Top-K-200-MR-0.3	5.69%	53.61 \pm 0.31 (0.82%)	67.90 \pm 2.00 (1.49%)	1.92 \pm 0.25 (8.73%)	4.46 \pm 0.33* (15.78%)
CD-Early-500-Top-K-50-MR-0.3	4.64%	53.47 \pm 0.32 (0.55%)	67.56 \pm 2.48 (0.99%)	1.93 \pm 0.26 (9.49%)	4.25 \pm 0.35 (10.30%)
CD-Early-500-Top-P-90-MR-0.3	4.91%	53.68 \pm 0.30 (0.94%)	68.35 \pm 1.44 (2.17%)	1.85 \pm 0.23 (4.59%)	4.42 \pm 0.33* (14.77%)
CD-Early-500-Top-P-95-MR-0.3	4.54%	53.60 \pm 0.29 (0.80%)	65.78 \pm 1.99 (-1.68%)	1.82 \pm 0.24 (2.86%)	4.28 \pm 0.33 (11.14%)
CD-Early-500-Top-P-97-MR-0.3	2.98%	53.63 \pm 0.30 (0.86%)	64.85 \pm 1.50 (-3.06%)	1.82 \pm 0.22 (3.06%)	4.23 \pm 0.31 (9.84%)

Table 6: Full sweep of all experiments. Naming scheme: NO-CONTRAST = ancestral sampling from p_G ; NO-CONTRAST- $\mathcal{V}_{\text{HEAD}}$ = ancestral sampling restricted to the α -head $\mathcal{V}_{\text{head}}(\cdot)$ of p_G ; CD-EARLY- k = contrastive decoding with the amateur p_B taken as an earlier training checkpoint at step k ; CD-SMALL- r = p_B is a smaller model (about $r \times$ fewer parameters than p_G); CD-DROP- p = p_G run with attention dropout rate p at inference; CD-SYNTH-RATIO- q = training mixture uses synthetic fraction q . G2500 denotes the fixed GOOD checkpoint used for generation (selected at training step 2500). Other conventions follow the universal caption: means \pm s.e.; * indicates significance vs. BASELINE; parentheses give relative change vs. BASELINE; $\mu_{\Delta REL}$ averages non-perplexity tasks; Reading/Eye Tracking values are the % increase in variance explained after adding the LM features.