Optimizing GPT-2 Pretraining on BabyLM Corpus with Difficulty-based Sentence Reordering

Nasim Borazjanizadeh
Williams College
nb11@williams.edu

Abstract

This paper focuses on enhancing the performance of GPT-2, pre-trained on the BabyLM Strict-Small challenge datasets, for the BLiMP zero-shot tasks. We explored various curriculum learning optimizations to supervise the order of training samples presented to the model. We discovered that training GPT-2 on a corpus consisting of one dataset sorted based on difficulty leads to improved BLiMP scores. Additionally, we measured the loss of contextual information by comparing the semantic similarity of neighboring sentences before and after reordering inputs of each dataset. A positive correlation is found between the measured contextual similarity of sentences in the difficulty-sorted dataset and the BLiMP performance of the model trained on the rearranged dataset. We conclude that reordering sentences based on difficulty while minimizing the loss of contextual and semantic similarity between sentences that follow each other in a context length can enhance the model's performance. Using this approach we trained a model with an average of 75.77% across all BLiMP's tasks. Additionally, data cleaning using ASR further enhanced the model performance on BLiMP to 75.84%, an improvement of over 6% compared to the baselines released for the BabyLM Strict-Small challenge.

1 Introduction

Language models have shown significant progress in natural language processing tasks, but their performance heavily relies on the diversity and quality of large-scale training data. This paper aims to enhance the performance of language models trained exclusively on the datasets from the BabyLM Strict-Small challenge (Warstadt et al., 2023). We evaluate the models using the average across all BLiMP's zero-shot tasks, which assess language models' knowledge of major English grammatical phenomena (Warstadt et al., 2020). The reason we exclusively relied on BLiMP results to optimize the performance of our models is that other evaluation tasks within the BabyLM evaluation pipeline, like (Super)GLUE and held-out MSGS tasks, require fine-tuning the model and demand more computational resources than we had available.

In this paper, we attempt to optimize the performance of language models on the BLiMP evaluation by using heuristics inspired by difficulty metrics proposed in Competence-based Curriculum Learning (Platanios et al., 2019) to reorder sentences in the datasets and remove semantically meaningless inputs. As a result, we achieved an improvement of over 6 percent on BLiMP compared to the baseline results released for the BabyLM Strict-Small track (Table 1).

We first manually analyzed the training data to gain a better understanding of the training data. The analysis revealed the sentences in the `gutenberg` dataset were fragmented across lines. This fragmentation could disrupt the intrinsic structure and the contextual information provided by each sentence during training, as irrelevant fragments would follow each other in a single context length due to the shuffling of training samples at each training epoch. To rectify this, we preprocessed the `gutenberg` dataset by merging subsegments of each sentence into a coherent sentence printed in one line (Table 1).

We then attempted to optimize the use of limited training samples by supervising the order of samples presented to the model using Curriculum Learning (CL) and Competence-based Curriculum Learning. These methods involve starting the training of the model with simpler examples and gradually introducing harder ones. In Competence-based CL, the training corpus is constructed using the competence function which samples from the difficulty-sorted training inputs based on the competence of the model at time `t` compared to

Proceedings of the 27th Conference on Computational Natural Language Learning: Volume 2: The BabyLM Challenge, pages 356–365

Baselines	BLiMP
OPT-125 - BabyLM baseline	62.63%
RoBERTa-base - BabyLM baseline	69.47%
T5-base - BabyLM baseline	58.83%
GPT-2 - gutenberg not merged	73.40%
GPT-2 - gutenberg merged	75.05%

Table 1: The BLiMP evaluation results comparing the baselines released for the BabyLM Strict-Small challenge and our baseline GPT-2 models. GPT-2 gutenberg not merged is trained on all raw datasets in the Strict-Small track, and GPT-2 - gutenberg merged model is trained on 9 unchanged datasets and the preprocessed gutenberg dataset, where sentences are merged into a single line.

Excluded	BLiMP	Excluded	BLiMP
aochildes	74.40%	open_subtitles	72.81%
bnc_spoken	73.91%	qed	74.12%
cbt	74.21%	simple_wikipedia	73.64%
children_stories	73.36%	switchboard	73.93%
gutenberg	73.48%	wikipedia	72.70%

Table 2: BLiMP evaluation results for GPT-2 model trained on all datasets in Strict-Small track beside the dataset listed under the 'Excluded' column. The sentences in the gutenberg dataset are merged into one line, and thus the baseline model for this experiment is GPT-2 - gutenberg merged with a BLiMP score of 75.05%.

the competence of the model at convergence time. Using this method leads to an overrepresentation of shorter sentences (which are sorted as easier using length-dependent difficulty metrics such as sentence length (SL) or sentence rarity (SR) suggested in Platanios et al. (2019)) in the training corpus. Shorter sentences tend to contain more grammatical errors in the BabyLM datasets, as these datasets consist largely of spoken language sentences. We hypothesize this could result in suboptimal results on BLiMP when implementing Competence-based CL optimizations. To address this, we proposed a novel length-independent difficulty metric, *average sentence rarity* (ASR), calculated by taking the average frequency of words in a sentence to determine the singular score for the difficulty of the sentence.

We hypothesize that when using CL optimizations, the performance of the model is also negatively impacted because the contextual information provided by neighboring sentences is disrupted when reordering sentences based on difficulty. To tackle the loss of contextual information, we narrow our focus to a smaller optimization problem, supervising the order of sentences within a context length rather than the order of all sentences in the training corpus, as determined by the competence function. To measure contextual information provided by nearby sentences, we propose a new heuristic, *local coherence*, calculated by quantifying the similarity between a central sentence and its adjacent ones using sup-simcse-roberta-large model (Gao et al., 2021) within a specific window of seven inputs. The size of this window is determined by the average number of samples combined into a context length after tokenization.

To enhance the model's performance and investigate our hypothesis about the correlation between the local coherence of sentences in the datasets sorted based on difficulty and the resulting improvement in the language model's performance on BLiMP, we conducted a series of 20 experiments. In these experiments, we exclusively reordered sentences from one dataset based on SR or ASR, while leaving the other 9 datasets unchanged. Upon analyzing the results, we found a positive correlation between the expected local coherence of the sorted datasets and the BLiMP performance of the models trained on the corpus compromising of one sorted dataset. The positive correlations indicate that reordering sentences based on difficulty while minimizing the loss of contextual and semantic similarity between sentences that follow each other in a context length can enhance the model's performance.

We were also able to improve the model's performance with data cleaning. ASR sorts inputs with high counts of frequent words and low counts of other words, as easy inputs. Through manual evaluation of datasets, we discovered that these characteristics often correspond to meaningless or grammatically incoherent inputs in the 3 following datasets: cbt, gutenberg, and bnc_spoken. Removing these redundant inputs from the gutenberg dataset, led to improved BLiMP performances for the model trained on the sorted and cleaned dataset (Table 6).

While we did achieve improvements in the BLiMP evaluation by training models only on the Strict-Small datasets using the described methods, the most significant intellectual contribution of this paper is highlighting the importance of considering contextual and knowledge-based similarity when reordering training inputs with any performance-

enhancing metrics for language models. This concept reflects how humans learn effectively. In schools, subjects like math and English are not mixed in the same class period, regardless of the difficulty of the subjects, unless students are already proficient in both.

If we compare a context length-sized input to a human's attention span of 5 minutes, teaching math to a model or human is more effective if we present 10 similar examples of arithmetic operations that follow the same logical pattern within that 5-minute span, rather than presenting examples of different mathematical operations (like basic combinatorics mixed with calculus and geometry) without any logical pattern connecting the examples, even if these examples share the same level of difficulty. Therefore, in curriculum learning for language models, we argue that sentences that are presented together within a context length should be semantically and contextually similar beyond having the same level of difficulty.

2 Model Architecture & Training Loop

To identify the optimal base model architecture for our experiments, we trained BERT(Devlin et al., 2018), RoBERTa(Liu et al., 2019), and GPT-2(Radford et al., 2019) on the given datasets, adhering to the conventional guidelines for language model training, and utilizing identical hyperparameters, without any extra optimizations. Our results revealed that GPT-2 not only converged at a faster rate but also marginally outperformed the other models in the BLiMP evaluation. Consequently, we selected GPT-2 as the base model architecture for all our following experiments.

The GPT2 models were trained for six epochs, with convergence typically occurring around the fifth epoch. Throughout the training process, we assessed the models on the evaluation dataset every 500 steps, with the gradient accumulation set to 1. We then selected the best checkpoint based on the evaluation loss to assess with BLiMP evaluation. For all experiments, we utilized the DataLoader function to load data in batches of size 64. We set the shuffle boolean to True, which rearranges the indices of all samples at each epoch for the baseline experiments and the ablation experiments (results in Table 2) that did not involve reordering the data.

The data preparation process involved reading each line of the dataset files as a separate sample. We then joined all the tokenized samples in a batch with an `eos_token_id` token in between and then divided the concatenated samples into sequences of size context-length. During experiments that involved sorting the sentences based on difficulty, we eliminated any duplicated inputs from the dataset. We used the preprocessed `gutenberg` dataset, with sentence fragments merged into one line, as our baseline `gutenberg` dataset for all the experiments besides GPT-2 `gutenberg` not merged (Table 1).

In our experimental setup, we tested our baseline model using different context length sizes. We observed that a context length of 64 resulted in a decline in the model's performance on BLiMP. On the other hand, context lengths of 512 and 256 did not yield any performance improvements over a context length of 128. However, they significantly increased the GPU memory usage and extended the training time. Consequently, we chose a context length of 128, the smallest size that did not adversely affect the model's performance, for all subsequent experiments.

We repeated a subset of baseline experiments multiple times to understand the effect of randomness on the outcome of experiments. The limited volume of data used to train our models introduces an inherent instability in the training process, resulting in some variation in the BLiMP evaluation results. We observed a variance of up to 0.6% in the experiments with the same setup when altering the seed before instantiating the model. To neutralize the randomness effect and ensure a valid comparison of different optimizations, we standardized the seed value to 1 for all the experiments discussed in this paper.

3 Dataset Analysis

In order to gain a better understanding of the training data, we conducted a manual analysis of the datasets. This examination revealed that the sentences in the Gutenberg dataset were fragmented across multiple lines. Given that each line is read as a separate sample in our baseline training loop, shuffling the sample indices results in unrelated sentence segments following one another in a context length. This disrupts the inherent structure of the sentences and interrupts the contextual information provided by the surrounding words when learning word embeddings during training.

To address this issue, we preprocessed the Gutenberg dataset by consolidating subsegments of each sentence into a single, coherent sentence printed

in one line. This modification led to an improvement of over 1.6% percent in the model's BLiMP evaluation results compared to the baseline (Table 1). This notable enhancement over the baseline, achieved through a straightforward preprocessing step, highlights the importance of maintaining the contextual information provided by the surrounding sentences when feeding the training data to the model.

To evaluate the influence of each dataset on the model's performance during the BLiMP assessment, we conducted an ablation study consisting of 10 experiments. In each of these experiments, the model was trained on nine datasets, with one dataset being excluded in each iteration (Table 2). The results show that removing the `aochildes` dataset has the least influence on the model's performance. However, excluding the `wikipedia` dataset significantly reduced the model's BLiMP score. A comparison between sentences in the `aochildes` and `wikipedia` datasets highlights their distinct grammatical characteristics. Sentences in the `aochildes` dataset, which are compiled from child-directed speech (Huebner et al., 2021), are short, informal, and often contain grammatical errors, including missing or misplaced pronouns and verbs. On the other hand, the `wikipedia` dataset contains longer sentences that strictly adhere to grammatical rules while avoiding unnecessary repetition.

As BLiMP is specifically designed to assess the sensitivity of language models to acceptability contrasts using grammar templates (Warstadt et al., 2020), it follows that the impact of excluding spoken language sentences in `aochildes`, which are incomplete and error-prone, on improving the model's performance in BLiMP evaluation is less significant. Additionally, we can observe that shorter sentences in the BabyLM datasets are less effective in training the model for BLiMP evaluation.

4 Curriculum Learning

To optimize the use of the limited training samples available and improve the model's performance, we chose to supervise the order in which samples are presented to the model. To this end, we implemented Curriculum Learning (CL) (Bengio et al., 2009) and Competence-based Curriculum Learning (Platanios et al., 2019). The fundamental idea behind CL is to initiate learning with simpler examples and gradually incorporate harder ones by sorting the samples based on their difficulty. In Competence-based CL, the training data is filtered based on the estimated difficulty of the sample and model competence.

Difficulty Metric	BLiMP
Sentence Length	69.93%
Sentence Rarity	71.49%
Average Sentence Rarity	74.51%

Table 3: BLiMP results for competence-based CL using different difficulty metrics. The `gutenberg` dataset is preprocessed to have complete sentences in each line before reordering the samples based on difficulty. Shuffle is set to false, and the number of training epochs is 1, as the competence function samples from the difficulty-sorted datasets multiple times when constructing the training corpus.

To implement Competence-based CL, we sorted the training samples based on the difficulty metrics outlined in the Platanios et al. (2019): Sentence Length (SL), which ranks samples based on length, considering shorter samples as easier, and Sentence Rarity (SR), which is the overall likelihood of a sentence, incorporating both word frequency and sentence length, with less likely or more rare sentences being considered more difficult. To build the training corpus with a supervised order of samples, we employed the square root competence function which determines which examples should be incorporated into the training corpus, based on the competence of the model at time `t` of training, and the pace at which new examples are introduced during the training process, where the rate of new examples added decreases over time, allowing the learner more time to assimilate the information (Table 3).

However, BLiMP results for models trained using SL or SR difficulty metrics were worse than the performance achieved when training the model on the base datasets (with `gutenberg` sentences merged) without any CL optimizations. We hypothesize that the sub-optimal performance is linked to the competence function's design and the unique attributes of the BabyLM datasets. The competence function samples more from easier sentences when constructing the training corpus, and both SR and SL heuristics employ sentence length as a criterion, either implicitly or explicitly, to determine the difficulty of sentences. Consequently, this leads to an overrepresentation of shorter sentences in the

training data created using this competence function. Furthermore, a higher portion of the BabyLM datasets includes transcribed speech, and shorter spoken language sentences are often fragmented and contain more grammatical errors due to the spontaneous flow of the speech. Our prior observations also show a negative correlation between sentence length and the importance of the sentence in training the model for BLiMP evaluation. Thus, we can deduce that the overrepresentation of short sentences in the Competence-based CL training dataset adversely affects the model's performance on BLiMP.

5 Proposed Methods

5.1 Average Sentence Rarity

Using word frequencies as a difficulty heuristic can be helpful when training language models with limited data. Training examples with rare words need repeated exposure for effective learning, making them difficult to learn (Platanios et al., 2019). Moreover, limited data can lead to high variance in gradients for rare word embedding due to insufficient contextual information. This suggests that word frequencies can be an effective difficulty heuristic.

Given a corpus of M sentences, $\{s_i\}_{i=1}^{M}$, where each sentence is a sequence of words, $s_i = \{w_1^i, ..., w_{N_i}^i\}$, word frequencies are defined as:

$$\hat{f}(w_j) \triangleq \sum_{i=1}^{M} \sum_{k=1}^{N_i} \mathbb{1}_{w_k^i = w_j}$$

where $j = 1, ...,$, #{unique words in corpus} and $\mathbb{1}_{condition}$ is the indicator function which is equal to 1 if its condition is satisfied and 0 otherwise. Here, we argue that using the product of the unigram probabilities of word frequency counts, which is employed to compute SR, is not an appropriate strategy for aggregating word frequencies into a singular difficulty score for sentences in the BabyLM Corpus. This approach implicitly incorporates sentence length into the difficulty score, resulting in shorter sentences being classified as easy and subsequently overrepresented in the training dataset when sampling from the difficulty-sorted datasets with the competence function. Instead, we propose using the average of the word frequencies as the singular score for sentence difficulty. This ensures that the difficulty metric is independent of sentence length. We thus propose the *average sentence rarity* difficulty heuristic:

$$d_{avg\_rarity}(s_i) \triangleq \frac{-1}{N_i} \sum_{k=1}^{N_i} \hat{f}(w_j)$$

For the easier sentences to receive a higher score using this metric, we incorporate the -1 factor in our difficulty metric. Implementing this difficulty metric along with the competence function to construct the training corpus led to a performance increase of over 3% on BLiMP, reaching 74.51% (Table 3).

5.2 Local Coherence

There is semantic similarity between consecutive sentences that convey information about the same concept. For instance, sentences from a Wikipedia article on engines are more similar compared to sentences from a conversation between parents and children about lunch. Therefore, adjacent sentences encoding the same concept tend to be more semantically similar. This semantic coherence between adjacent sentences is preserved when sentences from a dataset are in their original order. However, reordering sentences based on difficulty metrics can disrupt the semantic distribution of nearby sentences.

Learning contextualized word embeddings heavily relies on the sequence of words presented together within a context length. We hypothesize that the inferior performance of models developed using Competence-based CL optimizations, in comparison to baselines achieved with simple preprocessing steps, is likely due to the language model's inability to capture important context encoded by nearby sentences. This is because as a consequence of reordering sentences based on difficulty metrics, sentences are followed by others that are grammatically and semantically different, potentially sampled from other datasets, and encoding a completely different concept.

The objective here is to reorder sentences based on difficulty in a manner that minimizes the loss of contextual information encoded by nearby semantically similar sentences, to enhance model performance. To achieve this, we diverge from the competence algorithm proposed, which controls the order of all sentences that the model sees during training. Instead, we focus on a smaller-scale optimization problem by supervising the sequence of sentences that follow each other within a given context length. The order of sentences grouped at the context length level has a significant impact on the

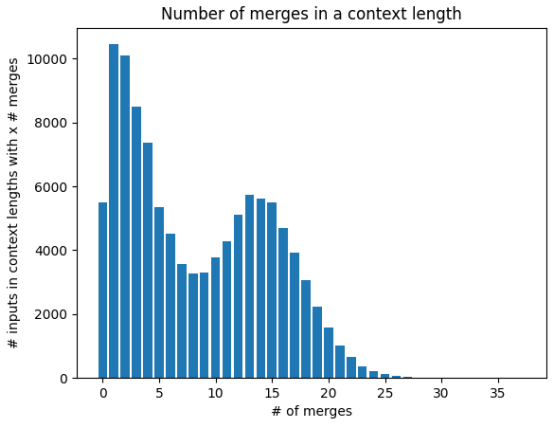

Figure 1: Frequency of context length size samples that are a merge of x number of tokenized inputs in the BabyLM datasets. The average number of merged inputs in a context length is 8.512. However, for the 80% longer portion of inputs, the average is 6.55.

model's performance, because, due to the shorter length of sentences in the BabyLM datasets, an average of 8 sentences are grouped within a context length of 128 tokens when using the concatenation algorithm to merge tokenized inputs (Figure 1).

To assess the extent of contextual information that is lost during sentence reordering, we use *local coherence* as a heuristic. This metric quantifies the pair-wise contextual similarity between a central sentence and its adjacent sentences within a window of seven sentences, as measured by `sup-simcse-roberta-large`, a model specifically designed to produce contextualized sentence embeddings (Gao et al., 2021). It's important to note that this measurement, produced by a pretrained Roberta model, is completely independent of the training process of our models. We define local coherence for sentence s_i as:

$$c(s_i) \triangleq \frac{1}{6} \sum_{\substack{k=-3 \\ k\neq 0}}^{3} sim(s_i, s_{i+k})$$

Where $sim(s_i, s_j)$ is the cosine similarity between the sentence embeddings encoded for s_i and s_j using `sup-simcse-roberta-large`. To determine the size of the local similarity window, we consider the average number of inputs concatenated in a context length of 128, which is 8.512 for all sentences in the BabyLM datasets. However, for the subset of sentences that make up the 80% of longer inputs, which are more influential in optimizing the model's performance for BLiMP, the average number of inputs merged in a context length of 128 reduces to 6.55. Thus, we opt for a window size of seven for this particular metric.

To unify the local coherence of individual sentences into a single metric for a given corpus, we use an average pooling function. However, due to limited computational resources and to enhance time efficiency, we opt for calculating the expected local coherence of a corpus. To compute this, instead of calculating the local coherence of all sentences in the corpus, we take the average of the local coherence values of 1000 randomly selected unique sentences from each dataset in the BabyLM Strict-Small track.

5.3 Data Cleaning With ASR

ASR sorts sentences based on the relative frequency of the words, classifying sentences with a high concentration of common words as the easiest and those with a high concentration of rare words as the hardest. Manual evaluation of datasets sorted using this metric indicates that sentences classified as easy tend to lack semantic meaning and appear fragmented in some datasets. This is expected, as this metric ignores sentence length, and thus, sentences classified as easy have few words besides the most frequent words, which include numbers, articles, and pronouns. This leaves limited room for meaningful development of concepts in those sentences. The datasets that display this pattern most prominently are `gutenberg`, `bnc`, and `cbt`. `gutenberg` contains thousands of lines consisting of a few words and a long series of numbers, likely corresponding to Project Gutenberg catalog numbers. These lines are isolated when the dataset is sorted by ASR and ranked as the easiest sentences.

We found that cleaning the datasets by removing redundant or semantically meaningless lines with a high count of common words can improve the model's performance. ASR also effectively identifies meaningless inputs containing a high count of rare words, as hard samples; however, we found that removing such samples did not provide an improvement in the model's performance. This might be because removing the limited contextual information available for the rare words either entirely erases them from the model's vocabulary or increases the variance in gradients of their embeddings, given the small size of our dataset. Alternatively, removing meaningless contextual information for high-frequency tokens from the datasets

can be advantageous, because when learning embeddings for the common words, the noise introduced by meaningless samples can be amplified due to the small size of our training datasets.

When employing SR to sort inputs in the datasets, the isolation of semantically meaningless lines does not occur, because this metric is dependent on sentence length. This difficulty metric fails to identify inputs with a high count of frequent tokens and a low count of all other tokens, which is a marker for meaningless inputs in the target datasets. Examples classified as hard tend to be very long, and at least segments of those sentences are coherent. On the other hand, the frequency of common words is lower compared to the frequency of other words in the inputs classified as easy, primarily due to the imposed short length limit for these inputs. As a result, sentence rarity cannot be used as an effective metric to clean these datasets.

6 Experiments

6.1 Reordering One Dataset

The primary objective of these experiments is to enhance the performance of the language model on BLiMP by grouping training samples with a similar difficulty, as quantified by either SR or ASR, in the same context length, and to measure the loss of contextual information when sentences are rearranged to this new order.

The BabyLM datasets are derived from various sources, each encoding distinct conceptual information. As a result, sentences from the same database exhibit a higher level of grammatical and semantic similarity. Thus, to preserve the maximum contextual information when rearranging sentences, we reorder sentences only within each dataset in this series of experiments.

To quantify the extent of contextual information loss following sentence reordering, we calculate the expected local coherence of each dataset in the Strict-Small track separately with the sentences of the dataset in their original order and with the sentences sorted based on either of the difficulty metrics (Table 4). As expected, rearranging sentences using either difficulty metric significantly reduced the expected local coherence across all datasets. When it comes to arranging sentences with a similar context close to each other, both metrics demonstrated comparable performance.

To assess the potential improvement of GPT-2's performance on the BLiMP evaluation through organizing sentences of a single dataset based on a difficulty metric, we conducted a series of 20 experiments. In each experiment, GPT-2 is trained on a training corpus consisting of 9 unchanged datasets concatenated with one dataset sorted based on difficulty. The model's performance is then evaluated on the BLiMP evaluation (Table 5). We also measure the correlation between the expected local coherence of the difficulty-sorted dataset and model performance to test our hypothesis that even though sorting inputs based on difficulty can improve performance, interrupting the semantic distribution of nearby contextual sentences can reduce the model performance.

We observed a positive correlation between the expected local coherence of datasets sorted by either difficulty metric and the evaluation results of the model on BLiMP (Figure 2). To assess the relationship between these two variables, we used Spearman's Rank correlation coefficient. The correlation coefficient between the coherence score of datasets sorted with SR and the BLiMP score of the models is 0.693, indicating a strong correlation. For datasets sorted with average sentence rarity, the coefficient is 0.559, indicating a moderate correlation.

The larger correlation coefficient achieved for datasets sorted with SR may be caused by the implicit similarity in length among neighboring sentences within the window of local coherence when sentences are sorted by SA. And this similarity in turn increases the local coherence score and BLiMP performance of the model. This suggests that considering sentence length when sorting sentences by difficulty is beneficial, however, it is the high sampling frequency from shorter sentences in our datasets, ranked as easier using SA, that reduces the model's performance when using the competence function.

Out of the 20 experiments conducted, 8 resulted in an improvement in the BLiMP evaluation relative to our baseline of 75.05% achieved by preprocessing `gutenberg`, and all results were above the 73.40% BLiMP score achieved with no optimizations. Notably, the model trained on `aochildes` sorted with SA achieved a 0.72% increase in BLiMP and reached a score of 75.77%.

The lower performance of certain models in this experiment on BLiMP is most likely attributed to the loss of significant contextual information in the dataset during the reordering of sentences based

	Datasets									
Order of Sentences	aochildes	bnc_spoken	cbt	children_stories	gutenberg	open_subtitles	qed	simple_wikipedia	switchboard	wikipedia
Original Order	0.303	0.228	0.227	0.326	0.307	0.204	0.240	0.348	0.249	0.400
SA	0.149	0.114	0.127	0.180	0.121	0.104	0.090	0.108	0.147	0.124
ASR	0.152	0.108	0.124	0.172	0.120	0.110	0.086	0.115	0.120	0.127

Table 4: Comparing the expected local coherence of each dataset when its sentences are in their original order to when the sentences are sorted based on sentence rarity (SA) or average sentence rarity (ASR).

	Rearranged Dataset In The Training Data									
Order of Sentences	aochildes	bnc_spoken	cbt	children_stories	gutenberg	open_subtitles	qed	simple_wikipedia	switchboard	wikipedia
SA	75.77%	75.19%	74.64%	75.42%	75.42%	74.58%	74.54%	74.29%	74.81%	75.48%
ASR	74.85%	75.37%	75.59%	75.40%	74.71%	74.69%	74.11%	74.32%	74.88%	74.74%

Table 5: BLiMP results for models trained on the BabyLM Strict Small Corpus with one dataset sorted based on SA or ASR.

on difficulty. This is evident from the positive correlation between the local coherence score of the dataset and the model's performance on BLiMP, which suggests that models that achieved lower performance on BLiMP were trained on datasets with higher contextual information loss.

The loss of contextual information may also be attributed to higher subject variance in certain datasets. In that case, to improve the preservation of local contextual information, it may be beneficial to sort sentences at a sub-dataset level. For instance, rearranging sentences from only a single story in the `children\_stories` dataset instead of rearranging all sentences in the dataset could potentially lead to better results. Furthermore, to enhance the model's performance on these datasets, it may be essential to implement a larger-scale supervision of the sentence order. This can be achieved through the development of a difficulty metric that considers the semantic similarity of consecutive sentences when reordering sentences from different datasets, leading to a minimum loss of contextual information when sorting sentences with different meanings and grammar styles.

6.2 Data Cleaning

In this series of experiments, we applied the previously discussed data-cleaning method to 3 datasets: `bnc`, `cbt`, and `gutenberg`. To set up these experiments, we initially sorted the datasets using ASR. Next, we determined the number of lines to eliminate from the easiest sentences in the dataset through manual evaluation. For every 200 lines, we assessed 10 lines and removed the preceding 200 lines if more than 1 out of the 10 lines contained grammatically incoherent or semantically meaningless sentences. Subsequently, the sorted and cleaned dataset was concatenated with the 9 base datasets to create the training corpus. We trained a model on each corpus and evaluated their performance using BLiMP. Table 6 compares the results of training GPT-2 on the training corpus composed of one dataset cleaned and sorted with ASR with the experiment results achieved earlier by only reordering the dataset with ASR.

	Data Cleaning with ASR		ASR
Dataset	BLiMP	# Lines Cut	BLiMP
`bnc_spoken`	75.53%	4600	75.37%
`cbt`	75.69%	800	75.59%
`gutenberg`	75.84%	3200	74.71%

Table 6: A comparison between the results of training GPT-2 on the training corpus consisting of one dataset cleaned and sorted with ASR and the earlier experiment results obtained by simply reordering the dataset with ASR. The number of lines eliminated from the sorted dataset (after duplicates were removed) is also stated.

By employing this method, we achieved a considerable improvement in the performance of the model trained on the cleaned `gutenberg` dataset. However, the improvement achieved in the performance of the two other models was negligible. We believe the substantial enhancement on `gutenberg` is because a higher portion of the excluded inputs was meaningless relative to the inputs cut from the other two datasets. The model trained on ASR sorted and cleaned `gutenberg` performed the best on BLiMP among the other models we trained and is the model submitted for the challenge. This model's perplexity on the BabyLM test datasets is 54.8.

7 Conclusion and Future Work

In conclusion, the primary objective of this paper was to enhance GPT-2's performance on BLiMP zero-shot tasks by pre-training the model on the

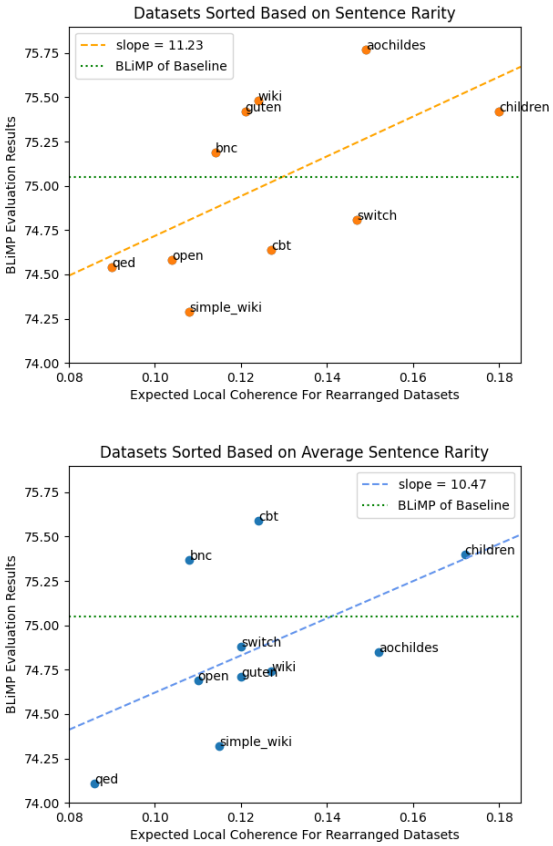

Figure 2: The graph illustrates a positive correlation between the expected local coherence of the sorted dataset and the BLiMP score of the model trained on it. The Spearman's Rank correlation coefficient is 0.693 for datasets sorted with SA (represented in orange) and 0.559 for those sorted with ASR. The green line indicates the best BLiMP score obtained without any CL optimizations, achieved by preprocessing the `gutenberg` dataset.

datasets provided in the BabyLM Strict Small track. Various difficulty metrics were explored to supervise the order of sentences presented to the model. It was observed there's a positive correlation between the BLiMP result of models trained on a corpus comprised of one dataset sorted based on difficulty and the contextual coherence of nearby sentences in the rearranged dataset. Thus, Training models on a dataset sorted by difficulty with preserved contextual coherence could lead to better performance on BLiMP. By employing difficulty-based sentence reordering, we trained a model that achieved an average accuracy of 75.77% on BLiMP's zero-shot tasks. Additionally, we used *average sentence rarity*, a length-independent sentence rarity metric, to clean and sort the `gutenberg` dataset, which further improved the performance to 75.84%.

Hence, to improve curriculum learning optimizations for language models, we argue that sentences presented together within a context length should exhibit not only the same level of difficulty but also semantic and contextual similarity. In our study, we employed similarity measures to assess the contextual coherence of rearranged datasets after the sentences were ordered based on word frequencies; the semantic similarity of sentences had no impact on the actual order of the sentences. A critical future advancement arising from this research lies in the development of more sophisticated difficulty metrics that consider both the similarity among sentences and their individual difficulty levels.

8 Limitations

No measure of grammatical similarity of sentences: When assessing the correlation between the expected local coherence of a dataset and the performance of the model trained on the rearranged dataset, we are considering the semantic similarity of sentences within a context length, but using a grammar-based evaluation to assess the model's performance. While we hypothesize that training the model on difficulty-sorted datasets that have more semantically similar sentences sequenced after each other improves the model's overall performance, leading to better BLiMP results, it might be more effective to optimize for higher BLiMP scores by evaluating the grammatical similarity of sentences that follow each other. Nevertheless, there is currently no reliable method to solely measure the grammatical similarity of two sentences to the best of our knowledge. Alternatively, using an evaluation pipeline that assesses the model's semantic understanding of sentences would be a good way to compare against the received local coherence scores. However, our available resources did not allow us to optimize our models using such pipelines.

Lack of scalability: Our current approaches to enhance model performance are not scalable as reordering two or more datasets did not yield any improvement in BLiMP scores in our experiments. This lack of scalability is the motivation behind the investigation of the semantic similarity of sentences that follow each other in a context length. We hypothesize that although sorting a higher number of datasets increases the number of context-length samples where the concatenated sentences have the same difficulty, sequencing sentences from different sources with distinct grammar styles and semantic meanings within a context length results in

a decrease in the model's performance. The resolution to this scalability issue lies in the development of more advanced difficulty metrics that take into account both the similarity between sentences and their individual difficulty levels when reordering the training samples.

References

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Philip A. Huebner, Elior Sulem, Fisher Cynthia, and Dan Roth. 2021. BabyBERTa: Learning more grammar with small-scale child-directed language. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 624–646, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabas Poczos, and Tom M Mitchell. 2019. Competence-based curriculum learning for neural machine translation. *arXiv preprint arXiv:1903.09848*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Gotlieb Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Adina Williams, Bhargavi Paranjabe, Tal Linzen, and Ryan Cotterell. 2023. Findings of the 2023 BabyLM Challenge: Sample-efficient pretraining on developmentally plausible corpora. In *Proceedings of the 2023 BabyLM Challenge*. Association for Computational Linguistics (ACL).

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.