Improving LLM Pretraining by Filtering Out Advertisements

Anonymous ACL submission

Abstract

Large language model (LLM) performance is increasingly linked to not just the size but also the quality of internet-derived datasets. While LLM data selection methods have evolved, their evaluations often rely on overall metrics that may not capture their impacts on different downstream task performances. Motivated by this gap, our study finds that selecting pretraining data based on loss metrics could result in poor performance on knowledge-intensive benchmarks, such as the MMLU. Addressing this, we focus on filtering out low-information content, specifically ads, and create an effective ad classifier for this purpose. Besides, the most straightforward approach to assess the quality of pretraining datasets is to train a full-scale LLM, but this is prohibitively expensive and impractical for large-scale comparative studies. To overcome this, we use a smaller, 100M parameter LLM as a proxy to predict the downstream performance of larger models. We effectively demonstrate the correlation between the small model's proxy indicators and the large SFT model's downstream task metrics. This smaller model evaluation technique not only greatly shortens the cycle time for refining data selection strategies but also achieves significant budget savings, amounting to 92.7%. Finally, our findings suggest eliminating advertisement content not only improves performance on knowledge-intensive benchmarks but also yields commendable results across various other capability dimensions within benchmarks. We will publish part of our work soon.

1 Introduction

Pre-training on extensive unlabeled and uncrated corpus sourced from internet snapshots (Gao et al., 2020; Penedo et al., 2023; Computer, 2023; Soldaini et al., 2024), empowers large language models (LLMs) with unprecedented capabilities across various domains. Meanwhile, the performance of LLMs scales as a power law with regards as to the data quantity (Kaplan et al., 2020). However, alongside quantity, the quality of the corpus is equally crucial. Recent consensus suggests that high-quality corpora have the potential to significantly alter scaling laws (Sorscher et al., 2022), enabling performance on par with large-scale models while requiring leaner training costs (Gunasekar et al., 2023; Eldan and Li, 2023)

Therefore, many studies have explored LLM pretraining data selection, including rule-based (Rae et al., 2021), metric-based (Coleman et al., 2019; Marion et al., 2023a; Tirumala et al., 2023), and semantics-based (Brown et al., 2020), each employing different criteria for data quality. Yet, these methods are commonly evaluated by overall metrics, overlooking the detailed influence on different downstream task performances.

Motivated by this gap, we investigate the impact of these strategies on downstream tasks. Surprisingly, our experiments reveal while loss filtering (Marion et al., 2023b) enhances text fluency, it can also diminish performance on knowledge-intensive benchmarks like MMLU (Hendrycks et al., 2020). This decline is linked to two main issues: first, the tendency of loss filtering to preferentially preserve fluency-centric marketing content, leading to its overrepresentation; second, the potential exclusion of knowledge-dense texts that incur higher losses when they elude the capturing capabilities of the underlying LLM. Moreover, domain-specific filtering(e.g., Wikipedia classifier (Brown et al., 2020)), although intended to curate domain-relevant data, risks losing valuable cross-domain information.

Based on the previous discussion, we pose two questions:

1. ***Is it possible to devise a data selection strategy that minimizes the inclusion of low-information content while preserving high-information content?***

2. ***How can we quickly assess the effectiveness***

of data selection strategies in pre-training scenarios?

To answer the first question, we focus on identifying common traits within web datasets to address the prevalence of low-information content in corpora. Our investigation reveal that advertisements significantly contribute to this issue. In response, we develop an ad classifier, a step beyond the initial mentions in prior work (Wu et al., 2021), providing a detailed approach and thorough analysis of its positive impact on LLM benchmarks, especially knowledge-intensive benchmarks.

To answer the second question, setting aside the costly approach of directly training an LLM end-to-end, D4 (Tirumala et al., 2023) have taken a step forward by exploring the use of proxy metrics from smaller models to validate the quality of pre-training data filtering. However, there are several limitations to these approaches. Firstly, insufficient training (e.g., 1.3B-parameter models on 40B tokens, and 6.7B-parameter models on 100B tokens) obscures the manifestation of higher-order abilities, such as knowledge comprehension as measured by tasks like the MMLU. Secondly, proxy indicators, including perplexity (PPL) from pre-training and various NLP task validation sets, lack sufficient correlation with downstream task performance, limiting domain-specific insights. To address these issues, on the one hand, we evaluate base models after SFT, which reveals higher-order skills like knowledge comprehension even with limited training. On the other hand, we enhance the proxy indicators for small models by including PPL based on validation sets converted from downstream tasks, enabling early downstream performance predictions and quantifying the correlation between small model proxies and post-SFT large-model downstream metrics. Specifically, we find that the performance of a larger-scale SFT model can be well characterized through the PPL of a 100M proxy LLM on the validation sets.

Using a 100M-parameter proxy model for rapid pre-training iterations (pretraining budget analysis see Section 5.4), we comprehensively assess popular data selection methods for LLMs, comparing them against our ad classifier's performance. As depicted in Figure 1, our analysis pipeline highlights the impact of various strategies on model efficacy. Our findings suggest that eliminating advertisement content not only improves performance on knowledge-intensive benchmarks but also yields commendable results across various other capability dimensions within benchmarks.

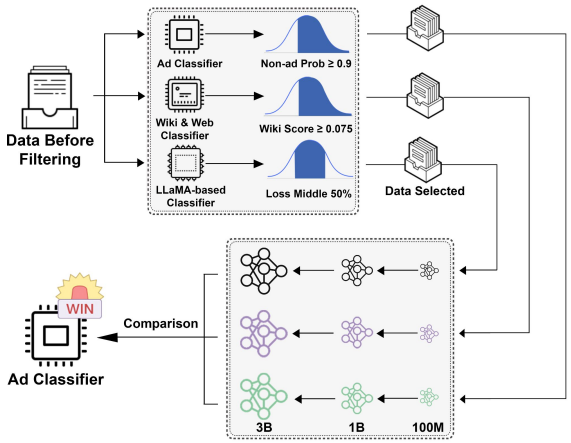

Figure 1: Ad Filtering Outperforms Other Methods Across Three Pre-training Data Selection Techniques

In summary, our contributions are as follows:

1. We demonstrate that employing a 100M-parameter LLM can reliably predict the utility of pretraining corpora for larger models. We comprehensively establish the correlation between the proxy indicators of the small model and the downstream task metrics of the large SFT model.
2. We emphasize that by using the proxy small model evaluation mechanism with 100M parameters, we can dramatically reduce the iteration cycles of pre-training data selection strategies, resulting in a substantial budgetary saving of 92.7%.
3. We highlight that eliminating advertisement content substantially not only enhances the efficacy of knowledge-intensive benchmarks but also yields commendable results across various other capability dimensions within benchmarks. Additionally, the extent of these performance enhancements varies depending on the data filtering applied, indicating differential downstream effects.

2 Related Work

2.1 Data Selection

As previously emphasized, the importance of high-quality data for training LLMs cannot be overstated. Research on data selection extends across various fields, sharing fundamental principles despite diverse applications. We identify four primary data selection methodologies and provide a systematic analysis of each in the following sections.

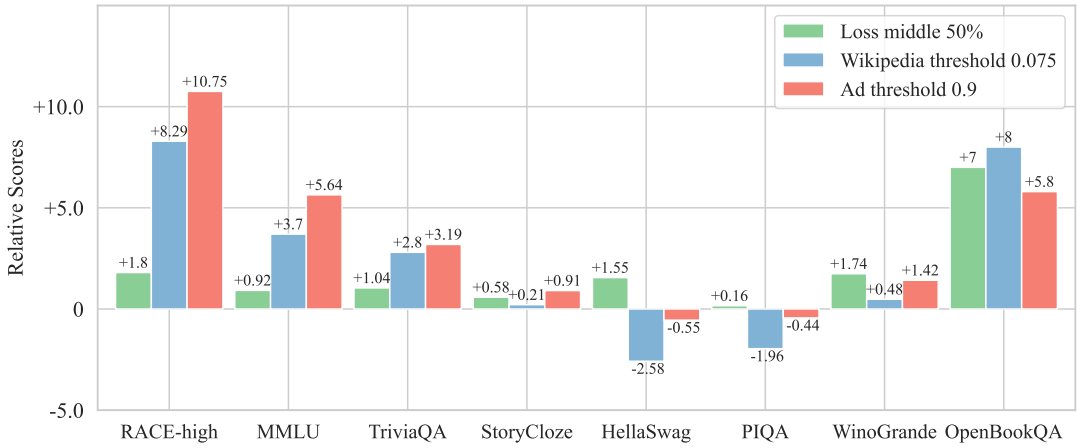

Figure 2: The relative score of performance between different data selection methods with Non-pruning method. In this Figure, each of the models is pre-trained with 300B tokens. See Table 1 and 2 for absolute performance of downstream tasks.

Metric-Based Data Selection This line of work primarily focuses on filtering data based on automated metrics generated through dynamic model training. One part of these works explores data filtering on computer vision (CV), with filtering strategies including prioritizing hard sample sampling(Coleman et al., 2019), moderate sample sampling(Xia et al., 2023), uncertainty sampling, and filtering based on dynamic changes in statistical values across different epochs(Paul et al., 2021). Another part of the work explores data filtering in the context of NLP and LLM scenarios. The filtering approaches include using perplexity scoring(Marion et al., 2023a; Wang et al., 2023), custom IFD(Li et al., 2023a), and multi-metric loss fitting(Cao et al., 2023). In summary, these efforts primarily rely on statistical patterns in the data to obtain valuable samples for model training. However, they struggle to perceive the semantic information in the samples and have difficulty understanding the diversity distribution of the samples.

Semantics-based Data Selection This line of work primarily involves scoring data based on the Wikipedia & Web classifier(Brown et al., 2020; Touvron et al., 2023), reward model(Du et al., 2023), and LLM(Eldan and Li, 2023; Chen et al., 2023; Li et al., 2023b). Intuitively, a semantics-based scoring strategy should have the ability to recognize semantics. However, at the same time, special attention must be paid to whether the filtering is biased(Gao, 2021).

Geometry-based Data Selection This line of work primarily involves conducting diversity-prioritized sampling based on the clustering situation in the feature space. These works often combine with metric-based or semantic-based data filtering strategies(Maharana et al., 2023; Du et al., 2023; Tirumala et al., 2023).

Rule-based Data Selection Several research works (Computer, 2023; Soldaini et al., 2024; Rae et al., 2021; Workshop et al.) tries to establish a number of hand-curated filtering techniques to remove low-quality examples. While these hand-curated filters can mitigate the inclusion of certain noisy examples, they cannot serve as a comprehensive substitute for a robust metric that assesses the 'quality' of individual training examples.

2.2 Evaluation of Pre-training Data Selection

In addition to D4 (Tirumala et al., 2023) as mentioned in section 1, (Marion et al., 2023b) exhibits pre-trained models of 124M and 1.5B parameters with Validation Set Perplexity and downstream SFT task evaluation. However, it is limited by the use of a validation set whose domain is aligned with the training dataset's distribution. Perplexity rankings within in-domain validation sets can be inconsistent across different data selection strategies, potentially misrepresenting a model's true capabilities. Furthermore, it only reports classification task performance on GLUE after SFT, offering a partial view of LLM's overall abilities. We not only extend beyond those mentioned in comparison with D4 but also include our choice of validation sets. We select three types of validation sets, which are all out of training set domains, to reflect the model's generalization on smaller scales.

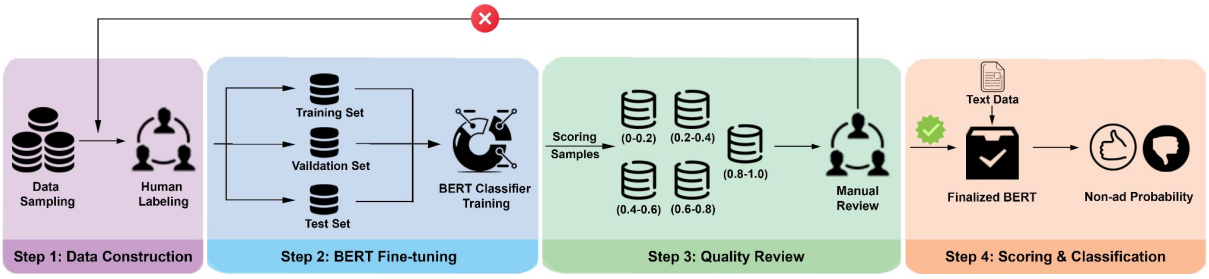

Figure 3: Pipeline of Data Labeling & BERT Classifier Training

3 Method

As previously outlined, the data selection pipeline is depicted in Figure 1. Within this pipeline, a small proxy model evaluation mechanism is employed to predict the downstream performance of the larger SFT models. Our investigation commences with an analysis of prevalent LLM data selection techniques, including the loss filter and the Wikipedia Classifier, with a focus on their influence on downstream tasks. Subsequently, we delve into the development and efficacy of the advertisement classifier. The critical components of this process are elucidated below.

3.1 Small Proxy Model Evaluation Mechanism

We scale up the parameters of our pre-trained model in a stepwise manner, initially from 100M to 1B and subsequently from 1B to 3B. Each data filtering strategy undergoes a thorough performance evaluation. We explore the potential of smaller models to predict the outcomes of their larger counterparts, utilizing the 100M model for hyperparameter selection and presenting a 1B model for comparison within a manageable training cost range. However, the substantial training cost associated with the 3B model prevents its use for hyperparameter experiments at this stage. As a result, we directly apply the optimal hyperparameters obtained from the 100M model to pre-train and SFT the 3B model, followed by downstream evaluation.

3.2 Loss Filter

This method leverages pre-trained models to compute perplexity for the entire dataset. It is indicated that employing moderate perplexity thresholds for data filtering can enhance training efficiency (Marion et al., 2023a; Xia et al., 2023), a hypothesis we will explore in depth. A detailed explanation of the relevant hyperparameters can be found in A.3.1

3.3 Wikipedia and Web Classifier

Contrasting with the ad filter, this strategy employs a binary classifier to separate high-quality, knowledge-rich text (e.g., Wikipedia) from low-quality Common Crawl data (Brown et al., 2020; Chowdhery et al., 2023; Touvron et al., 2023). Despite superficial similarities to the ad filter, this method focuses on the automatic segregation of text corpora, aiming to enhance data quality for pre-training. However, defining clear-cut divisions between these text types presents significant challenges and may inadvertently introduce biases. We will delve into a detailed analysis of these biases in subsequent Section 5.2.2. Details of the relevant hyperparameters can be found in A.3.2.

3.4 Advertisement Classifier

In our examination of the English Common Crawl corpus, we observe a significant prevalence of marketing content and product placements. Notably, product placements frequently exhibit redundancy and lack of fluency, whereas marketing content is typically distinguished by its high fluency. Given this background, we aim to sift through the data, removing ads to potentially enhance the corpus with knowledge-intensive material of higher quality for LLM pretraining. We filter out advertisements through a well-designed ad classification process, involving data sampling from RefinedWeb, human annotation, and a binary BERT model to distinguish non-ads from ads. The process was iterative, with continuous manual review and re-labeling of misclassified samples until achieving a desired low ad misclassification rate. The development of this ad classifier, aligned with human judgment, is depicted in Figure 3.

Unlike Yuan1.0, which uses a ternary classifier to filter a Chinese corpus into low-quality, advertising, or high-quality texts based on repetition rates (Wu et al., 2021), we categorize texts as advertising or non-advertising by focusing on promotional con-

tent and product placement. Yuan1.0's methodology, which targets coherent but redundant texts like website descriptions, differs from our content and style-based approach. Furthermore, while Yuan 1.0 has not disclosed their pre-training experiment results, we have detailed ours in 5.2.3.

4 Experiments

4.1 Training Details

Our pretrain experiments are conducted with the RefinedWeb dataset (Penedo et al., 2023), which uses advanced rule-based filtering and deduplication methods, **without any secondary classifier-based filtering**. In this way, we are able to implement detailed ablation studies, comparing the impacts of various filtering methods. and SFT experiments are with Flan Collection (Longpre et al., 2023). In our experiment, we train decoder-only Transformer from scratch only once for each experiment due to constraints of training costs. We provide full details of pre-training and SFT hyperparameters in Appendix A.1.1 and A.1.2. Meanwhile, we estimate computational costs in A.1.3.

4.2 Evaluation Metrics

We consider two key metrics for evaluation: validation set PPL and downstream benchmark metrics, with a detailed correlation analysis in Section 5.1.

Validation Set Perplexity To evaluate the model's impact on downstream tasks, we utilize three distinct validation datasets, with each catering to different domains, to offer an early performance assessment for models with 100M parameters. Detailed descriptions are available in Section A.1.4.

Downstream Benchmark Metrics We select 10 tasks across five categories to gauge our model's effectiveness on downstream tasks: text completion (Mostafazadeh et al., 2017), reading comprehension (Lai et al., 2017), common-sense question answering (Zellers et al., 2019; Bisk et al., 2020; ai2, 2019; Mihaylov et al., 2018), factual question answering (Kwiatkowski et al., 2019; Joshi et al., 2017), and examination(Hendrycks et al., 2020). An overview of these tasks is presented in A.1.5.

5 Result

5.1 Correlation Analysis of Proxy and Downstream Metrics

This study quantitatively assesses the correlation between the proxy metric (validation set PPL) of the 100M model and the downstream task metrics of the 3B SFT model. The evaluation employs a three-stage correlation analysis, using a 1B model as a bridge to handle the significant increase in training costs and improve the correlation calculation's reliability. The ranking correlation is quantified using Pearson and Spearman Correlation coefficients, with each of them corresponding to "P" and "S" in the figures respectively. Correlation values closer to 1 indicate a higher-ranking correlation.

In the first phase, our study commences with the analysis of 14 sets of experiments, focusing on proxy metrics for models with 100M and 1B parameters, resulting in 91 paired experiments over 11 validation sets. To counter early training instability, we utilize PPL values from models trained with 100B tokens as the proxy metric. As demonstrated in Figure 4, there's a high correlation in PPL between the 100M and 1B models across most validation sets, with exceptions noted in specific datasets such as RACE-middle and TrivialQA. Generally, smaller models can predict the PPL of larger models accurately, although discrepancies in correlation coefficients are observed. Nonetheless, a clear trend is evident: an increase in PPL differences among smaller models tends to predict similar trends in larger models. Further correlation details across validation sets are presented in section A.2.

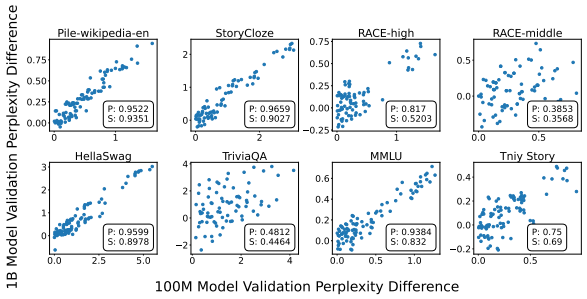

Figure 4: Validation Perplexity Difference Comparison Between 100M and 1B Model

In the second phase, we conduct experiments with 7 sets of data filtering hyperparameters, each comprising proxy indicators for both 1B and 3B models. We calculate the PPL difference between each paired hyperparameter set, resulting in 21 experimental pairings on each of the seven validation sets. Considering potential early training instability, we use PPL values at the 100-billion token training mark as our metric. As illustrated in Figure 5, the PPL of the 1B and 3B models show a significant correlation across most validation datasets, with a

lower correlation on RACE-middle and TrivialQA datasets, consistent with the first phase, More figures depicting the correlation on different validation sets can be seen in section A.2.

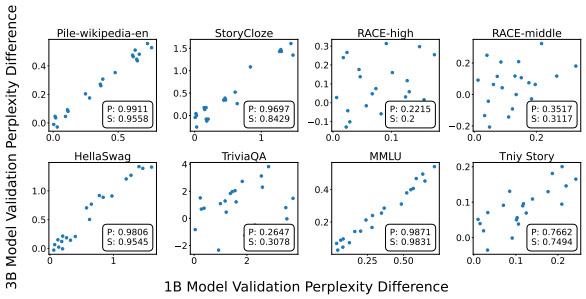

Figure 5: Validation Perplexity Difference Comparison Between 1B and 3B Model

The final phase involves experiments with 7 sets of data filtering hyperparameters, each containing 3B proxy indicators and corresponding downstream evaluation metrics. As depicted in Figure 6, A Correlation value approaching -1 indicates a strong negative correlation, suggesting that PPL in different 3B models on the validation sets correlates with higher downstream task metrics. For most tasks, PPL can effectively predict the performance of larger models on downstream tasks. Some tasks exhibit greater variance in downstream performance, resulting in a lower correlation coefficient. Nonetheless, the graph still reveals a distinct trend: as the PPL decreases, there is a gradual improvement in the performance of downstream tasks. Detailed analysis can be seen in Appendix A.2.3.

Summarizing the previous analysis, using a 100M parameter LLM can serve as a reliable indicator for the effectiveness of pretraining corpora when applied to larger models.

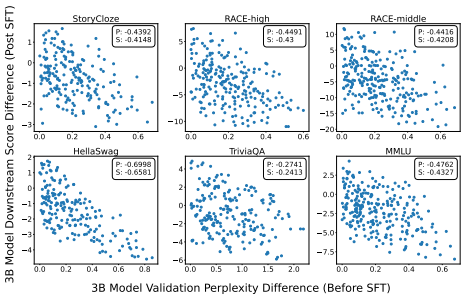

Figure 6: 3B Model Validation Perplexity Difference vs. 3B Model Downstream Score Difference

5.2 Pretraining Efficacy of Different Data Filtering Methods

5.2.1 Loss Filtering Performace

Our analysis of the impact of data selection strategies on the LLaMA2-7B model at 100M and 1B parameter scale reveals varied outcomes. Strategies include no filtering, and retaining the central 50%, and 30% of data by loss ranking. As detailed in Figure 7 and further in Figure 12, loss filtering shows mixed results: it decreases PPL (increase performance) on the HellaSwag test but increases PPL (decreases performance) on knowledge-intensive datasets like MMLU and Pile subsets. Conversely, it benefits the Tiny Story test set by reducing PPL. Based on these insights, we choose to retain the central 50% of data by loss, finding it to be the most effective strategy.

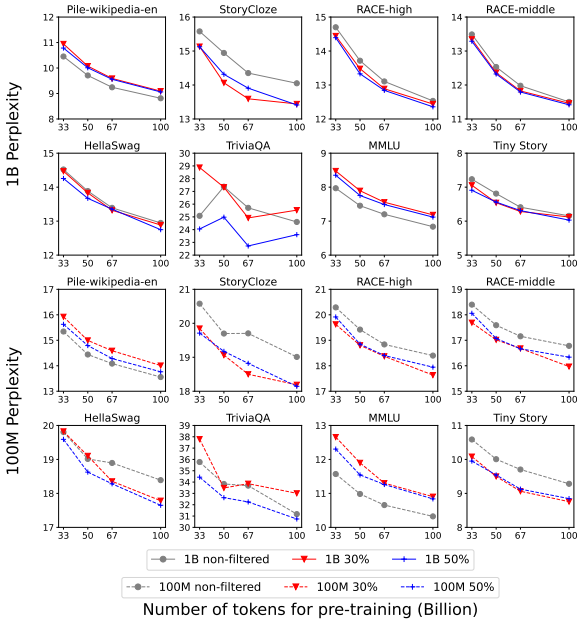

Figure 7: Validation Perplexities Comparison Between 100M & 1B Models with Moderate Perplexity Filtering

Further evaluation of the 3B SFT model, as shown in Table (1) and Table (2), indicates that the strategy of retaining the middle 50% of data based on loss generally surpasses the no-pruning method across most tasks, However, in knowledge-intensive tasks, this approach is less effective compared to other selection methods.

5.2.2 Wikipedia Classifier Performace

We analyze the perplexity curves of downstream validation sets for the 100M and 1B parameter pretrained models, as depicted in Figure 8, with additional results in Figure 13. These models process datasets refined by the Wikipedia & Web Classifier using different thresholds. The efficacy of this filtering varies: while some Pile validation subsets and the MMLU test show decreased PPL, indicating enhanced pre-training from filtering, the HellaSwag validation set see an increase in PPL, likely due to the loss of relevant data. In the case

	Data Remaining	Reading Comprehension		Exam	Factual QA	
		RACE-High	RACE-middle	MMLU	Natural Question	TriviaQA
No Pruning	100%	29.33	32.38	29.71	11.19	30.61
Loss middle 50%	53.9%	31.13	36.84	30.63	9.56	31.65
Wikipedia threshold 0.075	63.4%	37.62	41.57	33.41	**12.35**	33.41
Ad threshold 0.9	53.9%	**40.08**	**45.82**	**35.35**	12.08	**33.8**

Table 1: The downstream metric of each data selection method, including Reading Comprehension, Exam, and Factual QA, with 3B models pretrained with 300 billion tokens. Underlined results surpass the baseline performance with no pruning. The best results for each task are marked in bold.

	Data Remaining	Text Completion	Common-Sense QA			
		StoryCloze	HellaSwag	PIQA	WinoGrande	OpenBookQA
No Pruning	100%	75.15	64.75	77.15	57.93	22
Loss middle 50%	53.9%	75.73	66.3	**77.31**	**59.67**	29
Wikipedia threshold 0.075	63.4%	75.36	62.17	75.19	58.41	30
Ad threshold 0.9	53.9%	**76.06**	64.2	76.71	59.35	27.8

Table 2: Downstream metric of each data selection method, including Text Completion, Common-Sense QA, with 3B models pretrained with 300 billion tokens. Underlined results surpass the baseline performance with no pruning. The best results for each task are marked in bold.

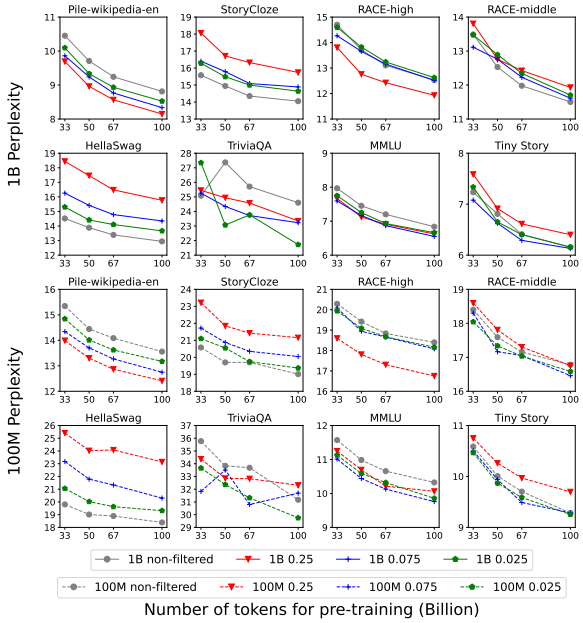

Figure 8: Validation Perplexities Comparison Between 100M & 1B Models with Wikipedia & Web

of Tiny Story, a 0.25 threshold increases perplexity compared to no filtering, but lower thresholds of 0.075 and 0.025 initially reduce PPL, aligning with unfiltered data by 24,000 steps. This pattern underscores the nuanced effect of data filtering on text generation fluency. Generally, RefinedWeb data demonstrates a marginal improvement with a 0.075 threshold, suggesting selective filtering benefits.

Building on this analysis, we further evaluate the downstream results for the 3B model, as presented in Table (1) and Table (2). Our observations indicate that applying a Wikipedia data selection threshold of 0.075 substantially enhances performance across a majority of evaluated tasks, in comparison to the baseline no-pruning method. However, this improvement does not extend to a subset of common sense question-answering tasks, where the method's efficacy appears to be limited.

5.2.3 Ad Classifier Performance

We train a BERT classifier using manually annotated data with non-ad text to be labeled 1 and ad text to be labeled 0. Then we apply the trained BERT on another batch of manually annotated data for ad classification to validate the effectiveness of our classifier, where we reach the average precision of 96.63% for non-ad classification and 80.66% for ad classification. The resulting Precision-Recall curve with confidence intervals is depicted in A.3.3. Additionally, we explore varying ad identification thresholds to refine our model, training across different scales: 100M, 1B, and 3B models, to optimize ad recognition capabilities.

Our analysis begins with an examination of PPL curves for downstream validation sets of the pretrained 100M and 1B models, as shown in Figure 9, with additional data in Figure 16. We observe that with the ad threshold of 0.9, both 100M and 1B models achieve a PPL lower than that observed with the no-pruning method across most validation sets. This performance is also better compared to other evaluated thresholds. Consequently, we select 0.9 for the pre-training of 3B model.

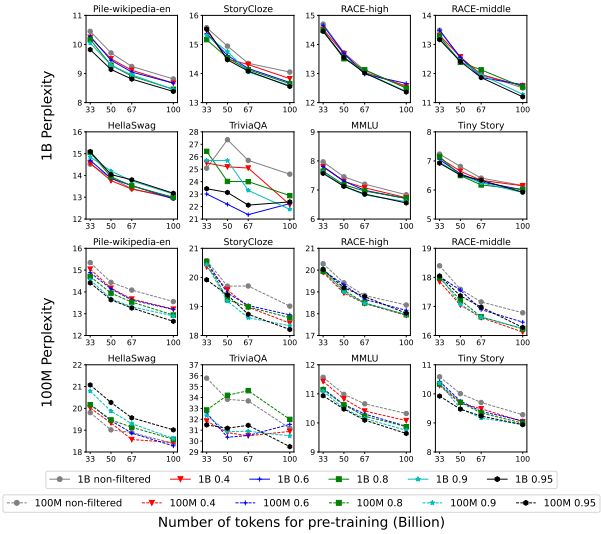

Figure 9: Validation Perplexities Comparison Between 100M & 1B Models with Ad Filtering

	Wiki threshold 0.075	loss middle 50%	Ad threshold 0.9
Pile-wikipedia	68.8%	17.5%	98.3%
StoryCloze	0.1%	63.2%	98.9%
RACE-High	67.6%	75.9%	74.5%
RACE-Middle	45.5%	70.8%	88.4%
HellaSwag	0.3%	52.2%	95.2%
TriviaQA	0.1%	7.2%	99.5%
MMLU	82.7%	11.1%	94.4%
Tiny Story	33.0%	5.0%	99.6%

Table 3: Data Remaining Rates for Different Data Filtering Schemes on Downstream Validation Sets of Different Domains

Downstream Results of 3B SFT model are displayed in Table (1) and (2). We observe that the ad threshold of 0.9 yields superior performance on most of tasks when compared to the no-pruning and other methods, especially in knowledge-intensive benchmark, MMLU. In other benchmarks, this method also shows commendable results.

5.3 Analysis of Data Remaining Ratios for Different Data Filtering Methods

We evaluate the data retention ratios of various filtering strategies on validation sets as an indirect measure of their influence on downstream tasks. Despite the validation set partly originating from downstream instruction tasks, which diverge in format from our pre-training corpus, we consider these tasks as domain-specific corpus material. Consequently, we propose that the varying data remaining ratios across domains within our validation set can provide insights into the impacts of data filtering strategies on these domains. Furthermore, comparing data retention ratios for different strategies within the same validation set domain can yield relative effectiveness insights.

As shown in Table (3), the loss filtering method results in a reduced data remaining ratio on the MMLU, indicating potential negative impacts on the MMLU benchmark. This observation aligns with the finding that loss filtering falls short of other strategies in the 3B SFT-enhanced MMLU context. Similarly, the Wikipedia filtering strategy, with its lower data retention ratio on HellaSwag, suggests a detrimental effect on the common sense benchmark, corroborating its underperformance in post-3B SFT HellaSwag evaluations. Interestingly, the ad filtering strategy consistently exhibits high data remaining ratios across the validation set, an outcome achieved without incorporating any information from the validation set in the development of our ad classifier.

5.4 Analysis about Cost of Proxy Model

Proxy small model evaluation mechanism dramatically reduces the iteration cycles for refining data selection methods, cutting down the computational expense from 3472 GPU hours for a 3B model to 253 GPU hours for a 100M model, thereby **saving approximately 3219 GPU hours**. Detailed computational costs see Appendix A.1.3.

6 Conclusion

In our study, we have shown that reliance on loss metrics for pretraining data selection can adversely affect performance on complex, knowledge-dependent tasks such as MMLU. By developing a specialized ad classifier to filter out low-information content, we have enhanced the data quality for LLM training, leading to measurable improvements in model performance across a range of benchmarks. Furthermore, we've introduced a cost-effective and time-efficient evaluation methodology using a smaller LLM to predict the potential downstream success of larger models. This proxy approach has proven to be a valuable tool for dataset refinement, offering a reduction in resource expenditure by 92.7% The significant budgetary savings and the ability to rapidly iterate on data selection strategies make this a scalable and practical solution for future LLM development.

Limitations

Small models to predict the reasoning ability of large models: The reasoning ability of existing LLMs emerges under certain conditions, such as model size, high-quality mixed data, and a certain computational budget. We do not have the time to explore whether it is possible to use smaller models on web datasets with appropriate proxy indicators to reflect the reasoning ability of a medium-sized model. There is no consensus yet on the origins of the reasoning mechanism produced by LLMs. If the changes in reasoning ability could be reflected through proxy indicators on smaller models, it would greatly aid in understanding the origins of reasoning abilities.

Ad filtering in conjunction with other filtering solutions: Ad filtering is about removing corpora with advertising content. Although loss filtering may discard knowledgeable content, it can still eliminate a lot of incoherent corpora. What kind of integrated scheme could complement the advantages of multiple filtering solutions? Limited by time and cost, we have not explored the integration of multiple existing filtering solutions in this work.

7 Ethics Statement

7.1 Data Collection

All the datasets we use in our work are from publicly available resources (RefinedWeb). And we will open part of quality scores of this dataset. The data License will follow RefineWeb.

7.2 Human Labeling

For the BERT advertisement classifier, we curate a dataset of 40,000 samples from RefinedWeb, which are then labeled as either advertisement (ad) or non-advertisement (non-ad) by annotators. Because the annotators are formal employees of the company and are subject to confidentiality requirements regarding their remuneration, it is not possible to provide information on average salaries to the outside. The form and instructions presented to human evaluators are shown in Figure 14.

References

2019. Winogrande: An adversarial winograd schema challenge at scale.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Yihan Cao, Yanbin Kang, Chi Wang, and Lichao Sun. 2023. Instruction mining: When data mining meets large language model finetuning.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, et al. 2023. Alpagasus: Training a better alpaca with fewer data. *arXiv preprint arXiv:2307.08701*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. 2019. Selection via proxy: Efficient data selection for deep learning. *arXiv preprint arXiv:1906.11829*.

Together Computer. 2023. Redpajama: an open dataset for training large language models.

OpenCompass Contributors. 2023. Opencompass: A universal evaluation platform for foundation models. https://github.com/open-compass/opencompass.

Qianlong Du, Chengqing Zong, and Jiajun Zhang. 2023. Mods: Model-oriented data selection for instruction tuning. *arXiv preprint arXiv:2311.15653*.

Ronen Eldan and Yuanzhi Li. 2023. Tinystories: How small can language models be and still speak coherent english? *arXiv preprint arXiv:2305.07759*.

Leo Gao. 2021. An empirical exploration in quality filtering of text data. *arXiv preprint arXiv:2109.00698*.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. 2023. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. 2023a. From quantity to quality: Boosting llm performance with self-guided data selection for instruction tuning. *arXiv preprint arXiv:2308.12032*.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Luke Zettlemoyer, Omer Levy, Jason Weston, and Mike Lewis. 2023b. Self-alignment with instruction backtranslation. *arXiv preprint arXiv:2308.06259*.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. *arXiv preprint arXiv:2301.13688*.

Adyasha Maharana, Prateek Yadav, and Mohit Bansal. 2023. D2 pruning: Message passing for balancing diversity and difficulty in data pruning. *arXiv preprint arXiv:2310.07931*.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. 2023a. When less is more: Investigating data pruning for pretraining llms at scale. *CoRR*, abs/2309.04564.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. 2023b. When less is more: Investigating data pruning for pretraining llms at scale.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*.

Nasrin Mostafazadeh, Michael Roth, Annie Louis, Nathanael Chambers, and James Allen. 2017. Lsdsem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 46–51.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. 2021. Deep learning on a data diet: Finding important examples early in training. *Advances in Neural Information Processing Systems*, 34:20596–20607.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, et al. 2024. Dolma: an open corpus of three trillion tokens for language model pretraining research. *arXiv preprint arXiv:2402.00159*.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. 2022. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536.

InternLM Team. 2023. Internlm: A multilingual language model with progressively enhanced capabilities. `https://github.com/InternLM/InternLM`.

Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari S Morcos. 2023. D4: Improving llm pretraining via document de-duplication and diversification. *arXiv preprint arXiv:2308.12284*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Yue Wang, Xinrui Wang, Juntao Li, Jinxiong Chang, Qishen Zhang, Zhongyi Liu, Guannan Zhang, and Min Zhang. 2023. Harnessing the power of david against goliath: Exploring instruction data generation without using closed-source models. *arXiv preprint arXiv:2308.12711*.

BigScience Workshop, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Shaohua Wu, Xudong Zhao, Tong Yu, Rongguo Zhang, Chong Shen, Hongli Liu, Feng Li, Hong Zhu, Jiangang Luo, Liang Xu, and Xuanwei Zhang. 2021. Yuan 1.0: Large-scale pre-trained language model in zero-shot and few-shot learning.

Xiaobo Xia, Jiale Liu, Jun Yu, Xu Shen, Bo Han, and Tongliang Liu. 2023. Moderate coreset: A universal method of data selection for real-world data-efficient deep learning. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

A Appendix

A.1 Experimental Setup Details

A.1.1 Hyperparameters for Pre-training

All models in our experiments use the SwiGLU activation function, similar to LLaMA. We use the Adam optimizer [26] with hyperparameters set to $\beta_1 = 0.9, \beta_2 = 0.95, \varepsilon = 10^{-8}$, and weight decay fixed at 0.01. Additionally, we implement gradient norm clipping with a threshold of 1.0. A cosine learning rate schedule is employed, ensuring that the final learning rate equals 10% of the maximal learning rate (3e-4). We maintain a global batch size of 4M and vary warm-up steps based on different model sizes. To avoid the complications of insufficient training and the need for secondary adjustments, the preset steps for all pre-training processes are configured to be sufficiently long. For all training parameters see Table (4). We conduct model training based on the InternEvo framework (Team, 2023).

A.1.2 Hyperparameters for SFT

During the SFT phase, we use a cosine learning rate schedule, such that the final learning rate (1e-5) is equal to 33.3% of the maximal learning rate (3e-5). Meanwhile, no warmup is used, and the number of training steps is set to 328 (1 epoch). Other training parameters remain consistent with pre-training.

A.1.3 Computation Cost Estimation

In a series of pretraining experiments, models with varying parameter counts are evaluated for computational efficiency. For a model with 100M parameters, processing 100B tokens necessitates approximately 253 GPU hours. When the model size increased to 1B parameters, the same number of tokens required about 1388 GPU hours. Further scaling the model to 3B parameters, the token processing demands roughly 3472 GPU hours. Additionally, a 3B SFT model over 328 steps is completed within an estimated 47 GPU hours

A.1.4 Validation Sets Details

To thoroughly assess the potential impact on downstream tasks, we have meticulously chosen three unique validation datasets (pile validation sets, downstream task validation sets, and synthetic validation set), each tailored to a specific domain.

- Pile validation sets (Gao et al., 2020), including Pile-arXiv, Pile-books, Pile-OpenWebText2, and Pile-Wikipedia. These subsets are used to test the model's language modeling capabilities across a variety of knowledge-intensive tasks:

- Downstream task validation sets, which simply join prompt with a right answer from downstream benchmarks (see 4.2). These validation sets are designed to evaluate the language modeling capabilities across a variety of downstream benchmarks.

- Synthetic data validation set, including the Tiny-Story dataset (Eldan and Li, 2023). This type of validation set is primarily designed to assess a model's language modeling capabilities on synthetic texts characterized by high fluidity.

A.1.5 Downstream Tasks Details

Here, we provide a detailed description of 10 different downstream tasks in Table (5), providing insights into our model's performance in diverse linguistic contexts. We use OpenCompass (Contributors, 2023) framework to evaluate downstream tasks.

Categories	Datasets	Metric
Text Completion	StoryCloze	Acc.
Reading Comprehension	RACE-high RACE-middle	Acc.
Common-Sense QA	HellaSwag PIQA WinoGrande OpenBookQA	Acc.
Factual QA	NaturalQuestion TriviaQA	EM
Examination	MMLU	Acc.

Table 5: Downstream Benchmarks

params	dimension	n heads	n layers	sequence length	warmup steps	maximal learning rate	preset maximal training tokens
100M	768	12	12	2048	2000	6e-4	377B
1B	2048	16	20	2048	2000	3e-4	377B
3B	3200	32	26	2048	2500	3e-4	1.1T

Table 4: Hyperparameters Setting for Pre-training Models of Different Sizes

A.2 Proxy Metric Ranking Correlation on All Validation Sets

Here we present the ranking correlations of proxy metrics on all validation sets, including 100M pre-trained model vs. 1B pre-trained model and also 1B pre-trained model vs. 3B pre-trained model.

A.2.1 100M Pre-trained vs. 1B Pre-trained

The data presented in Figure 10 show a general trend where a lower PPL in the 100M model on the validation set leads to lower PPL in the corresponding 1B model.

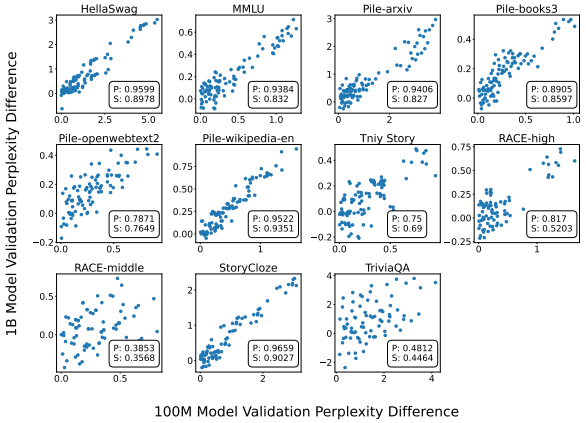

Figure 10: Validation Perplexity Difference Comparison Between 100M and 1B Model With "P" for Pearson Correlation Coefficients and "S" for Spearman Correlation Coeficients

A.2.2 1B Pre-trained vs. 3B Pre-trained

The data presented in Figure 11 show a general trend where a lower PPL in the 1B model on the validation set leads to lower PPL in the corresponding 3B model.

A.2.3 3B Pre-trained PPL vs. 3B SFT Metric

Specifically, to address the significant variance in downstream task performance, we enhance robustness by evaluating multiple checkpoints for the same experiment, with training steps ranging from 200 to 300 billion tokens, across 25 groups. So these hyperparameters are paired to compare the PPL differences in the 3B model against the differences in downstream metrics, resulting in 300 paired experiments on each of the seven validation

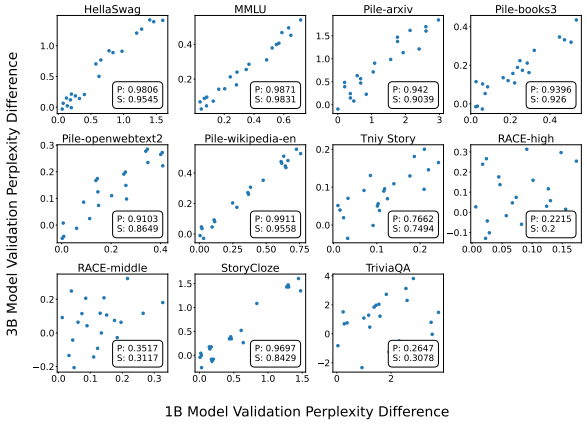

Figure 11: Validation Perplexity Difference Comparison Between 1B and 3B Model With "P" for Pearson Correlation Coefficients and "S" for Spearman Correlation Coeficients

sets. A value approaching -1 indicates a strong negative correlation, suggesting that a smaller PPL in different 3B models on the validation set correlates with higher downstream task metrics. To further mitigate the issue of large variances, we adopt the DBSCAN method to filter out outliers, obtaining non-outlier Pearson and Spearman correlation coefficients. As depicted in Figure 6, a lower PPL in the 3B model on the validation set corresponds to superior performance on downstream tasks. For most tasks, smaller models can effectively predict the performance of larger models on downstream tasks. Some tasks exhibit greater variance in downstream performance, resulting in a lower correlation coefficient. Nonetheless, the graph still reveals a distinct trend: as the PPL decreases, the performance of downstream tasks improves gradually.

A.3 Comparison Between 100M & 1B Pre-trained Models on All Validation Sets

Here we present 3 groups of comparison between 100M and 1B pre-trained models, with each group using different data selection methods: moderate loss filtering, Wikipedia & Web classifier, and Ad classifier.

A.3.1 Data Selection via Moderate Loss Filtering

We utilize LLaMA2-7B for dataset scoring and adopted a strategy of remaining mid-range data for

comparative experiments (Marion et al., 2023b). We evaluate the effects of no filtering, remaining the middle 50% of all data based on loss ranking, and retaining the middle 30% of all data based on loss ranking. The respective data remaining ratios for no pruning, loss middle 50%, and loss middle 30% are 100%, 53.9%, and 32%. Figure 12 shows the perplexities of 100M pre-trained model and 1B pre-trained model, which select the data with moderate loss filtering, on all validation sets.

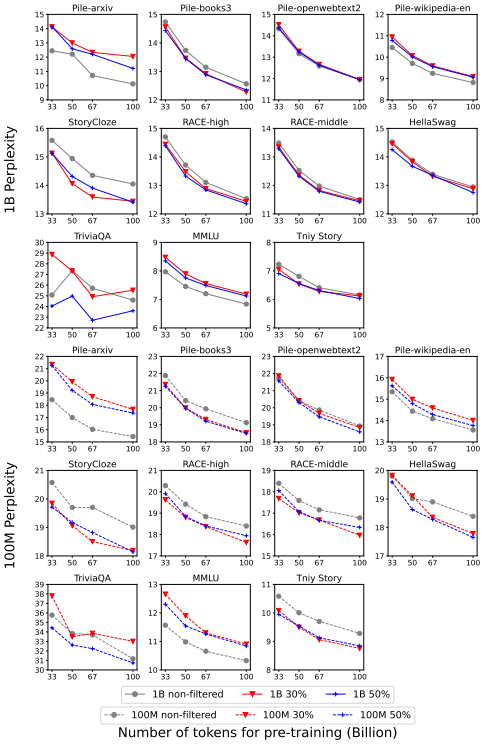

Figure 12: Validation Perplexities Comparison Between 100M & 1B Models with Moderate Perplexity

A.3.2 Data Selection via Wikipedia & Web Classifier

We employ a quality classifier trained with RedPajama. Although a threshold of 0.25 is recommended to filter out low-quality data, we compare the experimental effects of four sets of thresholds (0, 0.025, 0.075, 0.25). The data remaining rates of no pruning, threshold 0.025, threshold 0.075, and threshold 0.25 are 100%, 78.6%, 63.4%, and 42%. Figure 13 shows the perplexities of 100M pre-trained model and 1B pre-trained model, which select the data with Wikipedia & Web classifier, on all validation sets.

A.3.3 Data Selection via Ad Classifier

When evaluating the effectiveness of our BERT classifier, we employ a bootstrap method, sampling

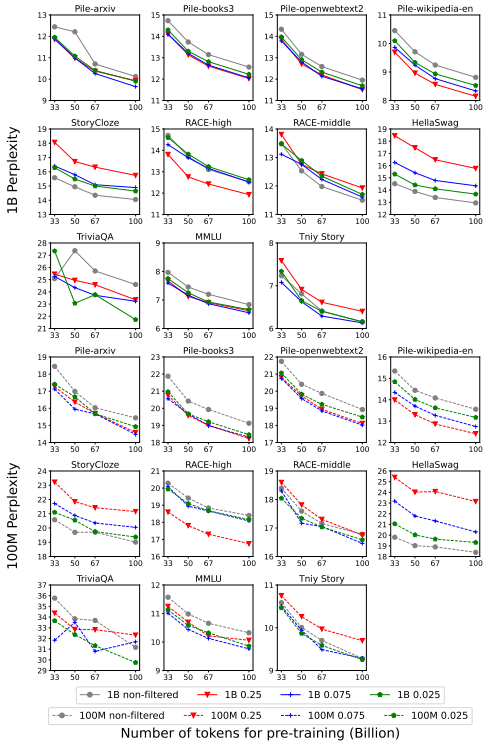

Figure 13: Validation Perplexities Comparison Between 100M & 1B Models with Wikipedia & Web

1000 times, with each time randomly selecting 50% of the data to calculate precision and recall values at different thresholds. The Precision-Recall curve for BERT training, complete with confidence intervals, is shown in 15, demonstrating our classifier's effectiveness in identifying ads, closely mirroring human judgment.

Furthermore, we try different thresholds(0.4, 0.6, 0.8, 0.9 and 0.95) for our BERT advertising classifier, which outputs a probability of a text being non-ad data. Not only do we include data remaining ratios under these thresholds in Table (6), but we also take the precisions and recalls of ad and non-ad prediction into account, so that we could make the best choice for the threshold of ad classification. Figure 16 shows the perplexities of 100M pre-trained model and 1B pre-trained model, which select the data with ad classifier, on all validation sets.

Threshold	Non-ad Precision	Non-ad Recall	Ad Precision	Ad Recall	Data Remaining
0	71.4%	100.0%	–	0.0%	100%
0.4	80.0%	96.6%	82.1%	39.7%	88.7%
0.6	86.2%	94.5%	81.8%	62.1%	82.9%
0.8	89.7%	89.7%	74.1%	74.1%	73%
0.9	91.9%	86.2%	70.2%	81.0%	64.1%
0.95	95.1%	80.0%	64.2%	89.7%	55.2%

Table 6: Data Remaining Ratio, Precision and Recall Under Different Non-ad Probability Thresholds

Figure 14: The form and instructions presented to human evaluators

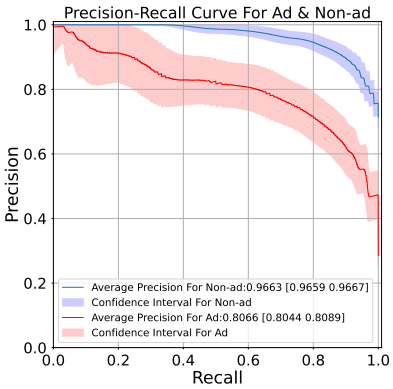

Figure 15: Effectiveness of Ad Classifier

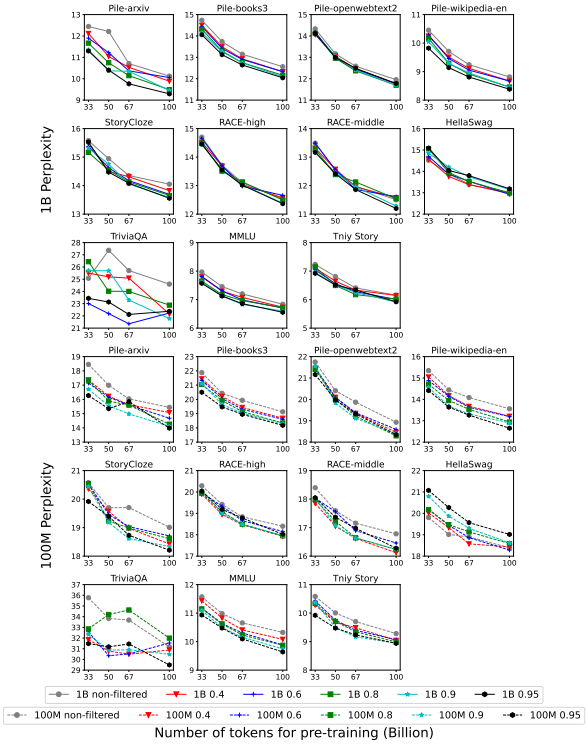

Figure 16: Validation Perplexities Comparison Between 100M & 1B Models with Ad Filtering