CoLoR-Filter: Conditional Loss Reduction Filtering for Targeted Language Model Pre-training

David Brandfonbrener
Kempner Institute at Harvard University

Hanlin Zhang
Harvard University

Andreas Kirsch
University of Oxford

Jonathan Richard Schwarz
Harvard University

Sham Kakade
Kempner Institute at Harvard University

Abstract

Selecting high-quality data for pre-training is crucial in shaping the downstream task performance of language models. A major challenge lies in identifying this optimal subset, a problem generally considered intractable, thus necessitating scalable and effective heuristics. In this work, we propose a data selection method, CoLoR-Filter (Conditional Loss Reduction Filtering), which leverages an empirical Bayes-inspired approach to derive a simple and computationally efficient selection criterion based on the relative loss values of two auxiliary models.

In addition to the modeling rationale, we evaluate CoLoR-Filter empirically on two language modeling tasks: (1) selecting data from C4 for domain adaptation to evaluation on Books and (2) selecting data from C4 for a suite of downstream multiple-choice question answering tasks. We demonstrate favorable scaling both as we subselect more aggressively and using small auxiliary models to select data for large target models. As one headline result, CoLoR-Filter data selected using a pair of 150m parameter auxiliary models can train a 1.2b parameter target model to match a 1.2b parameter model trained on 25b randomly selected tokens with 25x less data for Books and 11x less data for the downstream tasks.

Code: `https://github.com/davidbrandfonbrener/color-filter-olmo`

Filtered data: `https://huggingface.co/datasets/davidbrandfonbrener/color-filtered-c4`

1 Introduction

The content of the data that a language model is trained on can have profound effects on its performance and the efficiency of the training process [Rae et al., 2021, Longpre et al., 2023, Penedo et al., 2023, Soboleva et al., 2023, Li et al., 2024]. But it remains an open research question how to decide which data to include in the training set. In this paper, we analyze a family of loss-based approaches for targeted selection of pre-training data, propose a simple approach that outperforms existing methods, and provide some preliminary evidence of favorable scaling properties.

To formulate the data selection problem, we first need to specify an objective that quantifies whether the selected data is good. Defining this objective requires evaluating a pre-trained language model, which is an area of active research [Gao et al., 2023, Magnusson et al., 2023, Engstrom et al., 2024, Chang et al., 2024]. For this paper, we will take the goal to be to maximize performance on a set of downstream tasks. Since the preferred metrics on a given set of tasks are not necessarily the same nor amenable to direct optimization, we consider the likelihood of sequences sampled from the downstream tasks as a proxy objective. With this objective, we now have a straightforward goal: given a very large corpus of sequences and a small amount of high-quality data from a set of downstream tasks, we want to select a subset from the corpus so that training on the selected data

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

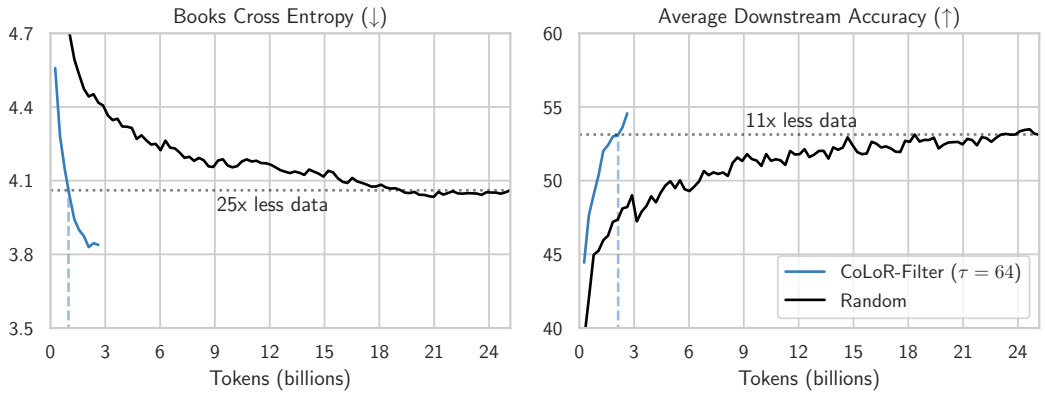

Figure 1: Learning curves for 1.2 billion parameter language models trained on data selected by CoLoR-Filter using smaller 150 million parameter auxiliary models for two different target distributions. (Left) We target and evaluate loss on Books, lower is better. (Right) We target and evaluate accuracy on a suite of 8 downstream tasks from [Groeneveld et al., 2024], higher is better. In both cases, test data is held out from the data used by CoLoR-Filter to guide selection. τ is the subset size multiplier denoting the number of examples considered for each selected data point. The CoLoR-Filter line terminates when we run out of data in C4 ($\approx$175b possible tokens).

maximizes likelihood on the downstream tasks. Then we can also test performance on the tasks under their preferred metrics.

From this objective, we derive an algorithm dubbed CoLoR-Filter (Conditional Loss Reduction Filtering). In Section 2 we derive this method by applying Bayes' rule and approximate empirical Bayes to the downstream likelihood objective. The resulting method is simple and intuitive: each sequence is scored by the difference in likelihood between a "prior" model and a "conditional" model that results from fine-tuning the prior model on the downstream data. Sequences that are more likely under the fine-tuned model are good. We also compare this algorithm to prior work (e.g., [Mindermann et al., 2022]) and discuss computational costs.

To evaluate our method, we consider two tasks. First, in Section 5, we consider a semi-synthetic task where the downstream task is language modeling on Books. Given access to C4 [Raffel et al., 2020] as potential pre-training data and a small (25 million tokens) sample of data from Books, we use CoLoR-Filter and a variety of baselines to select 3 billion tokens. We find that data selected by CoLoR-Filter can substantially outperform models trained on 8x as much randomly chosen data. Second, in Section 6, we consider a suite of 8 downstream multiple-choice tasks from Groeneveld et al. [2024]. As downstream data we take the training sets of the tasks, but we evaluate accuracy on the held-out test sets. We again find that selecting with CoLoR-Filter outperforms training on 8x as much randomly selected data. Moreover, in both tasks, performance scales smoothly with the hyperparameter τ that governs how aggressively we select the data, suggesting that further scaling would yield further improvements.

In addition to finding that CoLoR-Filter can select good subsets of data, we also consider the computational cost of the selection procedure itself. CoLoR-Filter only requires running inference of the two auxiliary models to select data. This is computationally beneficial compared to online methods like RHOLoss [Mindermann et al., 2022] since inference is cheaper than training and is entirely parallelizable. To maximize the computational benefits we also show that data selected with a small (150 million parameter) model can be transferred to a larger (1.2 billion parameter) model. Results are shown in Figure 1, showing substantial efficiency improvements.

2 Setting and Derivations

Assume that we are given a large pre-training dataset D_{train}, a small downstream dataset D_{down} from the downstream task(s) of interest, and a "prior" dataset D_{prior} we can use as prior knowledge (in practice we often just sample from D_{train}). We will assume for all practical purposes that D_{train}

is infinite and training proceeds in the "online" or "single pass" setting where we do not repeat data points. Our goal is to choose a subset $S \subset D_{\text{train}}$ of a fixed size $|S| = n$ that minimizes the downstream loss (maximizes the downstream likelihood).

This section introduces our CoLoR-Filter algorithm, inspired by and building upon the RHOLoss approach from prior work [Mindermann et al., 2022, Evans et al., 2023]. We also discuss related algorithms applicable to this setting such as DSIR [Xie et al., 2023] and DSDM [Engstrom et al., 2024]. Additional related work is discussed further in Section 3.

2.1 Bayesian Data Selection

Our objective can be formulated as a Bayesian optimization problem, where the goal is to select a set S so as to maximize the posterior probability of D_{down}, i.e.

$$\min_{S \subset D_{\text{train}}, |S|=n} -\log \Pr(D_{\text{down}}|S), \tag{1}$$

where $\Pr(D_{\text{down}}|S)$ is the posterior probability. Applying Bayes rule we get:

$$\min_{S \subset D_{\text{train}}, |S|=n} -\log \Pr(S|D_{\text{down}}) + \log \Pr(S) - \log \Pr(D_{\text{down}}) \tag{2}$$

Note that the last term does not depend on S, so it can be ignored when optimizing over S. Introducing a prior over model parameters θ, we get:

$$\min_{S \subset D_{\text{train}}, |S|=n} \underbrace{-\log \int_\theta \Pr(S|\theta) \Pr(\theta|D_{\text{down}})}_{\text{"conditional"}} + \underbrace{\log \int_\theta \Pr(S|\theta) \Pr(\theta)}_{\text{"marginal"}} \tag{3}$$

We will refer to the two terms as the conditional and marginal terms, respectively.[1] Note that the conditional and marginal terms together make up the negative pointwise mutual information between the selected and downstream data, which has deep connections to prior work on active learning and active sampling [Lindley, 1956, Moore and Lewis, 2010, Houlsby et al., 2011, Bickford Smith et al., 2023, Kirsch, 2023, Rainforth et al., 2024].

2.2 CoLoR-Filter

Given that we have access to prior knowledge from the dataset D_{prior}, we can replace the uninformed prior over θ with an empirical Bayes prior that conditions on D_{prior} to obtain:

$$\min_{S \subset D_{\text{train}}, |S|=n} -\log \int_\theta \Pr(S|\theta) \Pr(\theta|D_{\text{down}}, D_{\text{prior}}) + \log \int_\theta \Pr(S|\theta) \Pr(\theta|D_{\text{prior}}) \tag{4}$$

As this integration is still intractable, we now make our main simplifying assumption which is to replace this integration over parameters by a point estimate:

$$\approx \min_{S \subset D_{\text{train}}, |S|=n} -\log \Pr(S|\theta_{\text{prior+down}}) + \log \Pr(S|\theta_{\text{prior}}), \tag{5}$$

where θ_{prior} is a model trained on D_{prior} and $\theta_{\text{prior+down}}$ is a model trained on both D_{prior} and D_{down} (in practice, we use a model that is pre-trained on D_{prior} fine-tuned on D_{down}).

Moreover, this approximation leads to computational benefits by avoiding the full combinatorial optimization of subset selection. In particular, once we condition on a single model θ, and assuming the distribution over points $x \in S$ is independent, i.e. $\Pr(S|\theta) = \prod_{x \in S} \Pr(x|\theta)$, we have:

$$\min_{\{x_1,\ldots,x_n\} \subset D_{\text{train}}} -\log \prod_{i=1}^{n} \Pr(x_i|\theta_{\text{prior+down}}) + \log \prod_{i=1}^{n} \Pr(x_i|\theta_{\text{prior}}) \tag{6}$$

which simplifies to:

$$\min_{\{x_1,\ldots,x_n\} \subset D_{\text{train}}} \sum_{i=1}^{n} -\log \Pr(x_i|\theta_{\text{prior+down}}) - (-\log \Pr(x_i|\theta_{\text{prior}})) \tag{7}$$

[1] Prior work [Mindermann et al., 2022, Evans et al., 2023] has referred to the models that estimate these two terms as the "reference" and "learner" or "actor", respectively. We opt for the names conditional and marginal for clarity in connections to the Bayesian viewpoint.

This gives our CoLoR-Filter criteria that we use to select data. This optimization selects the points with the largest conditional loss reduction (CoLoR), i.e. the points where the negative log-likelihood loss of the conditional model $\theta_{\text{prior+down}}$ is lower than the marginal model θ_{prior}. Intuitively, this selects data points that are more likely under the conditional model than the marginal model.

A note on data diversity. While the factorization that results from our point estimate of the parameters is computationally convenient, it makes an important simplifying assumption. In particular, the CoLoR-Filter objective no longer encourages the selection of a diverse dataset, as scores are applied independently to each point. In practice, this is remedied by a few considerations: (1) we can run CoLoR-Filter on a corpus that has already been deduplicated to prevent degenerate duplications, (2) for large n, we must select many different data points, and (3) each datapoint is itself a sequence that may contain diverse signal across tokens. We should also note this is not a unique property of CoLoR-Filter and also happens in other methods that do offline scoring like DSDM and DSIR. We defer a detailed discussion of the nuances of this issue to Appendix C.

2.3 Related Algorithms

Connection to importance sampling. Since the CoLoR-Filter objective is written as a difference of logs, it can also be written as a log of the ratio between probabilities under $\theta_{\text{prior+down}}$ and θ_{prior}. If data were actually sampled from θ_{prior}, then this ratio would be the importance weight needed to reweight samples so that they are from the model defined by $\theta_{\text{prior+down}}$. Note that DSIR [Xie et al., 2023] directly attempts to perform importance sampling from D_{train} to D_{down} instead of optimizing performance on the downstream data. Thus, DSIR ends up with a somewhat related algorithm except in DSIR: (1) there is no language model, just features of a full data point (hashed n-grams), and (2) the algorithm samples rather than optimizes.

Connections to DSDM. Another closely related approach is DSDM [Engstrom et al., 2024] which uses a TRAK Datamodel estimator [Ilyas et al., 2022, Park et al., 2023] to score datapoints and then selects the top-n points. The motivation and setting of DSDM are similar to CoLoR-Filter, but DSDM relies on TRAK which constructs a linear approximation of the influence that data points have on each other. Instead, CoLoR-Filter operates directly in function space by comparing the loss between models directly rather than relying on linear approximations or Datamodels [Ilyas et al., 2022].

Connections to RHO-down. CoLoR-Filter is inspired by and builds on the RHOLoss approach introduced in prior work [Mindermann et al., 2022] with subtle but significant differences in the setting: the original RHO paper focuses on cases where the hold-out data is sampled from the same distribution as D_{train} over multiple epochs of training. In contrast, we focus on selecting data to target downstream distributions that are different from D_{train} and where we only take a single pass over the data. Here, we derive a straightforward adaptation of RHOLoss to our setting, which we call RHO-down.

We now derive RHO-down in our setting, aiming to illustrate the connections between RHO-down and CoLoR-Filter. First, RHO-down approximates the full subset selection problem from Equation (3) by a greedy (sequential) approximation where samples are added to S one (batch) at a time. Using a batch size of 1, the ith-sample would be ideally added according to the following criterion:

$$\approx \min_{x_i \in D_{\text{train}}} -\log \int_\theta \Pr(x_i|\theta) \Pr(\theta|D_{\text{down}}, x_{<i}) + \log \int_\theta \Pr(x_i|\theta) \Pr(\theta|x_{<i}), \tag{8}$$

where i ranges from 1 to n sequentially. RHO-down then uses a point estimate of the parameters (as we do in CoLoR-Filter):

$$\approx \min_{x_i \in D_{\text{train}}} -\log \Pr(x_i|\theta_{\text{down}+x_{<i}}) + \log \Pr(x_i|\theta_{x_{<i}}) \tag{9}$$

Finally, the RHO-down authors found that updating the conditional term to depend on $x_{<i}$ was unstable, so they instead approximate this by a fixed model θ_{down}:

$$\approx \min_{x_i \in D_{\text{train}}} -\log \Pr(x_i|\theta_{\text{down}}) + \log \Pr(x_i|\theta_{x_{<i}}). \tag{10}$$

Note that while both CoLoR-Filter and RHO-down approximate the posterior over parameters with a point estimate, RHO-down makes a few additional approximations. This is largely a result of

RHO-down attempting to increase data diversity by using a sequential approach to selection that conditions on the previously selected data $x_{<i}$. This is an understandable goal, but it introduces more approximations, can cause instability by creating a non-stationary data distribution, and is computationally expensive since the data selection is no longer parallelizable. A continued discussion of the pros and cons of online selection is in Appendix C.

RHO-down + prior. We also consider a version of the algorithm that we call "RHO-down + prior" that replaces $D_{\text{down}}, \theta_{\text{down}}$ in the RHO-down algorithm with $D_{\text{prior}} \cup D_{\text{down}}, \theta_{\text{prior+down}}$ to incorporate the prior information. This corresponds to conditioning on both D_{prior} and D_{down} instead of only D_{down}. Intuitively, this method can better leverage stronger features learned on the larger D_{prior} to integrate the information from the small D_{down}.

3 Further Related Work

We now discuss some related work, more broadly, with regards to active learning and data curation.

Active & Curriculum learning. Our formulation of data selection has connections to classic and deep active learning [Houlsby et al., 2011, Bickford Smith et al., 2023, Kirsch, 2023], which are deeply rooted in optimal Bayesian experimental design [Lindley, 1956, Rainforth et al., 2024], whose goal is to select a set of experiments to optimize certain information criteria [Pukelsheim, 2006] such as maximally reducing the uncertainty about model parameters. Various acquisition functions are proposed in deep learning regimes [Sener and Savarese, 2018, Ash et al., 2019, 2021] and most of them focus on label-efficient image classification. Another line of recent techniques share deep methodological connections but emphasize the sub-selection of available data during training (rather than the collection of additional examples typically considered in active learning) and could thus be classified as curriculum learning [e.g. Graves et al., 2017]. Among them, RHOLoss [Mindermann et al., 2022] seeks to select data based on the hold-out reference dataset from the same distribution as the training data. It has been later implemented in continual pre-training [Lin et al., 2024] and vision domains [Evans et al., 2023, Tack et al., 2024].

Data curation practices in pre-training. Though large-scale public web-crawled data are common data sources for pre-training models, low-quality, toxic, and uninformative content that can prevent successful pre-training is prevalent [Wenzek et al., 2020, Elazar et al., 2023, Sorscher et al., 2022, Allen-Zhu and Li, 2024]. Therefore, practitioners design sophisticated data pre-processing pipelines such as filtering [Brown et al., 2020], deduplication [Lee et al., 2022], and mixing [Touvron et al., 2023a,b] to improve the data quality. Due to the immense scale, state-of-the-art pre-training datasets usually depend on simple heuristic filters [Raffel et al., 2020, Rae et al., 2021, Computer, 2023] (e.g., URL, length, n-gram perplexity, fastest classifiers) that can be parallelized across CPU nodes. Besides the above rule-based filtering, model-based filtering concerns using machine learning models to score and filter data, which has been proven to be effective in vision and vision-text domains [Schuhmann et al., 2022, Abbas et al., 2023, Fang et al., 2023]. Such approaches usually leverage a given trustworthy data source like Wikipedia or Books as the reference and contrast the raw data with it. Due to computational cost, models are often designed to be small such as n-gram [Xie et al., 2023], single-layer neural networks [Joulin et al., 2017, Brown et al., 2020], k-means clustering [Tirumala et al., 2024]. There is also a growing line of work illustrating that data quality is important in shaping model training from a variety of perspectives, such as increasing data scale [Hoffmann et al., 2022, Meta, 2024] and using synthetic data [Gunasekar et al., 2023].

4 Algorithms

4.1 From Derivations to Practical Algorithms

In our experiments, we will consider four algorithms based on the above derivations. In this section we go through each of these in turn.

CoLoR-Filter. Our proposed algorithm is presented formally in Algorithm 1. Compared to the derivation, the main difference is the introduction of τ, a hyperparameter that acts as a compute-performance trade-off controlling how expensive and aggressive the data selection is. Rather than

Algorithm 1 CoLoR-Filter

Require: Prior data D_{prior}, downstream data D_{down}, training data D_{train}, budget n, subset size multiplier τ

1: Pre-train θ^{marg} on D_{prior}
2: fine-tune to get θ^{cond} on D_{down} initialized from θ^{marg}
3: Select a random subset D_τ of size τn from D_{train}
4: Select data:
$$S = \texttt{bottom-}n_{x \in D_\tau} - \log \Pr(x|\theta^{\text{cond}}) + \log \Pr(x|\theta^{\text{marg}})$$
5: **return** Selected dataset S to train θ on.

selecting data from all of D_{train}, we take a random subset D_τ of size τn. Thus, larger τ subselect more aggressively, but at the cost of more computation. A full discussion of this cost is in Section 4.2.

Conditional only. As an ablation of CoLoR-Filter, we follow prior work [Evans et al., 2023] and include a baseline that only uses the conditional model to select data. Essentially, this is CoLoR-Filter if we always assume that $\log \Pr(x|\theta^{\text{marg}}) = 0$ in Line 4 of Algorithm 1.

Algorithm 2 RHO-down

Require: Downstream data D_{down}, train data D_{train}, budget n, subset size multiplier τ, batch size b

1: Train θ^{cond} on D_{down}
2: Initialize a random θ_1^{marg} and $S = \varnothing$
3: **for** $t \in [1, \ldots, n/b]$ **do**
4: Randomly select a batch $B_t \subset D_{\text{train}}$ of size τb
5: Select data:
$$\bar{B}_t = \texttt{bottom-}b_{x \in B_t} - \log \Pr(x|\theta^{\text{cond}}) + \log \Pr(x|\theta_t^{\text{marg}})$$
6: $S = S \cup \bar{B}_t$
7: Update θ_t^{marg} to $\theta_{t+1}^{\text{marg}}$ by training on $\bar{B}_t$
8: **end for**
9: **return** Selected dataset S to train θ on.

RHO-down. We present a practical variant of RHO-down in Algorithm 2 based on the derivation presented in Section 2. The main changes to make a practical algorithm are (1) the introduction of τ as in CoLoR-Filter, and (2) performing the algorithm batch-wise instead of using single data points.

RHO-down + Prior. We can also incorporate the prior data D_{prior} into Algorithm 2 by simply replacing Line 1 where θ^{cond} is trained on D_{down} with a procedure where we first pre-train θ^{cond} on D_{prior} and then fine-tune it on D_{down}.

4.2 Computational Cost

To evaluate the computational cost of the various algorithms, we use units of "model forwards" per token where we assume that a backward pass is twice as expensive as a forward pass [Fleuret, 2023]. Note that our 150m models take about 5e8 FLOPs per model forward of a single token [Hoffmann et al., 2022, Casson, 2023]. The cost of running the selection algorithms depends on m, n, τ and L defined as follows: m is the size of the prior data D_{prior}, n is the size of the selected dataset S, τ is the hyperparameter controlling how aggressively we subselect data. Note that we assume that $|D_{\text{down}}|$ is so small that the cost of training a model on D_{down} is negligible towards the total cost (and all the methods we consider just fine-tune a model once on D_{down}). We will also be careful to note when computation can be done in parallel before training versus computation that must happen serially during a training run. Offline algorithms like CoLoR-Filter can take advantage of parallelism to improve efficiency. In this section, we go through each method in turn and aggregate the computational costs in table 1.

Scale transfer. We also include another parameter L to cover the case where we select data using small models and use it to train a larger model [Evans et al., 2023]. Specifically, L is the ratio of cost of one model forward of the *large* target model compared to the small auxiliary models used for data

Table 1: Compute cost of the various algorithms measured in "model forwards". The total cost of selection and training on the selected data is the sum of all costs across a row. The variables are $m = |D_{\text{prior}}|$, $n = |S|$, τ is a hyperparameter that controls how aggressively we subselect, and L is a multiplier of the cost of model forwards between the selection model(s) and the target model (approximately the ratio of parameter counts between the models).

Method	Prior cost	Serial cost	Parallel cost	Training cost
CoLoR-Filter	$3m$	0	$2\tau n$	$3nL$
Conditional Only	$3m$	0	τn	$3nL$
RHO-down	0	$\tau n + 2n$	τn	$3nL$
RHO-down + Prior	$3m$	$\tau n + 2n$	τn	$3nL$
Random	0	0	0	$3nL$

selection. For example, in our experiments, when we use 150 million parameter models to select data and then train a 1.2 billion parameter model on the resulting data, then $L \approx 5.5$[2]. Training thus costs $3nL$ across all methods since we run a forward and backward for the large model on all n sequences.

CoLoR-Filter. The cost of selection is $2\tau n$ forward passes. But, this selection process is *entirely* parallelizable. Training the prior model costs $3m$ forwards since $|D_{\text{prior}}| = m$. And training a model on the selected data costs $3nL$ forward passes. So the total cost is $3m + 2\tau n + 3nL$, but the $2\tau n$ scoring computation can be done in parallel.

Conditional Only. The conditional-only method is almost the same as CoLoR-Filter, except we only need τn forward passes for selection since we only run one model over the data. The cost is thus $3m + \tau n + 3nL$, with τn being parallelizable.

RHO-down. The cost of selection is still $2\tau n$ forward passes. Then we need an additional $2n$ to backward the output model (since the forward is already handled during scoring). Note that we need to evaluate the marginal model online, so it is not parallelizable, but the conditional model is fixed and can be computed offline. So, the cost is $2\tau n + 2n + 3nL$, and the τn conditional model computation can be done in parallel.

RHO-down + Prior. For the version with an added prior, we just add $3m$ cost for training the prior. Thus, the cost is $2\tau n + 2n + 3nL$ with τn parallelizable.

Overall, the methods all have comparable costs, with Conditional Only being the cheapest and RHO-down + Prior the most expensive. The main difference is that CoLoR-Filter and Conditional Only are easily parallelized while RHO-down and RHO-down + Prior are not. It should also be noted that when doing experimentation, offline methods like CoLoR-Filter also benefit from being able to re-use likelihoods multiple times, while RHO-based methods need to recompute the serial cost any time that some hyperparameter of the algorithm.

5 Domain Transfer: a Simple Testbed

5.1 Setup

Training. We train language models with 150 million non-embedding parameters using the OLMo codebase [Groeneveld et al., 2024] and following hyper-parameter choices from [Wortsman et al., 2024]. Unless otherwise noted, we use 150m models as the auxiliary models (θ^{cond}, θ^{marg}) as well as the target model θ. Full hyperparameters are described in detail in Appendix H.

We take D_{down} to be a small dataset of 25 million tokens sampled from the Project Gutenberg Books data subset of Dolma [Soldaini et al., 2024], D_{prior} to be a dataset of 3.1 billion tokens from C4 [Raffel et al., 2020], and D_{train} to be all of C4. We select a dataset S of 3.1 billion tokens (which is

[2]Even though there are 8x as many parameters in the large model, the FLOP multiplier is less since the attention computations take the same number of FLOPs regardless of parameters.

approximately the "chinchilla optimal" amount for models of this size). To get $\theta_{\text{prior+down}}$ or θ_{down}, we fine-tune or train for one epoch on D_{down}.

Evaluation. To evaluate the efficacy of our data selection, we report cross-entropy loss of next token prediction on a held-out dataset $\widetilde{D}_{\text{down}}$ from the same distribution as D_{down} (Books).

Baselines. The simplest baseline we consider is **Random** sampling, which has been shown to be a strong baseline for C4 pre-training [Engstrom et al., 2024]. We consider all four algorithms described in Section 4: **CoLoR-Filter**, **Conditional Only**, **RHO-down**, and **RHO-down + prior**. And as one extra baseline, we also include **DSIR** [Xie et al., 2023] which estimates n-gram importance weights between D_{train} and D_{down}, and similarly has a parameter like τ that controls how aggressively to subselect.

Note that while it is in a similar setting to ours, we do not include DSDM [Engstrom et al., 2024] as a baseline since there is no publicly available code and based on the appendix of that paper, it it much more computationally expensive than the methods we consider.

5.2 Results

We first run the domain transfer experiments on 150m models, sweeping across τ that controls the selected subset size. In Figure 2 we plot how the final performance scales with τ across methods. We see that CoLoR-Filter has the best scaling performance with increased τ, with no sign of saturation for $\tau = 16$. We hypothesize that by using strong models to select the data, CoLoR-Filter is able to more effectively scale to larger τ than the other methods. In Figure 7 in Appendix A, we plot the learning curves (evaluated on the held-out validation set) for the four methods introduced in Section 4. There, we see especially clean scaling for CoLoR-Filter across the entire learning curve, substantially outperforming random selection with much less data, similar to Figure 1.

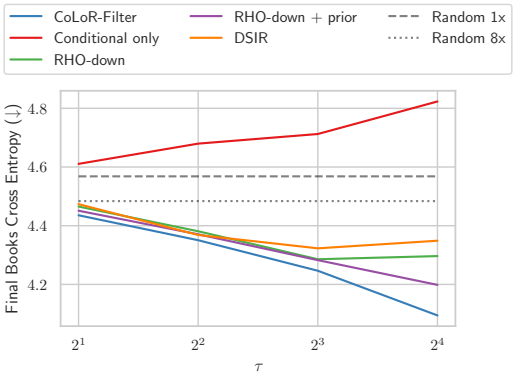

Figure 2: Scaling of final performance with τ when targeting **Books** with 150m parameter models.

Scale generalization. Finally, we also conduct an experiment in scale generalization (partially shown in Figure 1) using the data selected by our 150m auxiliary models to train a 1.2b target model. In Figure 3 we show learning curves for a sweep over τ. We still see consistent gains as we scale τ for a fixed number of training tokens. Interestingly, if we fix the total number of tokens we are *selecting from* (i.e. where the lines end when we run out of C4), then the final performance with $\tau = 32$ is better than all other values of τ. This shows how a strict subset of tokens can outperform a superset (e.g. $\tau = 16$). We should also point out here the computational savings when using CoLoR-Filter. As an example, consider $\tau = 16$ where we match the performance of 25 billion randomly selected tokens with about 1.5 billion filtered tokens. Considering the computational costs discussed above with $L = 5.5$ and measuring n in billions of tokens, the total cost for training the CoLoR-Filter model is $3m + 2\tau n + 3nL = 3 * 3.1 + 2 * 16 * 1.5 + 3 * 1.5 * 5.5 = 82$ while the cost for training on 25 billion random tokens is $3NL = 3 * 25 * 5.5 = 412.5$, illustrating a more than 5x total compute savings to achieve the same performance on Books. A full plot visualizing the cost in FLOPs for all τ is in Appendix D.

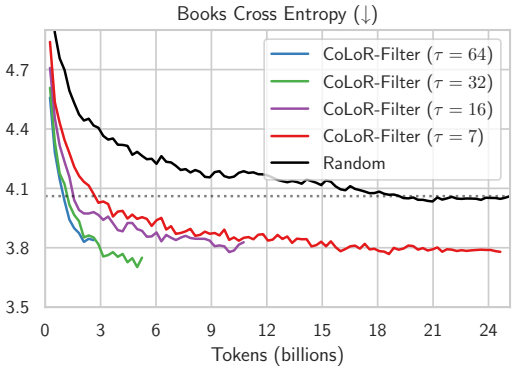

Figure 3: Scaling CoLoR-Filter with τ when training 1.2b models with data selected by 150m models. Curves end when we exhaust the data in C4.

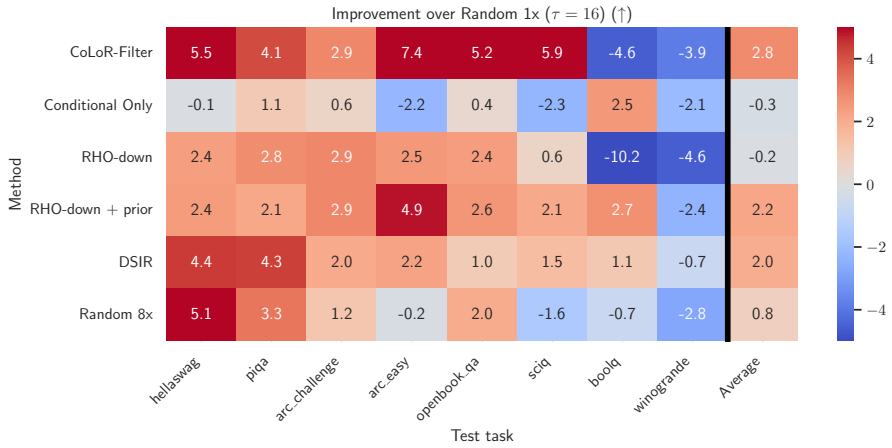

Figure 5: Performance improvement over training on an equivalent amount of random data broken down by task (except for Random 8x, which uses 8x more data). A table of results is in Appendix B.

6 Downstream Tasks

6.1 Setup

Training. We target the 8 tasks from the OLMo paper [Groeneveld et al., 2024]: Hellaswag [Zellers et al., 2019], PIQA [Bisk et al., 2020], ARC-challenge and ARC-easy [Clark et al., 2018], Openbook QA [Mihaylov et al., 2018], SciQ [Welbl et al., 2017], BoolQ [Clark et al., 2019], and Winogrande [Sakaguchi et al., 2021]. Each of these datasets has a separate train split. We use these train splits to construct D_{down} as follows: for each question we concatenate the question and the correct answer formatted as a grammatical continuation. Overall, this results in a small D_{down} dataset of 7.4 million tokens. D_{prior} and D_{train} are the same as before. And we again get $\theta_{\text{prior+down}}$ by fine-tuning θ_{prior} for one epoch on D_{down}.

Evaluation. We evaluate on held-out data from each downstream task test or validation sets (using val if test is not publicly available). We use the evaluation procedure from OLMo [Groeneveld et al., 2024] which follows [Gao et al., 2023] for evaluating these multiple-choice tasks using the rank classification approach of Brown et al. [2020]. We report aggregat perfromance across tasks as well as the task-specific performance.

Baselines. Same as in Section 5.

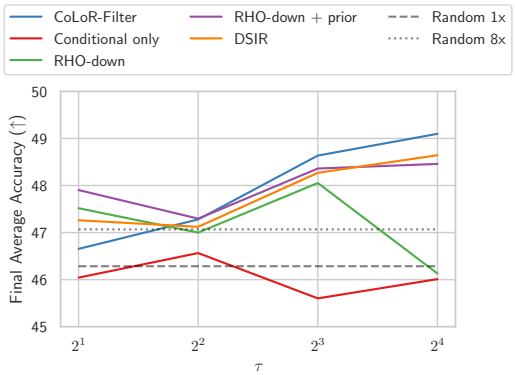

Figure 4: Final performance versus τ on the suite of downstream tasks for 150m models. CoLoR-Filter scales the best with τ.

6.2 Results

While the curves themselves are noisier now due to the noisier nature of accuracy evaluation on small datasets compared to cross entropy on a large one, the same trends hold as we saw for domain transfer to Books. CoLoR-Filter in particular is scaling the best as we increase τ. Other methods do not illustrate the same clean scaling as we increase τ, which is nearly linear on a log scale for CoLoR-Filter, as seen in Figure 4. Full learning curves are in Appendix A.

We can also look at the performance broken down by task and illustrated relative to training on an equivalent amount (3.1 billion tokens) of randomly selected data for $\tau = 16$ illustrated in Figure 5. We see especially large gains on Hellaswag, ARC easy, Openbook QA and SciQ and actually see performance decreases on BoolQ and Winogrande. However, we should note that at this scale and with all data selected from C4, we actually found BoolQ and Winogrande to be quite noisy and not even correlated with training on 8x as much random data, so it is not clear how much weight to place

on those results. Across the other tasks, the gains of CoLoR-Filter over the baselines are clear. It is an interesting direction for future work to probe more deeply into how task-dependent the gains from targeted data selection can be.

Scale generalization. We also consider scale generalization to a 1.2b target model and illustrate the full results of a sweep over τ in Figure 6. Again we find significant benefits of CoLoR-Filter across scales. A full table of per-task results is in Appendix B. Again we notice that training on a strict subset of data can outperform a larger dataset.

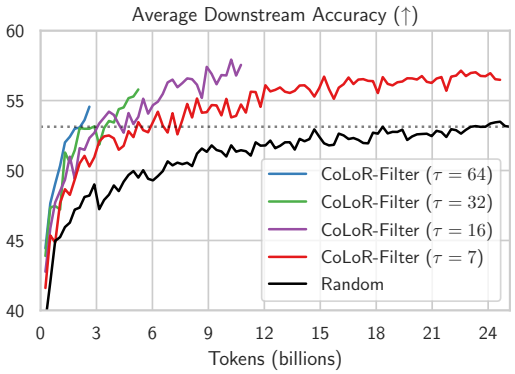

Figure 6: Scaling CoLoR-Filter with τ when training 1.2b models with data selected using smaller 150m models. Curves end when we exhaust the data in C4.

We can again do out the calculation of computational savings for $\tau = 16$. It now takes about 3 billion tokens for CoLoR-Filter to match the performance of training on 25 billion random tokens. This amounts to a total cost of $3m+2\tau n+3nL = 3*3.1+2*16*3+3*3*5.5 = 154.8$, which is still an upwards of 2.5x reduction in compute to achieve the same average performance across the suite of tasks. A full plot visualizing the cost in FLOPs for all τ is in Appendix D.

Task generalization. We can also test task generalization beyond the 8 tasks that were used to select the data on a few more tasks that test common sense reasoning [Wang et al., 2019, Socher et al., 2013, Talmor et al., 2018, Sap et al., 2019]. Results are presented in Table 2 compared to a random model trained on 10x as much data. The performance indicates that the data selected by CoLoR-Filter are not overfit to the particular evaluation tasks, but captures some general notion of good data for a range of tasks.

Table 2: Task generalization for the 1.2b models with $\tau = 64$.

Method	copa	rte	cb	sst2	commonsense qa	social iqa
Random (25b tokens)	**69.2**	48.9	42.8	46.8	**33.7**	**42.9**
CoLoR-Filter ($\tau = 64$, 2.5b tokens)	65.8	**52.6**	**46.0**	**55.8**	32.6	42.7

Note, we also conduct a few more experiments and ablations in the appendix: Appendix E considers using CoLoR-Filter in-distribution to target C4 loss, Appendix F considers applying CoLoR-Filter batchwise rather than globally, Appendix G considers finetuning on D_{down} after targeted pre-training, Appendix I inspects some of the selected and excluded examples, and Appendix J compared to FineWeb-edu [Penedo et al., 2024].

7 Discussion

While fairly simple to derive and implement, we show that CoLoR-Filter is an effective method for data selection on C4, with promising scaling behavior up to 1.2 billion models. In our experiments, CoLoR-Filter continues to improve when only using 1 out of 64 data points considered for selection and generalizes from small auxiliary models to larger target models. This opens many potential lines of research. First, while we have considered targeted pre-training, it is possible that CoLoR-Filter could be extended to fine-tuning, continual pre-training, and more general open-domain pre-training. In particular, it is an interesting open question whether the lack of an explicit consideration of data diversity hinders CoLoR-Filter in any of these settings. Second, CoLoR-Filter could be applied to more challenging domains in language like code generation or even applied beyond the language domain to other modalities. Finally, there is plenty of work to be done to make the algorithm more efficient and to test the limits of scale generalization.

Acknowledgments

HZ is supported by an Eric and Susan Dunn Graduate Fellowship. SK acknowledges support from the Office of Naval Research under award N00014-22-1-2377 and the National Science Foundation Grant under award #IIS 2229881. This work has been made possible in part by a gift from the Chan Zuckerberg Initiative Foundation to establish the Kempner Institute for the Study of Natural and Artificial Intelligence.

References

Amro Abbas, Kushal Tirumala, Dániel Simig, Surya Ganguli, and Ari S Morcos. Semdedup: Data-efficient learning at web-scale through semantic deduplication. *arXiv preprint arXiv:2303.09540*, 2023.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.3, knowledge capacity scaling laws, 2024.

Jordan Ash, Surbhi Goel, Akshay Krishnamurthy, and Sham Kakade. Gone fishing: Neural active learning with fisher embeddings. *Advances in Neural Information Processing Systems*, 34:8927–8939, 2021.

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.

Freddie Bickford Smith, Andreas Kirsch, Sebastian Farquhar, Yarin Gal, Adam Foster, and Tom Rainforth. Prediction-oriented Bayesian active learning. *International Conference on Artificial Intelligence and Statistics*, 2023.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439, 2020.

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

Adam Casson. Transformer flops, 2023. URL `https://adamcasson.com/posts/transformer-flops`.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45, 2024.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*, 2019.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Together Computer. Redpajama: an open dataset for training large language models, 2023. URL `https://github.com/togethercomputer/RedPajama-Data`.

Abhimanyu Das and David Kempe. Approximate submodularity and its applications: Subset selection, sparse approximation and dictionary selection. *Journal of Machine Learning Research*, 19(3): 1–34, 2018. URL `http://jmlr.org/papers/v19/16-534.html`.

Yanai Elazar, Akshita Bhagia, Ian Helgi Magnusson, Abhilasha Ravichander, Dustin Schwenk, Alane Suhr, Evan Pete Walsh, Dirk Groeneveld, Luca Soldaini, Sameer Singh, et al. What's in my big data? In *The Twelfth International Conference on Learning Representations*, 2023.

Logan Engstrom, Axel Feldmann, and Aleksander Madry. Dsdm: Model-aware dataset selection with datamodels, 2024.

Talfan Evans, Shreya Pathak, Hamza Merzic, Jonathan Schwarz, Ryutaro Tanno, and Olivier J Henaff. Bad students make great teachers: Active learning accelerates large-scale visual understanding. *arXiv preprint arXiv:2312.05328*, 2023.

Alex Fang, Albin Madappally Jose, Amit Jain, Ludwig Schmidt, Alexander Toshev, and Vaishaal Shankar. Data filtering networks, 2023.

François Fleuret. The little book of deep learning. *A lovely concise introduction*, page 297, 2023.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 12 2023. URL `https://zenodo.org/records/10256836`.

Alex Graves, Marc G Bellemare, Jacob Menick, Remi Munos, and Koray Kavukcuoglu. Automated curriculum learning for neural networks. In *international conference on machine learning*, pages 1311–1320. Pmlr, 2017.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hannaneh Hajishirzi. Olmo: Accelerating the science of language models, 2024.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. Textbooks are all you need, 2023.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models, 2022.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *stat*, 1050:24, 2011.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Predicting predictions from training data. *arXiv preprint arXiv:2202.00622*, 2022.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431. Association for Computational Linguistics, April 2017.

A Kirsch. *Advanced deep active learning and data subset selection: unifying principles with information-theory intuitions*. PhD thesis, University of Oxford, 2023.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8424–8445, 2022.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Gadre, Hritik Bansal, Etash Guha, Sedrick Keh, Kushal Arora, et al. Datacomp-lm: In search of the next generation of training sets for language models. *arXiv preprint arXiv:2406.11794*, 2024.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, and Weizhu Chen. Rho-1: Not all tokens are what you need, 2024.

Dennis V Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005, 1956.

Shayne Longpre, Gregory Yauney, Emily Reif, Katherine Lee, Adam Roberts, Barret Zoph, Denny Zhou, Jason Wei, Kevin Robinson, David Mimno, et al. A pretrainer's guide to training data: Measuring the effects of data age, domain coverage, quality, & toxicity. *arXiv preprint arXiv:2305.13169*, 2023.

Ian Magnusson, Akshita Bhagia, Valentin Hofmann, Luca Soldaini, Ananya Harsh Jha, Oyvind Tafjord, Dustin Schwenk, Evan Pete Walsh, Yanai Elazar, Kyle Lo, Dirk Groeneveld, Iz Beltagy, Hannaneh Hajishirzi, Noah A. Smith, Kyle Richardson, and Jesse Dodge. Paloma: A benchmark for evaluating language model fit, 2023.

Meta. Llama 3, 2024.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*, 2018.

Sören Mindermann, Jan M Brauner, Muhammed T Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, et al. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning*, pages 15630–15649. PMLR, 2022.

Robert C Moore and William Lewis. Intelligent selection of language model training data. In *Proceedings of the ACL 2010 conference short papers*, pages 220–224, 2010.

George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14:265–294, 1978. URL `https://api.semanticscholar.org/CorpusID:206800425`.

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. Trak: Attributing model behavior at scale. *arXiv preprint arXiv:2303.14186*, 2023.

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

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, Thomas Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. *arXiv preprint arXiv:2406.17557*, 2024.

Friedrich Pukelsheim. *Optimal design of experiments*. SIAM, 2006.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

Tom Rainforth, Adam Foster, Desi R Ivanova, and Freddie Bickford Smith. Modern bayesian experimental design. *Statistical Science*, 39(1):100–114, 2024.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. Socialiqa: Commonsense reasoning about social interactions. *arXiv preprint arXiv:1904.09728*, 2019.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

Daria Soboleva, Faisal Al-Khateeb, Robert Myers, Jacob R Steeves, Joel Hestness, and Nolan Dey. SlimPajama: A 627B token cleaned and deduplicated version of RedPajama. `https://www.cerebras.net/blog/slimpajama-a-627b-token-cleaned-and-deduplicated-version-of-redpajama`, 2023. URL `https://huggingface.co/datasets/cerebras/SlimPajama-627B`.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/D13-1170`.

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

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

Jihoon Tack, Subin Kim, Sihyun Yu, Jaeho Lee, Jinwoo Shin, and Jonathan Richard Schwarz. Learning large-scale neural fields via context pruned meta-learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*, 2018.

Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari Morcos. D4: Improving llm pretraining via document de-duplication and diversification. *Advances in Neural Information Processing Systems*, 36, 2024.

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

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023b.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32, 2019.

Johannes Welbl, Nelson F Liu, and Matt Gardner. Crowdsourcing multiple choice science questions. *arXiv preprint arXiv:1707.06209*, 2017.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Édouard Grave. Ccnet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, 2020.

Mitchell Wortsman, Peter J Liu, Lechao Xiao, Katie E Everett, Alexander A Alemi, Ben Adlam, John D Co-Reyes, Izzeddin Gur, Abhishek Kumar, Roman Novak, Jeffrey Pennington, Jascha Sohl-Dickstein, Kelvin Xu, Jaehoon Lee, Justin Gilmer, and Simon Kornblith. Small-scale proxies for large-scale transformer training instabilities. In *The Twelfth International Conference on Learning Representations*, 2024.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. Data selection for language models via importance resampling. *Advances in Neural Information Processing Systems*, 36: 34201–34227, 2023.

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

A Learning curves for 150m models

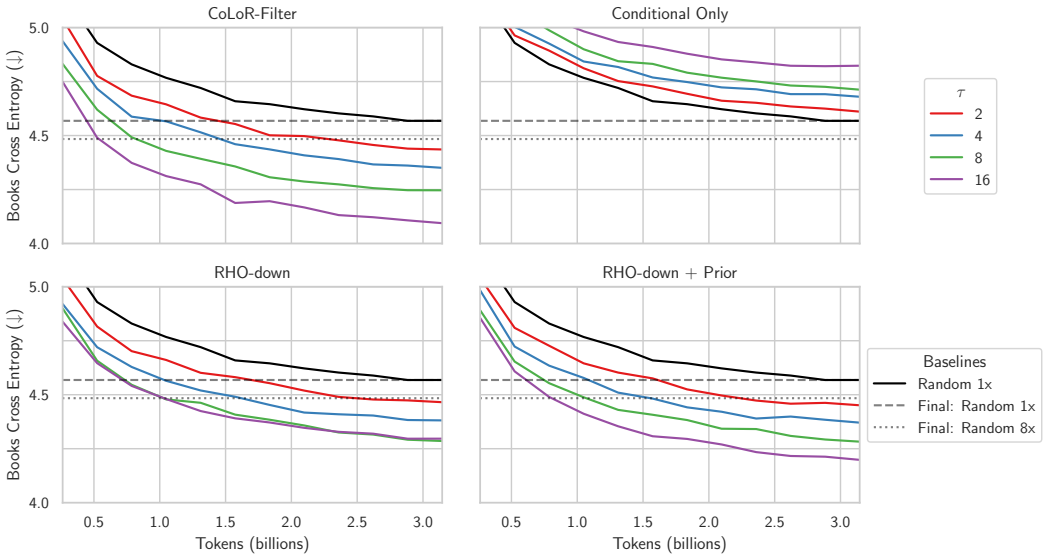

Figure 7: Sweeping over τ when targeting **Books** from C4 for 150m models.

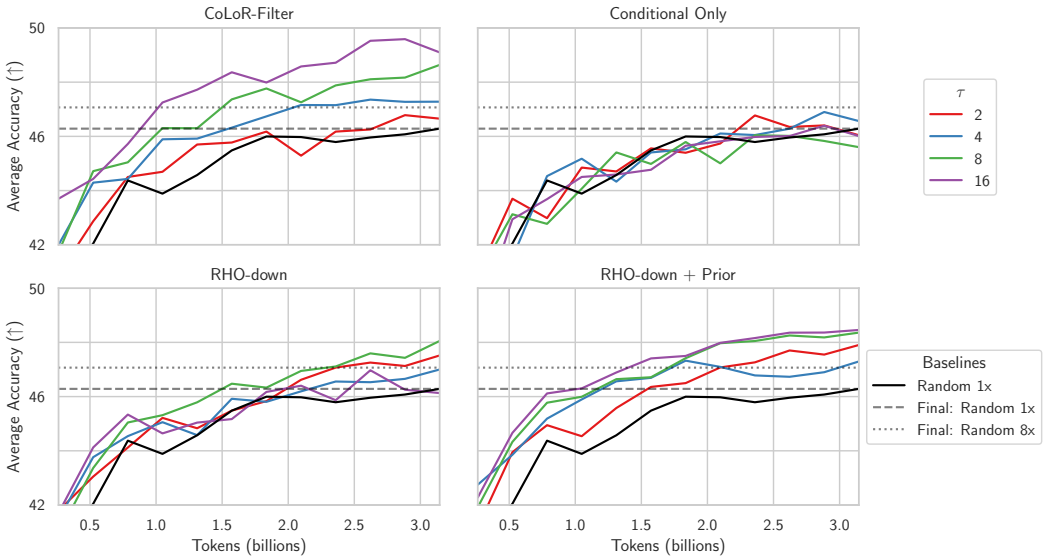

Figure 8: Sweeping over τ and measuring average performance on all downstream tasks for 150m models.

B Tables of downstream results

Table 3: Performance for all tasks for 150m models for data selection with $\tau = 16$.

Method	hella-swag	piqa	arc-c	arc-e	open-book qa	sciq	boolq	wino-grande	Avg
Random 1x	33.2	64.5	22.4	44.4	26.8	66.9	58.8	**53.3**	46.3
CoLoR-Filter	**38.6**	68.7	**25.3**	**51.8**	**32.0**	**72.8**	54.3	49.4	**49.1**
Conditional Only	33.0	65.6	23.0	42.2	27.2	64.6	61.4	51.1	46.0
RHO-down	35.5	67.3	**25.3**	46.9	29.2	67.5	48.6	48.7	46.1
RHO-down + prior	35.6	66.6	**25.3**	49.3	29.4	69.0	**61.6**	50.9	48.5
DSIR	37.6	**68.8**	24.4	46.6	27.8	68.4	59.9	52.6	48.3
Random 8x	38.2	67.8	23.5	44.2	28.8	65.3	58.1	50.5	47.1

Table 4: Final performance for all tasks for 1.2b models. Note that the CoLoR-Filter models do not train on as many tokens since we exhaust all of the tokens in C4 with these settings of τ.

Method	hella-swag	piqa	arc-c	arc-e	open-book qa	sciq	boolq	wino-grande	Avg
Random (25b tokens)	52.9	73.0	26.1	53.7	32.8	75.5	56.7	54.3	53.1
CoLoR-Filter ($\tau = 7$, 25b tokens)	**62.3**	**75.6**	29.7	60.3	**38.0**	79.7	48.3	**58.0**	56.5
CoLoR-Filter ($\tau = 16$, 10b tokens)	59.3	75.4	**31.7**	**62.7**	36.2	**81.0**	57.7	56.4	**57.6**
CoLoR-Filter ($\tau = 32$, 5b tokens)	54.8	74.3	29.4	60.9	35.4	78.4	59.1	54.1	55.8
CoLoR-Filter ($\tau = 64$, 2.5b tokens)	49.3	73.2	28.9	59.7	35.6	77.1	**59.8**	53.0	54.6

C Data diversity and online vs. offline selection

Much work on active learning focuses on ensuring that we select a diverse set of data points that cover the test distribution of interest. As explained in the main text, by making a point estimate of the parameters, CoLoR-Filter is simplifying the problem and sacrificing an explicit term for diversity in the objective. In practice, this seems to be saved by the facts that (1) C4 has already been deduplicated, (2) we still select a fairly large subset without replacement, and (3) an individual sequence contains diversity across tokens.

However, the fact that CoLoR-Filter sacrifices a notion of diversity in the objective is important to consider more deeply. Here, we derive what a loss-based algorithm for data selection that prioritizes diversity would look like and why it is computationally infeasible. Then we derive an approximation (that looks somewhat like RHOLoss [Mindermann et al., 2022]) and show how it is empirically unstable, as was also observed previously by [Mindermann et al., 2022].

To derive a CoLoR-Filter-like algorithm that values diversity, we can start from Equation (3) by a greedy approximation where samples are added to S one (batch) at a time, like in RHO:

$$\approx \min_{x_1,\ldots,x_n \subset D_{\text{train}}} \sum_{i=1}^{n} -\log \int_\theta \Pr(x_i|\theta) \Pr(\theta|D_{\text{down}}, x_{<i}) + \log \int_\theta \Pr(x_i|\theta) \Pr(\theta|x_{<i}) \tag{11}$$

Note that this sort of greedy algorithm for subset selection has a long history in active learning [Das and Kempe, 2018], is actually theoretically sound in some cases [Nemhauser et al., 1978], and is used in prior work [Ash et al., 2021, Mindermann et al., 2022]. Importantly, this algorithm still prioritizes selecting a diverse dataset. By conditioning on past data at step i, the objective encourages the algorithm to select data that is different from data that has already been selected.

We can also make an empirical bayes version by adding D_{prior}:

$$\min_{x_1,\ldots,x_n \subset D_{\text{train}}} \sum_{i=1}^{n} -\log \int_\theta \Pr(x_i|\theta) \Pr(\theta|D_{\text{prior}}, D_{\text{down}}, x_{<i}) \tag{12}$$

$$+ \log \int_\theta \Pr(x_i|\theta) \Pr(\theta|D_{\text{prior}}, x_{<i}) \tag{13}$$

This is, of course, still intractable since it requires integrating the parameters. But, since we have already introduced the greedy algorithm that encourages diversity, if we now make the point estimate

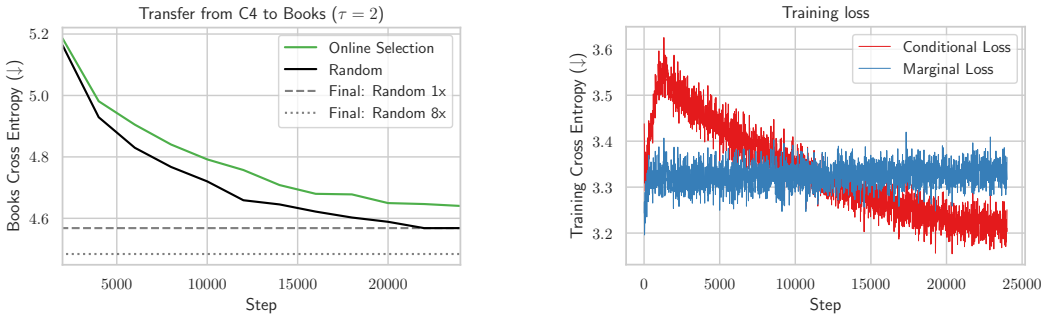

Figure 9: (Left) Performance of online selection with fine-tuning as outlined in Equation (14). Online selection is worse than random. (Right) Training curves for the conditional and marginal models on the selected data S. The conditional model faces training instability early on (associated with forgetting), and then eventually becomes better than the marginal on the selected data.

approximation, the incentive for data diversity remains. This results in:

$$\approx \min_{x_1,\ldots,x_n \subset D_{\text{train}}} \sum_{i=1}^{n} -\log \Pr(x_i|\theta_{\text{prior+down}+x_{<i}}) + \log \Pr(x_i|\theta_{\text{prior}+x_{<i}}) \tag{14}$$

The thorny issue here is how to define $\theta_{\text{prior+down}+x_{<i}}$ and $\theta_{\text{prior}+x_{<i}}$ in practice. In theory, these parameters should be trained on an iid sample from the union of the datasets. If we add the datapoints one at a time, the dynamics of the distribution shift over time can change how well the model corresponds to conditioning on the union of the dataset. But, this would require re-training the models every time we add a new x_i which is clearly impractical.

In practice, this encourages using a fine-tuning approach (as in RHO) where we continually fine-tune on the x_i as they are added. But when D_{down} is small and the data distribution changes over time, we can get catastrophic forgetting and unstable training dynamics. For these reasons, RHO avoids training the conditional model entirely (Appendix D of Mindermann et al. [2022]). We also conduct an experiment on the Books task where we use this online fine-tuning algorithm that updates both the marginal and conditional models as we add data to S. Results in Figure 9 show how the training is unstable and in fact performs worse than random.

Moreover, Note that the computational cost of even the cheapest fine-tuning algorithm is substantial compared to the algorithms in the paper. In particular, the serial cost is now $2\tau n + 4n$ (as compared to $\tau n + 2n$ for RHO) since we need to pass the full τn samples through both the conditional and marginal models. So this variant is clearly inferior in practice to the other approaches we consider.

D Compute cost for scale generalization

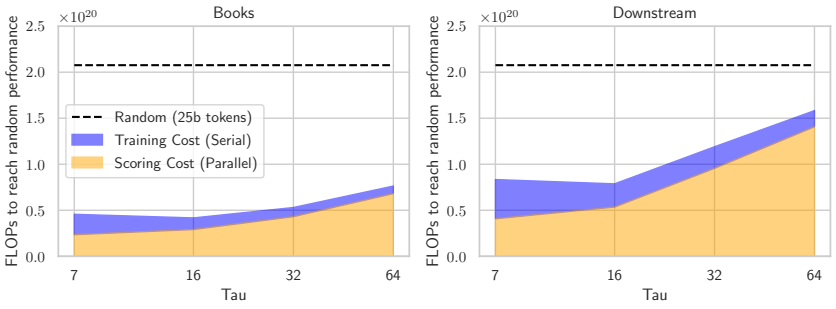

Figure 10: Costs in FLOPs to reach equivalent performance to the final random model trained on 25b tokens (i.e. cost until we reach the dotted line in Figure 1). We split cost into the scoring cost for filtering the data using the small auxiliary models and then training cost for the large model.

In the main text we computed the cost for $\tau = 16$ in terms of model forwards of 1 billion tokens. Here we can convert this to FLOPs and compute the cost for all values of τ. Results are in Figure 10 showing the breakdown of costs into scoring FLOPs for running the small auxiliary models over the data and training FLOPs for training the large model. We measure the cost it takes to reach the final performance of the random model, i.e. until the CoLoR-filter learning curve crosses the dotted line in Figure 1. The main tradeoff is that lower τ values require more scoring cost and less training cost because they are able to select better data.

We should also note that if multiple models are being trained with the same dataset, then this scoring cost can be amortized over those runs and the larger τ values will look even better.

E Can we do data selection in distribution?

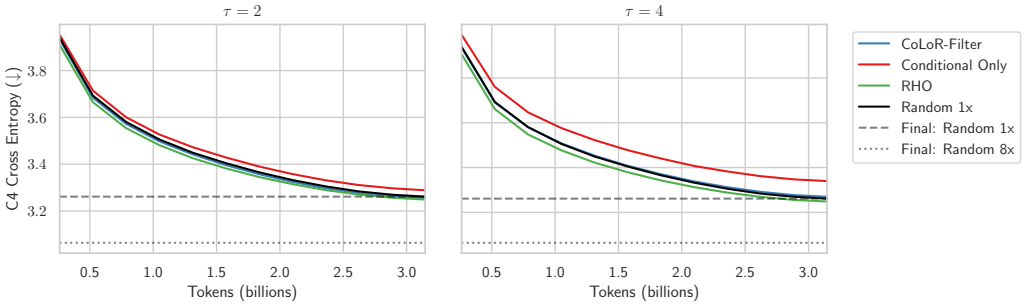

Figure 11: Using a sample of C4 as D_{down}. RHO provides marginal gains here, while CoLoR-Filter does not provide gains at all. Conditional Only is worse than random. Scaling τ does not change results as much as when we target downstream tasks.

One obvious question raised by these data selection techniques is whether they can work in distribution, i.e. can we select data to make the iid loss on C4 go down faster? In Figure 11 we present results for running this experiment with CoLoR-Filter as well as RHO and Conditional Only. Note that there is no difference between RHO and RHO + prior now (and we drop the "down" from the name) since the prior distribution and the downstream distribution are the same. To implement CoLoR-Filter in this setting, we just take two checkpoints from pre-training the prior model and call the earlier one (at 2.5b tokens) the marginal model and the later one (at 3.1b tokens) the conditional model.

We find that in distribution selection does not work effectively with these methods. There are small gains to RHO loss, but here they are massively outweighed by the computational cost of the selection. CoLoR-Filter sees no gain at all over random and Conditional Only is worse than random. These preliminary results suggest why it is important to recognize that data selection (especially with these methods) will be most effective when we genuinely want to target a different distribution from D_{train}.

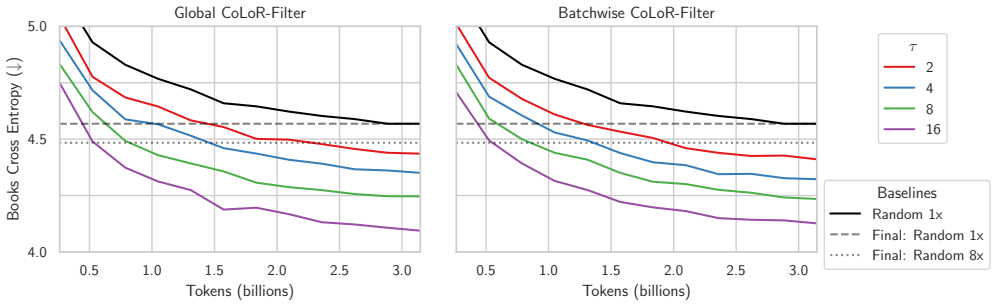

Figure 12: Comparison between global and batchwise variants of CoLoR-Filter on Books. The two perform nearly identically here.

F Global vs. batchwise selection

One more minor implementation aspect about CoLoR-Filter is that as presented in Algorithm 1, we do global selection where we take the best n data points across the entire train set, while in RHO-down in Algorithm 2 selection is done batchwise. Here we ablate whether the ability to do global selection is actually helpful for CoLoR-Filter. Results in Figure 12 suggest that there is not much difference between the two and at small τ, batchwise selection maybe even beat global selection. We provide this result to illustrate that CoLoR-Filter is fairly robust to how the selection is performed.

G Finetuning after targeted pre-training

One possible question about the targeted pre-training setting we consider is: what happens if we finetune on D_{down} after the targeted pre-training?

This is interesting since while the pre-trained models presented in the main text never have direct access to D_{down}, the selection algorithm does. In this section, we also allow access to D_{down} after pre-training and then compare the final performance of the finetuned models that are pre-trained on random data vs. selected data.

First, in Table 5 and Table 6 we present finetuning results for the 150m models. We find that CoLoR-Filter data outperforms 8x as much random data after finetuning. Note that the conditional model that we use to guide the selection of CoLoR-Filter is equivalent to a model that has been pre-trained on 3B random tokens and then finetuned on the task. Thus, these results show that we are substantially outperforming the conditional model when both models are finetuned on the downstream data.

Table 5: Performance after finetuning on Books for different pre-trained 150m models. Note that the Random (3.1b tokens) model is equivalent to the conditional model used to select data with CoLoR-Filter ($\tau = 16$).

Pre-training data	Finetuned Books Val Cross Entropy
Random (3.1b tokens)	3.441
Random (25b tokens)	3.357
CoLoR-Filter (3.1b tokens)	**3.258**

Table 6: Held out performance after finetuning on downstream data for different pre-trained 150m models. Note that the Random (3.1b tokens) model is equivalent to the conditional model used to select data with CoLoR-Filter ($\tau = 16$).

Pre-training data	hella-swag	piqa	arc-c	arc-e	open-book qa	sciq	boolq	wino-grande	Avg
Random (3.1b tokens)	34.4	66.6	24.8	51.7	28.0	89.9	**65.6**	**53.1**	51.8
Random (25b tokens)	**39.5**	69.8	**29.2**	53.9	30.2	**91.4**	64.2	52.9	53.9
CoLoR-Filter (3.1b tokens)	39.2	**71.1**	29.1	**55.3**	**33.2**	90.0	65.1	51.6	**54.3**

Table 7: Performance after finetuning on Books for different pre-trained 1.2b models. Note that the conditional model that selects data is only 150m parameters.

Pre-training data	Finetuned Books Val Cross Entropy
Random (25b tokens)	3.074
CoLoR-Filter (2.6b tokens)	**2.964**

Next, we present results for the 1.2b models in Table 7 and Table 8. We find that the CoLoR-Filter model outperforms or is competitive with training on about 10x as much data randomly selected data. We should also note that the CoLoR-Filter models are now dramatically outperforming the 150m conditional models that were used to filter the data, showing positive scale transfer of data selection.

Table 8: Held out performance after finetuning on downstream data for different pre-trained 1.2b models.

Pre-training data	hella-swag	piqa	arc-c	arc-e	open-book qa	sciq	boolq	wino-grande	Avg
Random (25b tokens)	**55.3**	74.6	35.2	63.0	35.8	**94.6**	**72.0**	**62.5**	**61.6**
CoLoR-Filter (2.6b tokens)	53.4	**76.1**	**35.8**	**65.6**	**36.8**	93.2	66.6	58.9	60.8

H Hyperparameters

Table 9: 150m model parameters, based on Wortsman et al. [2024], Groeneveld et al. [2024]

Parameter	Value
Residual dimension	1024
Depth	12
MLP hidden dimension	4096
Activation	GeLU
Head dimension	64
Context length	512
Positional encoding	RoPE
Biases	False
Normalization	PyTorch Layernorm
QK normalization	True
Precision	Mixed, bfloat16
Tokenizer	GPTNeox

Table 10: 1.2b model, based on Wortsman et al. [2024], Groeneveld et al. [2024]. Only reporting differences from 150m.

Parameter	Value
Residual dimension	2048
Depth	24
MLP hidden dimension	8192

Table 11: Training parameters, based on Wortsman et al. [2024], Groeneveld et al. [2024]

Parameter	Value
Optimizer	Adam
Batch size	256
Learning rate	1e-3
Schedule	Linear warmup, cosine decay
Warmup steps	5% of total steps
z-loss coefficient	1e-4
Weight decay	0.0
β_1	0.9
β_2	0.95
ϵ	1e-15

I Inspecting the selected data

In this section, we conduct some basic analysis of the data that is selected by CoLoR-Filter. We leave a full analysis to future work, but here we provide some high level statistics about the distributions of the scores of the conditional vs. marginal models and some representative examples from the datasets.

I.1 Distribution of scores

First, we simply plot the CDFs of the conditional loss reduction (CoLoR) score function used to select the data. We find that there are relatively few outliers and the CoLoR scores are fairly concentrated and normally distributed. Moreover, we note that the mean CoLoR in both experiments is positive, meaning that the conditional model actually has higher losses on the datapoints in C4 than the marginal model. This makes sense because the conditional model has been finetuned on D_{down} which is out of distribution relative to C4.

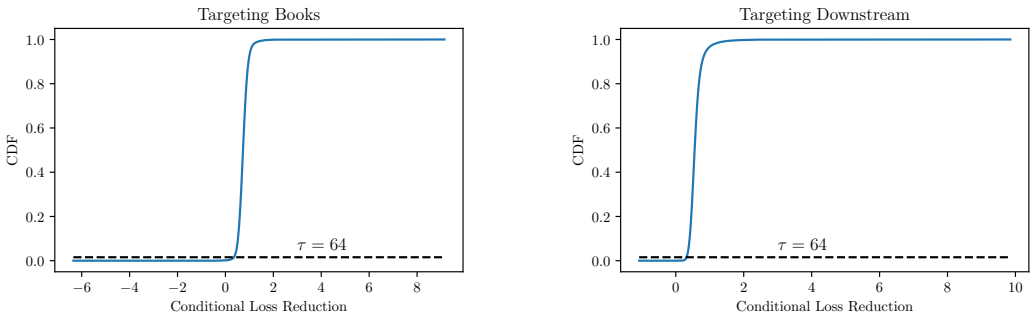

Figure 13: CDFs for the conditional loss reduction (CoLoR), i.e. $-\log \Pr(x|\theta_{\text{prior+down}}) - (-\log \Pr(x|\theta_{\text{prior}}))$. The dashed line highlights the cutoff point for $\tau = 64$. We select the points with the lowest CoLoR.

I.2 Representative examples

Now we just list a few representative examples to give a flavor for the types of outliers that exist under our ranking of sequences and the sorts of typical sequences that are selected versus excluded. The sequences are sampled randomly from different quantiles of the distribution and we shorten all the sequences so that they fit more easily on the page.

Figure 14 shows outliers when targeting Books and Figure 15 shows more typical examples when targeting Books. Generally, we found that the documents with very high scores contain things like old English, poetry, and tables of contents that are particularly unusual in books compared to the rest of the internet. Other things like fiction and dialogue are also highly scored. Negative outliers typically have things like poorly encoded text or advertisements.

Figure 16 shows outliers when targeting downstream tasks and Figure 17 shows more typical examples when targeting downstream tasks. Here the patterns are less clear since the target tasks are more diverse, but we did observe many scientific and wiki-style documents with high scores as well as some descriptions of physical interactions that may be useful for common sense tasks. Again, the negative outliers tend to have things like poorly encoded text or advertisements.

```
AS now shall ye wyt, what tyme of
the day ye shall angle.  From the
begynning of Maye vntill it be
September:  the byting tyme is early
in the morow from four of the clocke
vnto eyght of the clocke, at after
none from foure to eyght also, but
not so good as in the mornyng, and if
it be a colde wynde and a lowryng day,
it is muche better than a cleere daye.
Also many poole fysshes will byte
best in the morne tyde.  And if ye se
in any tyme of the day the Troute or
greylyng lepe angle to him with a dub
according to the same moneth.  And
where the water ebbeth and floweth:
the fish wyll byte in some place at
the ebbe and in some place at the
flud after they haue restyng
```

(a) Good outlier, CoLoR = -0.35

```
???????????????????????????????
????????????????????????????????
?????????????????????????????????
??????????????????????????????????
??????????????????????????????????
????????????????????????????????
???????????????????????????????????
????????????????????????????????????
?????????????????????????????????????
?????????????????????????????????????
?????????????????????????????????????
?????????????????????????????????????
????????????
?????????????????????????  m88
????????????????????????  ????  m88
??????????????????????????????????????
???????????????????????????  ????
??????????????????????????????????
???????????????????????????????  ?
???????????????????
```

(b) Bad outlier, CoLoR = 5.45

Figure 14: Examples of outliers when targeting **Books**. Examples are sampled randomly from the top or bottom 1000 sequences. The positive outlier is written in an older dialect of English which may be related to some documents in the Project Gutenberg corpus, while the negative outlier appears to be poorly encoded.

C: Mrs Mackenzie, was there ever a time when you felt like you could just hop on a plane and make that flight down to the next State to be with your boys? B: Oh my dear, yes. I feel sometimes as if I'm twenty and so fit and active and I can do whatever I want to do and then I remember, good grief, I'm 86, you old fool, you can't do that. I wish I could just fly down there and live with them all together just how it was when they were little and I was their Mum and they followed me because I was so bright and cheery and smart and active and all the things that I'm not now. Oh, I'm so sorry, listen to me. Maybe I'm just losing my marbles, what do you think, dear? C: Smiling - Imagine if I waved a magic wand and miraculously you were twenty again. What would you see yourself doing Beryl. Is it ok if I call you Beryl?

(a) Sequence from best 3%, CoLoR = 0.40

Chamber of Commerce and other business venues, such as the Gwinnett Civic & Convention Centers and is an ideal working environment for commercial businesses and corporations in Northeast Atlanta. The prominent location is on a heavily wooded, landscaped 6.5 acre site fronting on I-85. The exterior features green-tinted thermal glass and the entrance features a curtain wall glass leading into a granite-floored lobby with vaulted ceilings. Gwinnett County is home to leading Fortune 500 companies, drawn by its reputation as a commerce and technology hub, providing businesses with a regional market of five million people. SERVPRO of Gurnee can simplify the restoration process by handling both the initial water damage mitigation and rebuilding the affected areas. Having one qualified company for the entire process can save time and keep costs low.

(b) Sequence from median 3%, CoLoR = 0.73

Figure 15: Examples of more typical documents when targeting **Books**. First a document from the top 3% that would be selected with $\tau = 32$, and then a document that scores near the median of all documents. The selected document is fictional dialogue while the median document is an advertisement.

among the pinacoderm are the ostia that allow entry of water into the body of the sponge. These pores have given the sponges their phylum name Porifera-pore-bearers. In some sponges, ostia are formed by porocytes, single tube-shaped cells that act as valves to regulate the flow of water into the spongocoel. In other sponges, ostia are formed by folds in the body wall of the sponge. Between the outer layer and the feeding chambers of the sponge is a jelly-like substance called the mesohyl, which contains collagenous fibers. Various cell types reside within the mesohyl, including amoebocytes, the ‘‘stem cells’’ of sponges, and sclerocytes, which produce skeletal materials. The gel-like consistency of mesohyl acts like an endoskeleton and maintains the tubular morphology of sponges. The feeding chambers inside the sponge are lined by choanocytes (‘‘collar cells’’).

(a) Good outlier, CoLoR = -0.46

*** **********. ****** *** ***, *** ******* **** **** ** ******** ******* plates ** ****** ** ** **-** *** (*** ******* ** tested), ***** ******* ********. *** ** *** ***, *** ********* *.* ********* ******* ***** capture ****** ******** ******** ****** ** **** ** **** ******, >10 ***, *** ******, **+ ***, **** ** ****** ***** or ****, *** ** ***** ****** *** *** **** **** field ** ****, ***** **’. ***** ******* ********, *** ******** ** ***** ****** ****** ****** to ******* ****** ** ****** **** ** **** ****** ** night, ****** ******* ******* *** ********. ******** ******** ** */****, *** ******** ****** ******** ******** **** front *** **** ****** ****** ** *** *** **** ******. However, **** ******* ******* *** ********** ** *** ***** ** night, ****** ** **** ****** *** ******* ************ ** *** scene.

(b) Bad outlier, CoLoR = 5.36

Figure 16: Examples of outliers when targeting **downstream** tasks. Examples are sampled randomly from the top or bottom 1000 sequences. The positive outlier is a scientific document that could be relevant for tasks like SciQ, while the negative outlier appears to be poorly encoded.

summer plans. After thinking for a while I decided to spend my summer in Squamish, where I would work for the Admissions Team. However, due to a very large number of students interested to work on campus and a limited number of work positions, I ended up not getting a job on campus. I was very upset indeed and I began to think that there were not any job openings elsewhere, which would then result in me travelling back home. Surprisingly, there were many job opportunities in the Squamish community. Since Quest University Canada hosted a job fair on campus I, along with all the students, had the chance to meet local businesses that were looking for summer employees. It was a great opportunity to network and give my resume to the ones that interested me.

(a) Sequence from best 3%, CoLoR = 0.33

Can I install PDF Stacks on more than one computer? The license key is valid for only one device and is non-transferable. You can obtain additional license key(s) by placing an order. How do I use PDF Stacks? Click "File" and then "Import Folder" Once you import the PDF files, your files will be copied into PDF Stacks for easier ability to read, search, organize, take notes, print and share. Any questions, ask us! How do I create collections (virtual binders) and match/tag my documents for better organization? It’s easy. Watch the video for creating collections and tagging documents. Can multiple users access the same documents or can I access and sync my documents through multiple devices?

(b) Sequence from median 3%, CoLoR = 0.55

Figure 17: Examples of more typical documents when targeting **downstream** tasks. First a document from the top 3% that would be selected with $\tau = 32$, and then a document that scores near the median of all documents. The selected document appears to be a journal entry while the median document is software documentation

J Comparison to Fineweb-Edu

Concurrent to our initial work, Penedo et al. [2024] released FineWeb-edu, a classifier for educational content that can filter the FineWeb dataset. Here we provide a comparison between CoLoR-Filter and this classifier-based approach.

Specifically, we re-implement the CoLoR-Filter pipeline on top of the Fineweb dataset and with slightly smaller auxiliary models (125m) to make a more fair comparison to FineWeb-edu. Then we compare on the same suite of 8 downstream tasks over various settings of τ using the two scores: CoLoR-Filter or the FineWeb-edu classifier. We then train larger models (680M parameters) for 10B tokens of selected data. Results are shown in fig. 18. We find that CoLoR-Filter consistently outperforms FineWeb-edu, which is not so surprising since we are doing more targeted data selection by specifically targeting the downstream NLP tasks rather than a general notion of "educational content".

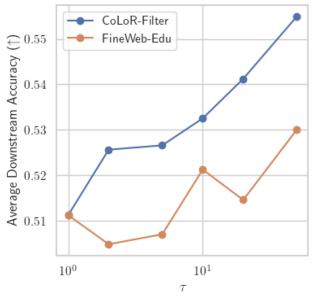

Figure 18: A comparison of the performance of 680m models trained on 10B tokens selected with various τ between CoLoR-Filter and FineWeb-edu.

K Broader Impact

The development of the CoLoR-Filter for data selection has notable broader impacts on both machine learning and society. It enhances efficiency in language model training, leading to reduced computational resources and environmental footprint, while its scalability democratizes access to high-performing models. The method's success in diverse downstream tasks promises advancements in fields like medical text processing and legal analysis. However, it also raises concerns about dataset bias, necessitating continuous evaluation and updates. Future research should focus on ensuring models do not inherit biases from the selected training data, extending applications, improving efficiency, and implementing safeguards to maximize societal benefits while minimizing risks.

L Compute resources

All training is conducted on an internal cluster using H100 GPUs. On one GPU, each 150m training run for 3.1b tokens takes about 4 hours, running the auxiliary models offline and in parallel can be faster. Training the 1.2b model to completion takes about 2 days on 4 GPUs.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and intro clearly state the key results directly.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss limitations with respect to approximations, data diversity, and computational cost.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We do not have formal theorems, but each step of our derivations is fully stated and assumptions are clear.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All hyperparameters are listed and algorithms clearly explained.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Code is included in supplementary material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Appendix H, Section 5, Section 6

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Since we do computationally intensive language modeling experiments and results are clear cut, we do not include error bars.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Appendix L.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We conform to the code of ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See K.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [NA] .

    Justification: We only train small models on well established benchmark data.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the code and data we use for the project (all very standard tools in open source ML).

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA] .

    Justification: No new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA] .

    Justification: No human subjects

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA] .

    Justification: No human subjects

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.