Data-Efficient Training by Evolved Sampling

Ziheng Cheng
UC Berkeley
ziheng_cheng@berkeley.edu

Zhong Li & Jiang Bian
Microsoft Research Asia
{lzhong,jiang.bian}@microsoft.com

Abstract

Data selection is designed to accelerate learning with preserved performance. To achieve this, a fundamental thought is to identify informative data samples with significant contributions to the training. In this work, we propose **Evolved Sampling** (**ES**), a simple yet effective framework for *dynamic* sampling performed along the training process. This method conducts *batch* level data selection based on *differences* of historical and current losses, significantly reducing the back propagation time while maintaining the model performance. ES is also readily extensible to incorporate *set* level data selection for further training accelerations. As a plug-and-play framework, ES consistently achieves lossless training accelerations across various models, datasets, and optimizers, saving up to 40% wall-clock time. Particularly, the improvement is more significant under the *noisy supervision* setting. When there are severe corruptions in labels, ES can obtain accuracy improvements of approximately 20% relative to the standard batched sampling.

1 Introduction

Deep learning has showcased remarkable performance across a variety of real-world applications, particularly leading to unparalleled successes of large "foundation" models [Touvron et al., 2023, Rombach et al., 2022]. On the other hand, since these large models are usually trained on web-scale datasets, the overall computation and memory loads are considerably increasing, calling for more *efficient* developments of modern machine learning. Efficient learning involves several aspects, centering around models, data, optimization, systems, and so on [Shen et al., 2023].

For data-efficient machine learning, the core is to properly evaluate the importance per data sample in the original (large-scale) datasets. A broad array of methods is applied in a *static* manner, where the samples' importance is determined before the training. However, these approaches can be prohibitively expensive to apply in practice, since their dependence on feature representations requires additional (pre-)training in advance.

Another array of methods lies in a *dynamic* sense, where the samples' importance is simultaneously evaluated along the training process. Dynamic sampling methods can be further divided into two categories: *set* level selection, to prune the whole dataset at the beginning of each epoch [Qin et al., 2024, Raju et al., 2021, Thao Nguyen et al., 2023, Attendu and Corbeil, 2023], and *batch* level selection, to sample subsets from original batches for back propagation [Kawaguchi and Lu, 2020, Katharopoulos and Fleuret, 2017, 2018]. Nevertheless, these dynamic sampling methods leverage similar strategies to evaluate the samples' importance. Based on the naive intuition that samples' contributions to the learning are directly associated with gradient updates, most previous methods re-weight data samples with scales of gradients [Mirzasoleiman et al., 2020, Killamsetty et al., 2021], current losses [Jiang et al., 2019, Loshchilov and Hutter, 2016, Schaul et al., 2016], or adopt reference models [Mindermann et al., 2022, Deng et al., 2023, Xie et al., 2023]. However, these approaches suffers from significant computation loads and exploit the historical information inadequately.

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

Table 1: The comparison of different dynamic sampling methods. The "history" column denotes whether the method uses historical information along the training. The "robust" column represents the performance robustness under (severe) label noises. The last column summarizes the ratio of samples used for back propagations (BPs) relative to the standard training. Here, r stands for the pruning ratio for *set* level methods (pruning data samples of the whole epoch), and b/B represents the pruning ratio for *batch* level methods (selecting a mini-batch $\mathfrak{b}$ (subset) from a meta-batch $\mathcal{B}$).

	set	*batch*	history	robust	# of samples for BP
UCB [Raju et al., 2021]	✓		✓		$1-r$
KA [Thao Nguyen et al., 2023]	✓				$1-r$
InfoBatch [Qin et al., 2024]	✓		✓		$1-r$
Loss [Katharopoulos and Fleuret, 2017]		✓			b/B
Order [Kawaguchi and Lu, 2020]		✓			b/B
ES (ours)		✓	✓	✓	b/B
ESWP (ours)	✓	✓	✓	✓	$(1-r)b/B$

To tackle these challeges, we propose a novel dynamic sampling framework, **Evolved Sampling** (**ES**), which incorporates the loss *evolution* or *differences* along the training process to determine samples' importance and conduct *batch* level selection, without the demand of pre-trained reference models. Due to its simplicity, this procedure is effortless to implement and only introduces mild computational overheads with negligible memory costs, while significantly reducing the number of samples used for back propagations (BPs) and consequently saving the overall wall-clock time, without degrading the model performance. Moreover, ES facilitates convenient extensions to data pruning on the *set* level, i.e. **Evolved Sampling with Pruning** (**ESWP**), leading to further accelerations with lossless model performance. We demonstrate the differences in details between our proposed methods and previous dynamic sampling methods in Table 1.

2 Methods

2.1 Preliminaries

The classic setting of general machine learning tasks is as follows. Given a dataset $\mathcal{D}$ of size n, the goal is to solve the empirical risk minimization (ERM) problem:

$$\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \hat{L}_n(\boldsymbol{\theta}) := \frac{1}{n}\sum_{i=1}^{n} \ell_i(\boldsymbol{\theta}),. \tag{2.1}$$

Here, $\ell_i(\cdot)$ denotes the *non-negative* loss function of the i-th sample, and $\hat{L}_n(\boldsymbol{\theta})$ represents the empirical averaged loss over n data samples. When n is large, a common routine is to compute stochastic gradient on a random batch instead of the whole training set. For instance, starting from an initialization $\boldsymbol{\theta}(0)=\boldsymbol{\theta}_0$, the SGD optimizer updates model by

$$\boldsymbol{\theta}(t+1) = \boldsymbol{\theta}(t) - \frac{\eta_t}{B}\sum_{j=1}^{B} \nabla_{\boldsymbol{\theta}} \ell_{i_j}(\boldsymbol{\theta}(t)) \approx \boldsymbol{\theta}(t) - \eta_t \nabla_{\boldsymbol{\theta}} \hat{L}_n(\boldsymbol{\theta}(t)), \tag{2.2}$$

where $B \le n$ is the batch size. The standard method is to draw the batch uniformly without replacement for $\lceil n/B \rceil$ iterations in one epoch, which we refer as the standard batched sampling.

2.2 Evolved Sampling

For the loss-weighted sampling, one calculates the sampling probability as

$$p_i(t) \propto w_i(t) = \ell_i(\boldsymbol{\theta}(t)), \tag{2.3}$$

In general machine learning tasks, the typical behaviors of loss curves often appear decent trends overall, but can oscillate meanwhile due to certain noises. This introduces the sensitivity or instability issue of the sampling scheme (2.3). A natural smoothing operation is to use the EMA of losses

$$p_i(t) \propto w_i(t) = \beta w_i(t-1) + (1-\beta)\ell_i(\boldsymbol{\theta}(t)), \quad w_i(0) = 1/n \tag{2.4}$$

where the hyper-parameter $\beta \in [0, 1]$ is typically selected close to 1 to capture more historical information. However, the EMA can potentially erase too many dynamical details (including noises) shown in the loss dynamics. To see this, we give an illustration in Figure 1. The black curve denotes a (polynomially) decayed function with random perturbations, which is designed to mimic typical behaviors of loss curves in general machine learning tasks and fails to provide information robustly due the noises. On the other hand, the blue curve represents the EMA, which leads to over-smoothing due to the average effect.

Decoupled EMA. To sufficiently leverage the loss dynamics in a more robust sense, we propose to calculate the sampling probability as

$$\begin{aligned} p_i(t) \propto w_i(t) &= \beta_1 s_i(t-1) + (1-\beta_1)\ell_i(\boldsymbol{\theta}(t)), \\ s_i(t) &= \beta_2 s_i(t-1) + (1-\beta_2)\ell_i(\boldsymbol{\theta}(t)), \quad s_i(0) = 1/n \end{aligned} \tag{2.5}$$

with $\beta_1, \beta_2 \in [0, 1]$ as two hyper-parameters. Here, the intermediate series $\{s_i(t)\}_{t\in\mathbb{N}}$, updated in the EMA scheme, is also referred as the score (for the i-th sample). The scheme (2.5) is the so-called *decoupled EMA*, which reduces to (2.4) when $\beta_1 = \beta_2 = \beta$. In Figure 1, it is shown by the red curve and appears an "interpolation" between the original loss and single EMA: When losses oscillate, the decoupled EMA reacts moderately by not only capturing detailed dynamics of losses, but also remaining necessary robustness , exhibiting the flexibility to trade-off (by tuning two betas).

Intuitively, by setting $(\beta_1, \beta_2) \to (0^+, 1^-)$, we are able to exploit the long-term historical information along the training (via β_2), while focusing on the importance of current losses (via β_1) and thus can get the best of both world. This simple and elegant design turns out to be surprisingly beneficial in practice, which is further verified in numerous experiments in Section 3.

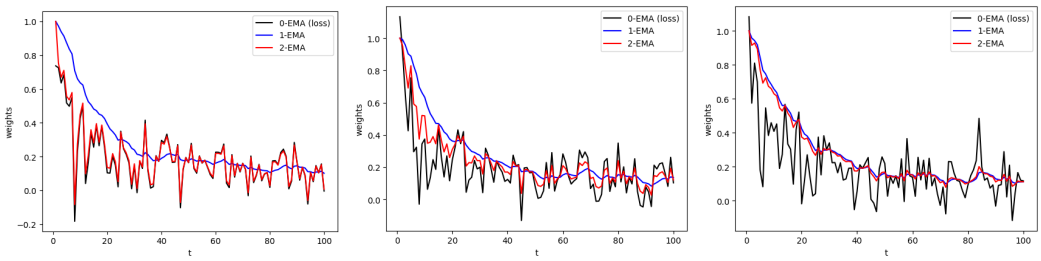

Figure 1: The effect of EMAs, where the output weight is a function of the time step t. From left to right: $\beta_1 = 0.1, 0.5, 0.8$, and $\beta = \beta_2 \equiv 0.9$.

Annealing. Notably, similar to other loss-weighted sampling methods, the decoupled EMA sampling scheme (2.5) also assigns different weights on the respective gradient of data samples, leading to a biased estimation on the true gradient $\nabla_{\boldsymbol{\theta}} \hat{L}_n(\cdot)$ (that assigns uniform weights). Inspired by Qin et al. [2024], we adopt the *annealing* strategy, to perform normal training (with the standard batched sampling, no data selection) at the last few epochs. Besides, to get a better initialization of the score $\{s_i(\cdot)\}_{i\in[n]}$, we also apply the annealing strategy at the first few epochs.

Combining the decoupled EMA sampling scheme (2.5) with the annealing strategy, we obtain the **Evolved Sampling** (**ES**) framework (formalized in Algorithm 1).

Pruning. Note that applying the decoupled EMA sampling scheme (2.5) to meta-batches (with the batch size B) has already introduced data selection in a *batch* level, since one can always select a smaller batch (with the batch size $b < B$) out of the meta-batch, according to the sampling probability $p_i(t)$ defined in (2.5). For more aggressive data pruning and enhanced data efficiency, we can further extend ES by involving the *set* level data selection (i.e. randomly pruning the whole dataset according to the probability proportional to the score $\{s_i(e)\}_{i=1}^n$ at the beginning of the e-th epoch), which is **Evolved Sampling with Pruning** (**ESWP**; formalized in Algorithm 1).

3 Experiments

In this section, we provide numerical simulations on the proposed method (ES(WP); Algorithm 1) to demonstrate its effectiveness, efficiency, robustness and flexibility. For all sampling methods, the hyper-parameters used in data augmentation are maintained the same.

Algorithm 1 Learning by Evolved Sampling (with Pruning)

Require: Dataset $\mathcal{D} = \{\boldsymbol{z}_i\}_{i=1}^n$, model space $\boldsymbol{\Theta} \ni \boldsymbol{\theta}$, optimizer (e.g. SGD, Adam)
Require: Pruning ratio r, meta-batch size B, mini-batch size $b \leq B$, decoupled EMAs' hyper-parameters $\beta_1, \beta_2 \in (0, 1)$, total number of epochs E, number of annealing epochs E_a
Initialize the model $\boldsymbol{\theta}(0) = \boldsymbol{\theta}_0$, the score $\boldsymbol{s}(0) = \frac{1}{|\mathcal{D}|}\mathbf{1}_n = \frac{1}{n}\mathbf{1}_n$, $t = 0$
for $e = 0, 1, \cdots, E-1$ **do**
  if $E_a \leq e < E - E_a$ **then**
    Sample a sub-dataset $\mathcal{D}_e$ ($|\mathcal{D}_e| = (1-r)|\mathcal{D}|$) from $\mathcal{D}$ without replacement, according to the probability $p_i'(e) \propto s_i(e)$ (normalized w.r.t. $i \in [n]$) ▷ "pruning"
  else
    Set $\mathcal{D}_e = \mathcal{D}$
  end if
  for $j = 0, 1, \cdots, \lceil \frac{|\mathcal{D}_e|}{B} \rceil - 1$ **do**
    Sample a meta-batch $\mathcal{B}_t$ ($|\mathcal{B}_t| = B$) uniformly from $\mathcal{D}_e$ without replacement
    Compute the loss $\ell_i(\boldsymbol{\theta}(t))$ for $\boldsymbol{z}_i \in \mathcal{B}_t$
    Update the score: $s_i(e+1) \leftarrow \beta_2 s_i(e) + (1-\beta_2)\ell_i(\boldsymbol{\theta}(t))$ for $\boldsymbol{z}_i \in \mathcal{B}_t$
    Update the weight: $w_i(e) \leftarrow \beta_1 s_i(e) + (1-\beta_1)\ell_i(\boldsymbol{\theta}(t))$ for $\boldsymbol{z}_i \in \mathcal{B}_t$
    if $E_a \leq e < E - E_a$ **then**
      Sampling a mini-batch $\mathfrak{b}_t$ ($|\mathfrak{b}_t| = b$) from $\mathcal{B}_t$ without replacement, according to the probability $p_i(e) \propto w_i(e)$ (normalized w.r.t. $\{i \in \mathbb{N}_+ : \boldsymbol{z}_i \in \mathcal{B}_t\}$)
      Update the model: $\boldsymbol{\theta}(t+1) \leftarrow \text{optimizer}(\boldsymbol{\theta}(t); \mathfrak{b}_t)$
    else
      Update the model: $\boldsymbol{\theta}(t+1) \leftarrow \text{optimizer}(\boldsymbol{\theta}(t); \mathcal{B}_t)$ ▷ "annealing"
    end if
    $t \leftarrow t+1$
  end for
end for

3.1 Effectiveness and Efficiency

Configurations. For ES/ESWP, the default hyper-parameters are as follows: The annealing ratio is $E_a/E = 5\%$; the pruning ratio is $r = 20\%$ for ESWP; in decoupled EMAs, $(\beta_1, \beta_2) = (0.2, 0.9)$ for ES, $(\beta_1, \beta_2) = (0.2, 0.8)$ for ESWP; for both ES and ESWP, the ratio of mini-batch size over meta-batch size is $b/B = 25\%$. For the two *batch* level selection methods (Order, Loss), we use the same mini/meta-batch size.

Results. (i) For small-scale tasks, we train ResNet models on CIFAR datasets. It is shown in Table 2 that the batch level selection methods (Loss, Order, ES) typically exhibits limited accelerations on these small-scale tasks, since these methods often require additional forward propagation overheads that are not negligible compared to BPs. Nevertheless, ES is the only algorithm that achieves lossless accelerations across all methods. Notably, ESWP saves the most computation time while maintaining the best performance (also comparable to Baseline) among set level selection methods.

Table 2: The test accuracy (%) and saved time of training ResNet models on CIFAR datasets.

	CIFAR-10 (R-18)		CIFAR-100 (R-18)		CIFAR-100 (R-50)	
Baseline	95.4		78.8		81.1	
UCB [Raju et al., 2021]	$95.2_{\downarrow 0.2}$	18%	$77.6_{\downarrow 1.2}$	18%	$80.5_{\downarrow 0.6}$	24%
KA [Thao Nguyen et al., 2023]	$95.3_{\downarrow 0.1}$	21%	$78.1_{\downarrow 0.7}$	21%	$80.2_{\downarrow 0.9}$	24%
InfoBatch [Qin et al., 2024]	$95.3_{\downarrow 0.1}$	21%	$78.4_{\downarrow 0.4}$	**24%**	$80.4_{\downarrow 0.7}$	28%
Loss [Katharopoulos and Fleuret, 2017]	$95.3_{\downarrow 0.1}$	11%	$78.4_{\downarrow 0.4}$	10%	$80.5_{\downarrow 0.6}$	12%
Order [Kawaguchi and Lu, 2020]	$\mathbf{95.4}_{\uparrow 0.0}$	11%	$78.5_{\downarrow 0.3}$	10%	$80.9_{\downarrow 0.2}$	12%
ES	$\mathbf{95.4}_{\uparrow 0.0}$	10%	$\mathbf{78.8}_{\uparrow 0.0}$	10%	$\mathbf{81.1}_{\uparrow 0.0}$	11%
ESWP	$95.3_{\downarrow 0.1}$	**24%**	$78.6_{\downarrow 0.2}$	**24%**	$80.6_{\downarrow 0.5}$	**31%**

(ii) For large-scale tasks, we fine-tune the ViT-Large model on the ImageNet-1K dataset, and summarize the performance of different sampling methods in Table 3. Under this setting, ES continues to show the best performance among batch level selection methods and the second-to-highest accuracy across all sampling methods. Notably, ESWP achieves the best performance and

most significant time reduction, suggesting that ESWP inherits the advantages of *both* set and batch level selection methods. In addition, it is observed that the training speed-up of batch level methods gets far more significant given these large-scale tasks, conversely surpassing the set level methods compared to (i). This is due to the dominance of the saved computation in BPs. Furthermore, many sampling methods achieve higher accuracies than the baseline, implying huge potentials of data selection in large-scale machine learning.

Table 3: The validation accuracy (%) and saved time of fine-tuning ViT-Large on the ImageNet-1K.

	Baseline	UCB	KA	InfoBatch	Loss	Order	ES	ESWP
Accuracy	84.4	84.2	84.3	84.7	84.3	84.2	84.7	**85.0**
Time↓	-	23.6%	25.3%	23.5%	36.4%	38.2%	26.0%	**40.7%**

3.2 Robustness under Label Noises

In this section, we further demonstrate that ES(WP) exhibits more notable advantages when there are label noises. We train ResNet models on CIFAR datasets under both light (10%) and heavy (40%) label noises, which are injected randomly with uniform probabilities or flipped to another class.

In Table 4, we summarize the results of training the ResNet-18 model on the CIFAR-100 dataset under different levels and types of label noises. It is shown that ES/ESWP consistently outperforms all the other sampling methods (including the baseline) with clear gaps, and the improvement is more significant when the label noises become severer.

Table 4: The test accuracy (%) of training the ResNet-18 on the CIFAR-100 with label noises.

	Baseline	UCB	KA	InfoBatch	Loss	Order	ES	ESWP
Flip (10%)	72.3	68.7$_{\downarrow 3.6}$	67.0$_{\downarrow 5.3}$	71.5$_{\downarrow 0.8}$	72.9$_{\uparrow 0.6}$	70.8$_{\downarrow 1.5}$	**73.1**$_{\uparrow 0.8}$	**73.1**$_{\uparrow 0.8}$
Flip (40%)	46.8	43.9$_{\downarrow 2.9}$	45.0$_{\downarrow 1.8}$	46.6$_{\downarrow 0.2}$	53.6$_{\uparrow 6.8}$	47.8$_{\uparrow 1.0}$	57.1$_{\uparrow 10.3}$	**58.2**$_{\uparrow 11.4}$
Uniform (10%)	68.3	66.6$_{\downarrow 1.7}$	65.4$_{\downarrow 2.9}$	67.8$_{\downarrow 0.5}$	67.0$_{\downarrow 1.3}$	65.4$_{\downarrow 2.9}$	**68.7**$_{\uparrow 0.4}$	**68.7**$_{\uparrow 0.4}$
Uniform (40%)	50.8	44.1$_{\downarrow 6.7}$	44.0$_{\downarrow 6.8}$	50.8$_{\uparrow 0.0}$	57.3$_{\uparrow 6.5}$	37.9$_{\downarrow 12.9}$	**61.1**$_{\uparrow 10.3}$	60.1$_{\uparrow 9.3}$

3.3 Ablation Studies

Decoupled EMA and annealing. We numerically test the effectiveness of two important components applied in ES, i.e. the decoupled EMA and annealing. Here, we perform ablations on combinations of "Loss", "A" (Annealing), "E" (single EMA) and "DE" (decoupled EMA). From Table 5, it is observed that: (i) Annealing is an effective technique to boost performance; (ii) EMA also contributes to the improvements; (iii) Compared to the single EMA, the decoupled EMA provides more substantial benefits to the training process.

Table 5: Ablations on decoupled EMAs and annealing for different models, datasets and noises.

	ResNet-18		ResNet-50		ALBERT-Base
	CIFAR-10 (40%)	CIFAR-100 (10%)	CIFAR-100	CIFAR-100 (40%)	CoLA
Loss	83.3	67.0	80.5	53.8	55.1
Loss + A	84.4	68.4	80.8	60.1	55.8
Loss + E	83.4	66.2	80.5	53.6	57.6
Loss + DE	83.7	66.8	**81.1**	54.2	57.5
Loss + A + E	84.6	68.0	80.4	60.3	57.6
ES = Loss + A + DE	**85.2**	**68.7**	**81.1**	**60.9**	**58.4**

4 Conclusion

In this work, we propose a simple yet effective framework, Evolved Sampling, which can be applied to general machine learning tasks to improve the data efficiency in a dynamic manner. By further adopting differences of historical losses to determine samples' importance for data selection, Evolved Sampling can achieve lossless training with significant accelerations, particularly when there are severe noises in labels. Studies in the future may include: (i) more rigorous mathematical analysis on the effect of data selection (e.g. Kolossov et al. [2024]); (ii) More specific applications, such as data selection/reduction on domain mixtures (e.g. Chen et al. [2023], Xie et al. [2023]); (iii) More efficient and scalable implementation, such as data parallelism [You et al., 2017, 2020].

References

Amro Kamal Mohamed Abbas, Kushal Tirumala, Daniel Simig, Surya Ganguli, and Ari S. Morcos. Semdedup: Data-efficient learning at web-scale through semantic deduplication. In *ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2023. URL https://openreview.net/forum?id=4vlGm9gv6c.

Jean-Michel Attendu and Jean-Philippe Corbeil. NLU on data diets: Dynamic data subset selection for NLP classification tasks. In Nafise Sadat Moosavi, Iryna Gurevych, Yufang Hou, Gyuwan Kim, Young Jin Kim, Tal Schuster, and Ameeta Agrawal, editors, *Proceedings of the Fourth Workshop on Simple and Efficient Natural Language Processing (SustaiNLP)*, pages 129–146, Toronto, Canada (Hybrid), July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.sustainlp-1.9. URL https://aclanthology.org/2023.sustainlp-1.9.

Olivier Bachem, Mario Lucic, and Andreas Krause. Coresets for nonparametric estimation - the case of DP-Means. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 209–217, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/bachem15.html.

Valeriu Balaban, Jayson Sia, and Paul Bogdan. Robust learning under label noise by optimizing the tails of the loss distribution. In *International Conference on Machine Learning and Applications (ICMLA)*, pages 520–527, 2023. doi: 10.1109/ICMLA58977.2023.00078.

Mayee F. Chen, Nicholas Roberts, Kush Bhatia, Jue Wang, Ce Zhang, Frederic Sala, and Christopher Ré. Skill-it! A data-driven skills framework for understanding and training language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 36000–36040. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/70b8505ac79e3e131756f793cd80eb8d-Paper-Conference.pdf.

Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. Selection via proxy: Efficient data selection for deep learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HJg2b0VYDr.

Sanjoy Dasgupta, Daniel Hsu, Stefanos Poulis, and Xiaojin Zhu. Teaching a black-box learner. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1547–1555. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/dasgupta19a.html.

Zhijie Deng, Peng Cui, and Jun Zhu. Towards accelerated model training via Bayesian data selection. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 8513–8527. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1af3e0bf5905e33789979f666c31192d-Paper-Conference.pdf.

Melanie Ducoffe and Frederic Precioso. Adversarial active learning for deep networks: A margin based approach. *arXiv preprint arXiv:1802.09841*, 2018.

Ayoub El Hanchi, David A. Stephens, and Chris J. Maddison. Stochastic reweighted gradient descent. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 8359–8374. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/hanchi22a.html.

Sariel Har-Peled and Soham Mazumdar. On coresets for k-means and k-median clustering. In *Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '04, page 291–300, New York, NY, USA, 2004. Association for Computing Machinery. ISBN 1581138520. doi: 10.1145/1007352.1007400. URL https://doi.org/10.1145/1007352.1007400.

Lingxiao Huang, Shaofeng H.-C. Jiang, Jianing Lou, and Xuan Wu. Near-optimal coresets for robust clustering. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Nc1ZkRW8Vde.

Angela H. Jiang, Daniel L.-K. Wong, Giulio Zhou, David G. Andersen, Jeffrey Dean, Gregory R. Ganger, Gauri Joshi, Michael Kaminksy, Michael Kozuch, Zachary C. Lipton, and Padmanabhan Pillai. Accelerating deep learning by focusing on the biggest losers. *arXiv preprint arXiv:1910.00762*, 2019.

Angelos Katharopoulos and François Fleuret. Biased importance sampling for deep neural network training. *arXiv preprint arXiv:1706.00043*, 2017.

Angelos Katharopoulos and Francois Fleuret. Not all samples are created equal: Deep learning with importance sampling. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2525–2534. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/katharopoulos18a.html.

Kenji Kawaguchi and Haihao Lu. Ordered SGD: A new stochastic optimization framework for empirical risk minimization. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 669–679. PMLR, 26–28 Aug 2020. URL https://proceedings.mlr.press/v108/kawaguchi20a.html.

Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, Abir De, and Rishabh Iyer. GRAD-MATCH: Gradient matching based data subset selection for efficient deep model training. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5464–5474. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/killamsetty21a.html.

Germain Kolossov, Andrea Montanari, and Pulkit Tandon. Towards a statistical theory of data selection under weak supervision. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=HhfcNgQn6p.

Ramnath Kumar, Kushal Majmundar, Dheeraj Nagaraj, and Arun Sai Suggala. Stochastic re-weighted gradient descent via distributionally robust optimization. *arXiv preprint arXiv:2306.09222*, 2023.

Michael Langberg and Leonard J. Schulman. Universal ϵ-approximators for integrals. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '10, page 598–607, USA, 2010. Society for Industrial and Applied Mathematics. ISBN 9780898716986.

Evan Zheran Liu, Behzad Haghgoo, Annie S. Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. Just train twice: Improving group robustness without training group information. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 6781–6792. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/liu21f.html.

Ilya Loshchilov and Frank Hutter. Online batch selection for faster training of neural networks. In *ICLR 2016 Workshop Track*, 2016.

Katerina Margatina, Giorgos Vernikos, Loïc Barrault, and Nikolaos Aletras. Active learning by acquiring contrastive examples. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 650–663, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.51. URL https://aclanthology.org/2021.emnlp-main.51.

Sören Mindermann, Jan Brauner, Muhammed Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N. Gomez, Adrien Morisot, Sebastian Farquhar, and Yarin Gal. Prioritized training on points that are learnable, worth learning, and not yet learnt. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of*

the 39th International Conference on Machine Learning, volume 162 of *Proceedings of Machine Learning Research*, pages 15630–15649. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/mindermann22a.html.

Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6950–6960. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/mirzasoleiman20a.html.

Alexander Munteanu, Chris Schwiegelshohn, Christian Sohler, and David Woodruff. On coresets for logistic regression. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, pages 6561–6570. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/63bfd6e8f26d1d3537f4c5038264ef36-Paper.pdf.

Yonatan Oren, Shiori Sagawa, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust language modeling. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4227–4237, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1432. URL https://aclanthology.org/D19-1432.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 20596–20607. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/ac56f8fe9eea3e4a365f29f0f1957c55-Paper.pdf.

Ziheng Qin, Kai Wang, Zangwei Zheng, Jianyang Gu, Xiangyu Peng, Zhaopan Xu, Daquan Zhou, Lei Shang, Baigui Sun, Xuansong Xie, and Yang You. Infobatch: Lossless training speed up by unbiased dynamic data pruning. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=C61sk5LsK6.

Ravi Raju, Kyle Daruwalla, and Mikko Lipasti. Accelerating deep learning with dynamic data pruning. *arXiv preprint arXiv:2111.12621*, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, June 2022.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. In *International Conference on Learning Representations*, 2016.

Li Shen, Yan Sun, Zhiyuan Yu, Liang Ding, Xinmei Tian, and Dacheng Tao. On efficient training of large-scale deep learning models: A literature review. *arXiv preprint arXiv:2304.03589*, 2023.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 19523–19536. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/7b75da9b61eda40fa35453ee5d077df6-Paper-Conference.pdf.

Truong Thao Nguyen, Balazs Gerofi, Edgar Josafat Martinez-Noriega, François Trahay, and Mohamed Wahib. KAKURENBO: Adaptively hiding samples in deep neural network training. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 37900–37922. Curran Associates,

Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/7712b1075f5e0eae297702845714098f-Paper-Conference.pdf.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J. Gordon. An empirical study of example forgetting during deep neural network learning. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BJlxm30cKm.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothee 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. *arXiv preprint arXiv:2302.13971*, 2023.

Ziteng Wang, Jianfei Chen, and Jun Zhu. Efficient backpropagation with variance controlled adaptive sampling. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=gEwKAZZmSw.

Xiaobo Xia, Jiale Liu, Jun Yu, Xu Shen, Bo Han, and Tongliang Liu. Moderate coreset: A universal method of data selection for real-world data-efficient deep learning. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=7D5EECbOaf9.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. DoReMi: Optimizing data mixtures speeds up language model pretraining. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 69798–69818. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/dcba6be91359358c2355cd920da3fcbd-Paper-Conference.pdf.

Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large batch optimization for deep learning: Training BERT in 76 minutes. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=Syx4wnEtvH.

A Related Work

Static sampling. Methods to sampling statically can be based on geometry, uncertainty, error, meta optimization, dataset distillation, etc. With numerous studies on theoretical guarantees [Har-Peled and Mazumdar, 2004, Huang et al., 2023, Bachem et al., 2015], the coreset selection is designed to approximate original datasets with smaller (re-weighted) subsets, typically achieved by clustering in representation spaces [Xia et al., 2023, Abbas et al., 2023, Sorscher et al., 2022]. Uncertainty-based methods use probability metrics such as the confidence, entropy [Coleman et al., 2020] and distances to decision boundaries [Ducoffe and Precioso, 2018, Margatina et al., 2021, Dasgupta et al., 2019, Liu et al., 2021]. Sampling methods based on errors assume that training samples with more contributions to errors are more important. Errors are evaluated with merics such as forgetting events [Toneva et al., 2019], GRAND & EL2N score [Paul et al., 2021], and sensitivity [Langberg and Schulman, 2010, Munteanu et al., 2018]. As is discussed before, these static sampling methods require extra training, leading to considerable costs in both computation and memory.

Dynamic sampling. Methods to sampling dynamically typically leverage metrics based on losses and gradients along the training process. Loss-adaptive sampling re-weights data points during the training according to current losses [Katharopoulos and Fleuret, 2017, Jiang et al., 2019, Loshchilov and Hutter, 2016, Schaul et al., 2016] and historical losses[Oren et al., 2019, Sagawa et al., 2020]. To name a few, Ordered SGD [Kawaguchi and Lu, 2020] selects top-q samples in terms of the loss ranking per training step. InfoBatch [Qin et al., 2024] randomly prunes a portion of less informative samples with losses below the average and then re-scales the gradients. KAKURENBO [Thao Nguyen et al., 2023] combines current losses with the prediction accuracy and confidence to design a sampling framework with moving-back. Kumar et al. [2023] and Balaban et al. [2023] assign weights as functions of current losses based on the robust optimization theory. Attendu and Corbeil [2023] and Raju et al. [2021] use the exponential moving average over past losses for sampling. There are also studies adopting reference models, including Mindermann et al. [2022], Deng et al. [2023], Xie et al. [2023] and so on. These methods either exploit the information of losses inadequately, or require to train additional architectures. Gradient-based sampling methods involve (i) gradient matching, such as CRAIG [Mirzasoleiman et al., 2020] and GRAD-MATCH [Killamsetty et al., 2021], which approximate the "full" gradients computed on original datasets via the gradients computed on subsets; (ii) gradient adaption, where the sampling probability is basically determined by current scales of gradients [Hanchi et al., 2022, Katharopoulos and Fleuret, 2018]. A recent work [Wang et al., 2024] uses a intricate layer-wise sampling scheme with complex variance control. Obviously, gradient-based sampling methods lead to much more computation and memory overheads than loss-based methods.

***Set* level versus *batch* level.** Dynamic sampling methods can be divided into two categories based on the level where data selection is performed: (i) *set* level selection, to prune the whole dataset at the beginning of each epoch [Qin et al., 2024, Raju et al., 2021, Thao Nguyen et al., 2023, Attendu and Corbeil, 2023]; (ii) *batch* level selection, to sample subsets from the original batches for back propagations [Kawaguchi and Lu, 2020, Katharopoulos and Fleuret, 2017, 2018, Mindermann et al., 2022]. These two types of methods, facilitating training accelerations from different perspectives, are not mutually exclusive. However, to the best of our knowledge, we are not aware of any algorithms combining both of them.