Optimistic Online Learning for Data Mixture Optimization

Anonymous Authors

Abstract

Data mixture optimization is important for language model pretraining, where models are trained on heterogeneous corpora with domain-dependent usefulness. We propose *Optimistic DoReMi*, a minimal modification of DoReMi that replaces its multiplicative domain-weight update with an optimistic update inspired by Optimistic Hedge. Instead of using only the current per-domain excess-loss vector, our method updates weights using a linear prediction from the two most recent estimates. Experiments on SlimPajama and the Pile datasets show that this simple modification yields more stable and accurate mixtures over DoReMi across scales.

1. Introduction

The composition of pretraining data has a substantial effect on the generalization behavior of language models. Modern pretraining corpora are assembled from heterogeneous sources, including web text, code, books, scientific documents, forums, and reference data. Since different domains contribute different types of information, the choice of data mixture can significantly affect validation perplexity and downstream performance. As a result, data mixture optimization has become an important component of language model pretraining.

A growing line of work studies how to choose domain proportions efficiently. Offline approaches estimate mixture ratios before training, often by fitting surrogate models that predict performance from small-scale training runs [5, 7, 13]. Other methods adapt mixture weights during training using signals from the evolving model. DoGE, for example, updates domain weights using gradient information to estimate each domain's contribution to optimization, while DoReMi uses proxy models and group distributionally robust optimization (Group DRO) to upweight domains with large excess loss relative to a reference model [2, 11]. More recent work such as Chameleon further emphasizes the need for efficient and flexible data-mixing methods that avoid repeated expensive proxy retraining when the domain composition changes [12].

In this work, we focus on DoReMi as a representative proxy-based domain-reweighting method. DoReMi learns domain weights by training a small proxy model with a minimax objective over domains, where the adversarial signal is the excess loss relative to a reference model [11]. This framework provides a way to identify domains that remain informative during training, but its domain-weight update is based only on the current minibatch excess-loss estimate. Consequently, the learned mixture may be sensitive to the update step size and to short-term fluctuations in the estimated domain losses.

This raises the question of whether a more predictive update rule can improve the stability and quality of the learned mixture while preserving the simplicity of DoReMi. We propose Optimistic DoReMi, a minimal modification of DoReMi that replaces the standard multiplicative-weights update with an optimistic variant inspired by Optimistic Hedge [1, 3]. This variant uses a linear prediction based on the two most recent excess-loss vectors.

© .

Contributions. Our contributions are as follows:

- We recast the DoReMi domain-weight update as an online learning step over domain losses and introduce *Optimistic DoReMi*, a drop-in replacement of DoReMi that uses the extrapolated excess-loss signal $2g^{(t)} - g^{(t-1)}$ in place of the current signal $g^{(t)}$, following the update of Optimistic Hedge.
- Our approach is a drop-in replacement of DoReMi, leaving the reference-model, proxy-training, and target-training pipeline unchanged, adding no additional proxy runs or target-model cost.
- In experiments on both SlimPajama and the Pile, we isolate the effect of optimism and show that mixtures learned with Optimistic DoReMi yield lower validation log perplexity than standard DoReMi for 60M, 124M, and 1.2B target models. We further analyze the effect of the update step size η, showing that the optimistic update preserves the qualitative behavior of DoReMi while producing distinct domain-weight trajectories and final mixtures under matched hyperparameters.

2. Background: Domain Reweighting via DoReMi

Rather than learning the data mixture directly with the target model, DoReMi estimates domain weights using smaller auxiliary models and transfers the resulting mixture to the final training run. The method consists of three stages: first, a small reference model x_{ref} is trained on a fixed baseline mixture; second, a proxy model is trained with Group DRO in order to learn domain weights; finally, the target model is trained on data resampled according to the learned mixture.

Let $D_1, \ldots, D_k$ denote the training domains and let $\alpha \in \Delta^k$ be a sampling distribution over domains. DoReMi formulates domain reweighting as a minimax problem over the proxy model parameters θ and the domain weights α. The objective is based on excess loss relative to the reference model:

$$\min_{\theta} \max_{\alpha \in \Delta^k} \mathcal{L}(\theta, \alpha) = \sum_{i=1}^{k} \alpha_i \left[\frac{1}{\sum_{x \in D_i} |x|} \sum_{x \in D_i} \sum_{j=1}^{|x|} \left(\ell_{\theta,j}(x) - \ell_{\text{ref},j}(x) \right) \right],$$

where $\ell_{\theta,j}(x)$ and $\ell_{\text{ref},j}(x)$ denote the token-level losses of the proxy and reference models at position j of sequence x. This is a Group DRO objective over domains [8, 9], where the usual domain loss is replaced by the excess loss. The inner maximization emphasizes domains on which the proxy model performs poorly relative to the reference model, while the outer minimization trains the proxy model to reduce this worst-case excess loss.

Domains are upweighted when the proxy model exhibits higher loss than the reference model on that domain. This reduces the influence of domains that are uniformly easy, as well as domains whose difficulty is already captured by the reference loss. In this way, large excess loss serves as a signal that a domain may still be informative for training.

In practice, the objective is optimized using minibatch estimates of the per-domain excess losses. For stability and to satisfy the non-negativity requirement of the Group DRO update, DoReMi clips token-level excess losses at zero [11]. Thus, at step t, the update signal for domain i is computed as

$$g_i^{(t)} = \frac{1}{\sum_{x \in B_t \cap D_i} |x|} \sum_{x \in B_t \cap D_i} \sum_{j=1}^{|x|} \max \left\{ \ell_{\theta_t,j}(x) - \ell_{\text{ref},j}(x), 0 \right\}.$$

Since the maximization over α is linear, solving the inner problem exactly would place all mass on the domain with the largest excess loss. This is undesirable in practice, as it would eliminate exploration across domains. DoReMi therefore updates the domain weights using a multiplicative-weights rule, closely related to Hedge [3]:

$$\tilde{\alpha}_i^{(t+1)} = \alpha_i^{(t)} \exp\Big(\eta g_i^{(t)}\Big),$$

where the step size $\eta > 0$ controls the sensitivity of the update to the excess-loss signal. The resulting weights are normalized over domains before being used in the next proxy-model update, and the target training mixture is constructed by averaging the learned weights over the proxy training trajectory [11].

3. Optimistic DoReMi

We modify DoReMi only at the level of the domain-weight update. Our proposed modification is motivated by optimistic variants of no-regret learning, which incorporate a prediction of the next loss into the current update [1, 6]. In particular, Optimistic Hedge can be viewed as a recency-biased variant of multiplicative weights, where the update uses an extrapolated estimate of the next signal. Such optimism can improve adaptation when the observed sequence exhibits temporal consistency. In our setting, this corresponds to smooth evolution of the per-domain excess losses during proxy training, so that the previous update direction provides useful information about the next update.

The resulting proxy-training procedure is summarized in Algorithm 1.

Algorithm 1: Optimistic DoReMi Domain Reweighting

Data: Domain datasets $D_1, \ldots, D_k$, training steps T, batch size b, step size η
Result: Average domain weights $\frac{1}{T}\sum_{t=1}^{T} \alpha^{(t)}$
Initialize domain weights $\alpha^{(0)} \leftarrow \frac{1}{k}\mathbf{1}$;
for $t \leftarrow 1$ **to** T **do**
  Sample minibatch $B = \{x_1, \ldots, x_b\}$ from P_u;
  Compute per-domain excess losses $g_i^{(t)}$:

$$g_i^{(t)} \leftarrow \frac{1}{\sum_{x \in B \cap D_i} |x|} \sum_{x \in B \cap D_i} \sum_{j=1}^{|x|} \max\big(\ell_{\theta_{t-1},j}(x) - \ell_{\mathrm{ref},j}(x), 0\big)$$

  Perform optimistic update:

$$\alpha_i^{(t+1)} \propto \alpha_i^{(t)} \exp\big(\eta(2g_i^{(t)} - g_i^{(t-1)})\big)$$

  Update proxy model weights θ_{t+1} for objective $L(\theta_t, \alpha^{(t+1)})$;
end
return $\frac{1}{T}\sum_{t=1}^{T} \alpha^{(t)}$;

Let $g^{(t)} \in \mathbb{R}^k$ denote the clipped per-domain excess-loss vector used by DoReMi at step t. DoReMi updates the domain weights using only the current signal $g^{(t)}$. Following Optimistic

Hedge, we replace the current excess-loss vector with a linear prediction of the next vector based on the two most recent observations

$$m^{(t)} = g^{(t)} + \left(g^{(t)} - g^{(t-1)}\right) = 2g^{(t)} - g^{(t-1)},$$

with initialization $g^{(0)} = 0$. The multiplicative update becomes

$$\tilde{\alpha}_i^{(t+1)} = \alpha_i^{(t)} \exp\left(\eta m_i^{(t)}\right) = \alpha_i^{(t)} \exp\left(\eta(2g_i^{(t)} - g_i^{(t-1)})\right),$$

where i indicates the domain index and t the proxy training iteration.

Thus, the key difference from DoReMi is the replacement of the current excess-loss vector $g^{(t)}$ by the optimistic estimate $2g^{(t)} - g^{(t-1)}$.

This update has a simple interpretation. If a domain's excess loss increases across consecutive iterations, then the optimistic signal is larger than the current excess loss, causing that domain to be upweighted more rapidly. Conversely, if the excess loss is decreasing, the extrapolation yields a smaller multiplicative factor than the original DoReMi update rule. Optimistic DoReMi therefore reacts more adaptively to the excess-loss trend. The potential benefit is faster adaptation to persistent trends in domain usefulness during proxy training.

4. Results

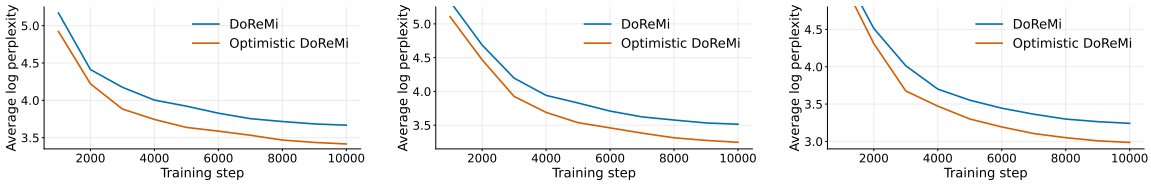
We evaluate whether our optimistic update rule improves DoReMi for language model pretraining. Experiments are conducted on SlimPajama [10] and The Pile [4], using decoder-only GPT-style models trained for 10,000 update steps. We compare the two settings: DoReMi reweighting and our Optimistic DoReMi variant. Following the proxy-based protocol of the original DoReMi paper [11], we train 60M reference and proxy models. In our experiments, the reference model is trained using a uniform baseline mixture over domains. We then train target models of sizes 60M, 124M, and 1.2B using the learned mixtures.

Training setup. All target models are trained for 10,000 update steps using decoder-only GPT-style architectures. The 60M model uses context length 512, embedding dimension 768, 3 Transformer layers, and 6 attention heads. The 124M model uses context length 1024, 12 layers, and 12 attention heads, with the same embedding dimension of 768. The 1.2B model uses context length 1024, embedding dimension 1600, 36 layers, and 25 attention heads. On SlimPajama, we use the seven-domain split consisting of ArXiv, Book, Common Crawl, C4, Github, StackExchange, and Wikipedia [10]. On the Pile, we use a 17-domain partition covering scientific, legal, web, conversational, code, and reference data sources [4].

Target models are trained using the domain weights produced by the corresponding proxy-stage reweighting procedure. The learned mixtures for SlimPajama and the Pile are reported in Appendix A, Tables 1 and 2, respectively.

On SlimPajama, Figure 1 shows that target models trained with Optimistic DoReMi weights achieve significantly lower validation perplexity than those trained with standard DoReMi weights across all three model scales. The learned mixtures differ: Optimistic DoReMi assigns more mass to Common Crawl and less to ArXiv, Github, StackExchange, and Wikipedia than standard DoReMi, as reported in Appendix A, Table 1.

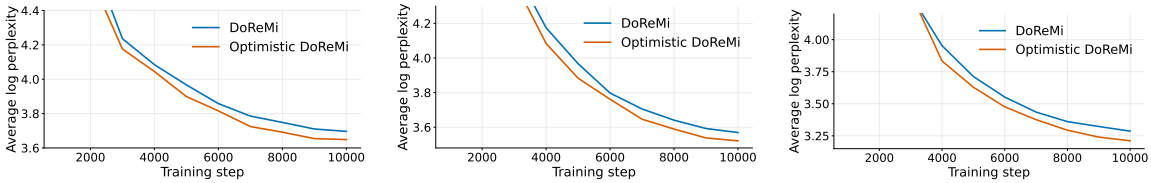
On the more complex Pile dataset, Optimistic DoReMi also achieves lower validation perplexity than standard DoReMi across all target model scales. Figure 2 shows that this improvement



(a) 60M target model. (b) 124M target model. (c) 1.2B target model.

Figure 1: Validation uniform average log perplexity on SlimPajama over training steps for target models trained with domain weights obtained from DoReMi and Optimistic DoReMi. Across all three target scales, optimistic variant achieves consistently lower perplexity throughout training.

holds consistently for the 60M, 124M, and 1.2B target models. These results suggest that optimistic reweighting improves standard DoReMi not only on SlimPajama, but also on a broader multi-domain corpus with a finer-grained domain partition.



(a) 60M target model. (b) 124M target model. (c) 1.2B target model.

Figure 2: Validation uniform average log perplexity on the Pile during target model training with mixtures learned by DoReMi and Optimistic DoReMi. The optimistic update gives lower perplexity across scales, showing that the improvement carries over to the finer-grained Pile domain split.

To study the sensitivity of domain reweighting to the update step size, we vary η for both DoReMi and Optimistic DoReMi and compare the learned mixtures under matched values of η. The evolution of the learned domain weights during proxy-model training for $\eta \in \{0.1, 0.5, 1.0\}$ is reported in Appendix A, Figure 3. Increasing η leads to faster and more aggressive changes in the learned domain weights, with probability mass shifting more strongly toward a smaller set of domains. Optimistic DoReMi exhibits broadly similar behavior to standard DoReMi, although the resulting trajectories differ in smoothness and in the final learned mixtures. In our main experiments, we use $\eta = 1.0$ to match the original DoReMi setup [11].

5. Conclusion

We introduce Optimistic DoReMi, an efficient modification of DoReMi for data mixture optimization. Our method replaces the standard multiplicative domain-weight update with an optimistic update that uses a linear prediction of the next excess-loss signal from recent proxy-training dynamics. We demonstrate that Optimistic DoReMi produces improved data mixtures on both SlimPajama and the Pile, yielding lower validation perplexity than standard DoReMi across scales. Since the method only changes the domain-weight update rule, it adds essentially no additional training cost and can be easily incorporated into existing proxy-based reweighting pipelines. In future work, we aim to study whether the learned domain weights transfer to downstream adaptation by fine-tuning 7B-scale models.

References

[1] Constantinos Daskalakis, Maxwell Fishelson, and Noah Golowich. Near-optimal no-regret learning in general games. *Advances in Neural Information Processing Systems*, 34:27604–27616, 2021.

[2] Simin Fan, Matteo Pagliardini, and Martin Jaggi. Doge: Domain reweighting with generalization estimation. *arXiv preprint arXiv:2310.15393*, 2023.

[3] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

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

[5] Ce Ge, Zhijian Ma, Daoyuan Chen, Yaliang Li, and Bolin Ding. Bimix: A bivariate data mixing law for language model pretraining. *arXiv preprint arXiv:2405.14908*, 2024.

[6] Ehsan Asadi Kangarshahi, Ya-Ping Hsieh, Mehmet Fatih Sahin, and Volkan Cevher. Let's be honest: An optimal no-regret framework for zero-sum games. In *International Conference on Machine Learning*, pages 2488–2496. PMLR, 2018.

[7] Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. Regmix: Data mixture as regression for language model pre-training. In *International Conference on Learning Representations*, volume 2025, pages 38305–38339, 2025.

[8] Yonatan Oren, Shiori Sagawa, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust language modeling. In *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, 2019.

[9] 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. *arXiv preprint arXiv:1911.08731*, 2019.

[10] Zhiqiang Shen, Tianhua Tao, Liqun Ma, Willie Neiswanger, Zhengzhong Liu, Hongyi Wang, Bowen Tan, Joel Hestness, Natalia Vassilieva, Daria Soboleva, et al. Slimpajama-dc: Understanding data combinations for llm training. *arXiv preprint arXiv:2309.10818*, 2023.

[11] Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy S Liang, Quoc V Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. *Advances in Neural Information Processing Systems*, 36:69798–69818, 2023.

[12] Wanyun Xie, Francesco Tonin, and Volkan Cevher. Chameleon: A flexible data-mixing framework for language model pretraining and finetuning. *arXiv preprint arXiv:2505.24844*, 2025.

[13] Jiasheng Ye, Peiju Liu, Tianxiang Sun, Jun Zhan, Yunhua Zhou, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. In *International Conference on Learning Representations*, volume 2025, pages 82263–82287, 2025.

Appendix A. Additional experimental details

Compared to DoReMi [11], in the experiments we use $\eta = 1.0$ as in their setups, and set their uniform smoothing parameter $c = 0$ to isolate the impact of the optimistic update. Tables 1 and 2 compare the learned domain mixtures on SlimPajama and the Pile produced by DoReMi and Optimistic DoReMi. Although both methods concentrate weight on a subset of domains, the optimistic update changes the final allocation, indicating that incorporating recent excess-loss trends can affect which domains are emphasized during training. Optimistic DoReMi assigns substantially more weight to Common Crawl on SlimPajama and slightly more weight to Github and USPTO Backgrounds on the Pile.

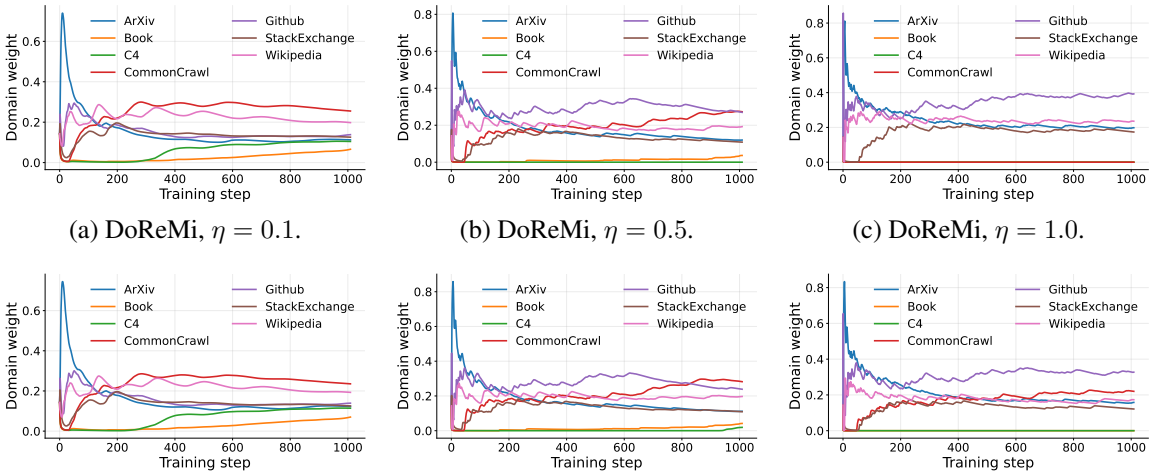
Domain	**DoReMi**	**Optimistic DoReMi**
ArXiv	0.201	0.158
Book	0.001	0.001
Common Crawl	0.001	0.217
C4	0.001	0.001
Github	0.385	0.331
StackExchange	0.169	0.121
Wikipedia	0.242	0.171

Table 1: Learned domain weights on SlimPajama for DoReMi and optimistic DoReMi.

Domain	**DoReMi**	**Optimistic DoReMi**
ArXiv	0.099	0.090
DM Mathematics	0.032	0.026
Enron Emails	0.001	0.021
Europarl	0.049	0.043
FreeLaw	0.001	0.001
Github	0.359	0.396
Gutenberg (PG-19)	0.001	0.001
HackerNews	0.001	0.001
NIH ExPorter	0.041	0.030
PhilPapers	0.131	0.122
Pile-CC	0.001	0.001
PubMed Abstracts	0.215	0.195
PubMed Central	0.001	0.001
StackExchange	0.001	0.001
Ubuntu IRC	0.018	0.007
USPTO Backgrounds	0.048	0.063
Wikipedia (en)	0.001	0.001

Table 2: Learned domain weights on the Pile for DoReMi and optimistic DoReMi.

Figure 3 provides the domain-weight trajectories for DoReMi and Optimistic DoReMi under $\eta \in \{0.1, 0.5, 1.0\}$. Across both DoReMi and Optimistic DoReMi, larger values of η lead to sharper early reweighting, while the optimistic update produces different final mixtures under matched step sizes.



(a) DoReMi, $\eta = 0.1$. (b) DoReMi, $\eta = 0.5$. (c) DoReMi, $\eta = 1.0$.

(d) Optimistic DoReMi, $\eta = 0.1$. (e) Optimistic DoReMi, $\eta = 0.5$. (f) Optimistic DoReMi, $\eta = 1.0$.

Figure 3: Evolution of learned domain weights on SlimPajama under different values of the domain-weight update parameter η. The top row shows standard DoReMi and the bottom row shows Optimistic DoReMi, each for $\eta \in \{0.1, 0.5, 1.0\}$.