Optimizing regularization path in knowledge distillation

M. Gorpinich, O. Yu. Bakhteev, V. V. Strijov

Moscow Institute of Physics and Technology

2021

Distillation knowledge

Purpose

To propose a metaparameter optimization method in the knowledge distillation task.

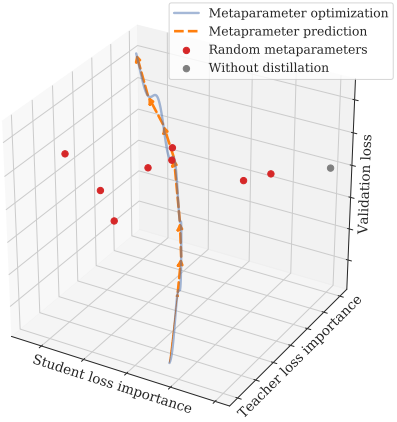
Problem

Metaparameter selection is a computationally expensive problem.

Solution

Formulate bi-level optimization problem. We solve the problem by gradient-based methods. To accelerate the computationally expensive metaparameter optimization we predict with linear models.

Optimization procedure



A principle idea of the proposed method. The metaparameters control the final loss of the model. Instead of optimize the metaparameters straightforwardly we analyze the optimization trajectory and predict it using linear model.

Problem description

Metaparameters $\boldsymbol{\lambda}$ are the parameters of knowledge distillation optimization problem. Namely, the coefficients before terms in loss function and the temperature:

$$\boldsymbol{\lambda} = [\lambda_1, T].$$

The temperature is a factor of logits of models in softmax function.

Knowledge distillation is the model parameter optimization problem. It takes:

1. information of the initial dataset;
2. information of the cumbersome model.

The teacher model has a more complex structure. It is trained using the initial dataset. The student model has a simpler structure. We optimize it by transferring the knowledge of the cumbersome model to it.

Main references

- Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. "Distilling the Knowledge in a Neural Network". In: *CoRR* abs/1503.02531 (2015). URL: http://arxiv.org/abs/1503.02531.
- Jelena Luketina et al. "Scalable Gradient-Based Tuning of Continuous Regularization Hyperparameters". In: *CoRR* abs/1511.06727 (2015). URL: http://arxiv.org/abs/1511.06727.
- Oleg Yu. Bakhteev and Vadim V. Strijov. "Comprehensive analysis of gradient-based hyperparameter optimization algorithms". In: *Ann. Oper. Res* 289.1 (2020), pp. 51–65.
- Marcin Andrychowicz et al. "Learning to learn by gradient descent by gradient descent". In: *CoRR* abs/1606.04474 (2016). URL: http://arxiv.org/abs/1606.04474.

Distillation problem statement

There is given a dataset

$$\mathfrak{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m},\ \mathbf{x}_i \in \mathbb{R}^n, \qquad y_i \in \mathbb{Y} = \{1, \ldots, K\}, \qquad \mathfrak{D} = \mathfrak{D}_{\text{train}} \sqcup \mathfrak{D}_{\text{val}}.$$

f is the fixed teacher model, **g** is the student model.

Definition 1. Let function $D : \mathbb{R}^s \to \mathbb{R}_+$ defines the distance between the student model **g** and the teacher model **f**. D-distillation of a student model is a student model parameter optimization problem that minimizes the function D.

Loss functions

Train (distillation) loss

$$\mathcal{L}_{\text{train}}(\mathbf{w}, \boldsymbol{\lambda}) = -\lambda_1 \sum_{(\mathbf{x},y)\in\mathfrak{D}_{\text{train}}} \underbrace{\sum_{k=1}^{K} y^k \log \frac{e^{\mathbf{g}(\mathbf{x},\mathbf{w})_k}}{\sum_{j=1}^{K} e^{\mathbf{g}(\mathbf{x},\mathbf{w})_j}}}_{\text{classification term}} - (1-\lambda_1) \sum_{(\mathbf{x},y)\in\mathfrak{D}_{\text{train}}} \underbrace{\sum_{k=1}^{K} \frac{e^{\mathbf{f}(\mathbf{x})_k / T}}{\sum_{j=1}^{K} e^{\mathbf{f}(\mathbf{x})_j / T}} \log \frac{e^{\mathbf{g}(\mathbf{x},\mathbf{w})_k / T}}{\sum_{j=1}^{K} e^{\mathbf{g}(\mathbf{x},\mathbf{w})_j / T}}}_{\text{distillation term}},$$

Validation loss

$$\mathcal{L}_{\text{val}}(\mathbf{w}, \boldsymbol{\lambda}) = - \sum_{(\mathbf{x},y)\in\mathfrak{D}_{\text{val}}} \sum_{k=1}^{K} y^k \log \frac{e^{\mathbf{g}(\mathbf{x},\mathbf{w})_k / T_{\text{val}}}}{\sum_{j=1}^{K} e^{\mathbf{g}(\mathbf{x},\mathbf{w})_j / T_{\text{val}}}}$$

Metaparameter set:

$$\boldsymbol{\lambda} = [\lambda_1, T]$$

Optimization problem:

$$\hat{\boldsymbol{\lambda}} = \arg\min_{\boldsymbol{\lambda}\in\mathbb{R}^2} \mathcal{L}_{\text{val}}(\hat{\mathbf{w}}, \boldsymbol{\lambda}), \tag{1}$$

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^s} \mathcal{L}_{\text{train}}(\mathbf{w}, \boldsymbol{\lambda}).$$

Special case of D-distillation

Lemma 1. When $\lambda_1 = 0$ we minimize loss function which is a D-distillation with $D = D_{KL}\left(\sigma\left(\mathbf{f}/T\right), \sigma\left(\mathbf{g}/T\right)\right)$, where σ is a softmax function.
When $\lambda_1 = 0$ we have:

$$\mathcal{L}_{\text{train}}(\mathbf{w}, \boldsymbol{\lambda}) = \sum_{(\mathbf{x},y)\in\mathfrak{D}_{\text{train}}} \sum_{k=1}^{K} \frac{e^{\mathbf{f}(\mathbf{x})_k/T}}{\sum_{j=1}^{K} e^{\mathbf{f}(\mathbf{x})_j/T}} \log \frac{e^{\mathbf{g}(\mathbf{x},\mathbf{w})_k/T}}{\sum_{j=1}^{K} e^{\mathbf{g}(\mathbf{x},\mathbf{w})_j/T}}$$

$$= D_{KL}\left(\sigma(\mathbf{f}(\mathbf{x})/T), \sigma(\mathbf{g}(\mathbf{x},\mathbf{w})/T)\right).$$

The function $D_{KL}\left(\sigma\left(\mathbf{f}/T\right), \sigma\left(\mathbf{g}/T\right)\right)$ defines the distance between logits of model $\mathbf{f}$ and model $\mathbf{g}$. Therefore, the definition of the D-distillation is satisfied.

Gradient-based optimization

Definition 2. *Optimization operator* is an algorithm U of parameter vector $\mathbf{w}'$ selection using the parameters on the next step $\mathbf{w}$:

$$\mathbf{w}' = U(\mathbf{w}).$$

Optimize the parameters $\mathbf{w}$ using η optimization steps:

$$\hat{\mathbf{w}} = U \circ U \circ \cdots \circ U(\mathbf{w}_0, \boldsymbol{\lambda}) = U^{\eta}(\mathbf{w}_0, \boldsymbol{\lambda}),$$

where $\mathbf{w}_0$ is an initial value of parameter vector $\mathbf{w}$. Redefine the minimization problem using the definition of operator U:

$$\hat{\boldsymbol{\lambda}} = \arg \max_{\boldsymbol{\lambda} \in \mathbb{R}^3} \mathcal{L}_{\text{val}}\big(U^{\eta}(\mathbf{w}_0, \boldsymbol{\lambda})\big), \qquad U(\mathbf{w}, \boldsymbol{\lambda}) = \mathbf{w} - \gamma \nabla \mathcal{L}_{\text{train}}(\mathbf{w}, \boldsymbol{\lambda}).$$

Update metaparameters successively according to the rule:

$$\boldsymbol{\lambda}' = \boldsymbol{\lambda} - \gamma_{\boldsymbol{\lambda}} \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\text{val}}(U(\mathbf{w}, \boldsymbol{\lambda}), \boldsymbol{\lambda}) = \boldsymbol{\lambda} - \gamma_{\boldsymbol{\lambda}} \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\text{val}}(\mathbf{w} - \gamma \nabla \mathcal{L}_{\text{train}}(\mathbf{w}, \boldsymbol{\lambda}), \boldsymbol{\lambda}). \tag{2}$$

Hypothesis: gradient optimization regularization path can be approximated with locally linear model:

$$\boldsymbol{\lambda}' = \boldsymbol{\lambda} + \mathbf{c}^{\top} \begin{pmatrix} z \\ 1 \end{pmatrix},$$

where z is a remainder of iteration number divided by the period of linear model training, $\mathbf{c}$ is a vector of linear model parameters.

Resulting algorithm

Algorithm 1 Metaparameter optimization

Require: number e_1 of iterations to use gradient-based optimization

Require: number e_2 of iterations to use linear predictions for metaparameters $\boldsymbol{\lambda}$

1: **while** not converged **do**

2: Optimize $\boldsymbol{\lambda}$ and $\mathbf{w}$ for e_1 iterations solving a bi-level optimization problem

3: **traj** =trajectory of $(\nabla\boldsymbol{\lambda})$ changes during optimization;

4: Set $\mathbf{z} = [1, \ldots, e_1]^T$

5: Optimize $\mathbf{c}$ using least square method:

$$\hat{\mathbf{c}} = \arg\min_{\mathbf{c}\in\mathbb{R}^2} ||\mathbf{traj} - \mathbf{z} \cdot c_1 + c_2||_2^2$$

6: Optimize $\mathbf{w}$ and predict $\boldsymbol{\lambda}$ for e_2 iterations using linear model with parameters $\mathbf{c}$.

7: **end while**

Correctness of the approximation by a linear model

Theorem 1. If function $\mathcal{L}_{\text{train}}(\mathbf{w}, \boldsymbol{\lambda})$ is smooth and convex and its Hessian $\mathbf{H} = \nabla^2_{\mathbf{w}} \mathcal{L}_{\text{train}}$ is invertible and can be well approximated by identity, $\mathbf{H} \approx \mathbf{I}$, then greedy algorithm (2) finds optimum solution of the bi-level problem (1). If there is a domain $\in \mathbb{R}^2$ in metaparameter space where the gradient of metaparameters can be well approximated by a constant, then the optimization is linear w.r.t. the metaparameters.

Experiment setup

Datasets

Synthetic dataset, CIFAR-10 (whole dataset and reduced training subset), Fashion-MNIST

Optimization methods

1) optimization without distillation;
2) optimization with randomly initialized metaparameter values. The metaparameters were sampled from uniform distribution
$$\lambda_1 \sim \mathcal{U}(0; 1), \quad T \sim \mathcal{U}(0.1, 10).$$
3) optimization with "naive" metaparameter assignment: setting
$$\lambda_1 = 0.5, T = 1;$$
4) gradient-based optimization;
5) proposed method with $e_1 = e_2 = 10$.

The external criterion:

$$\text{accuracy} = \frac{1}{m} \sum_{i=1}^{m} [\mathbf{g}(\mathbf{x}_i, \mathbf{w}) = y_i].$$

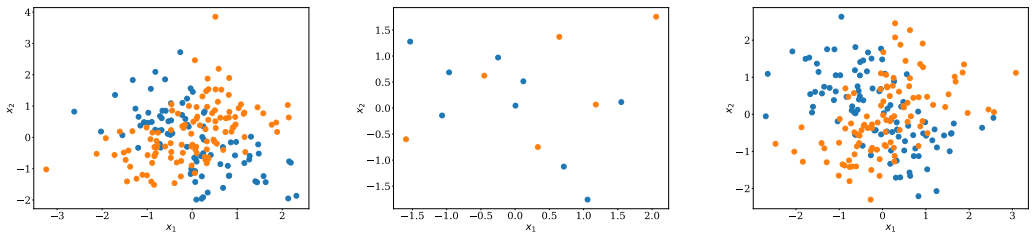
Experiment on the synthetic data

The dataset

$$\mathfrak{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m},\ x_{ij} \in \mathcal{N}(0,1), \qquad j = 1,2, \qquad x_{i3} = [\text{sign}(x_{i1}) + \text{sign}(x_{i2}) > 0],$$

$$y_i = \text{sign}(x_{i1} \cdot x_{i2} + \delta) \in \mathbb{Y}.$$

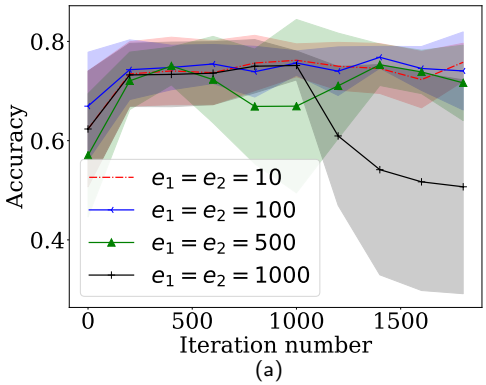
The size of student model dataset is significantly smaller then the one of the teacher model.

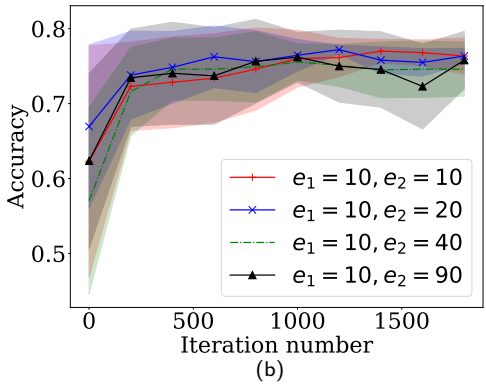


Visualization of a) teacher model dataset; b) student model dataset; c) test part

Optimization procedure tuning

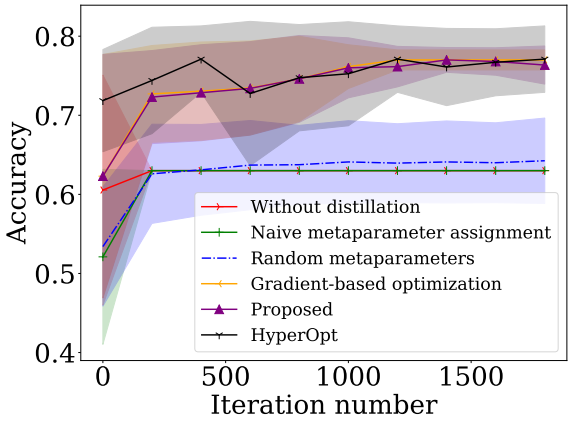
Model accuracy with e_1 and e_2 values: a) $e_1 = e_2$; b) variation of e_2 with $e_1 = 10$.





The best accuracy is obtained with $e_1 = e_2 = 10$.

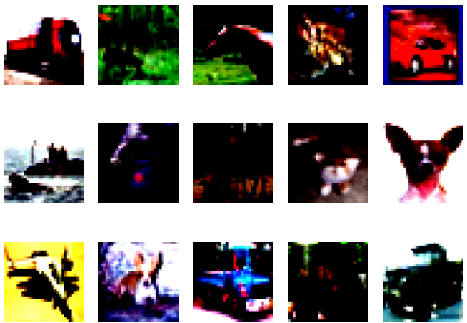
Comparison of optimization approaches



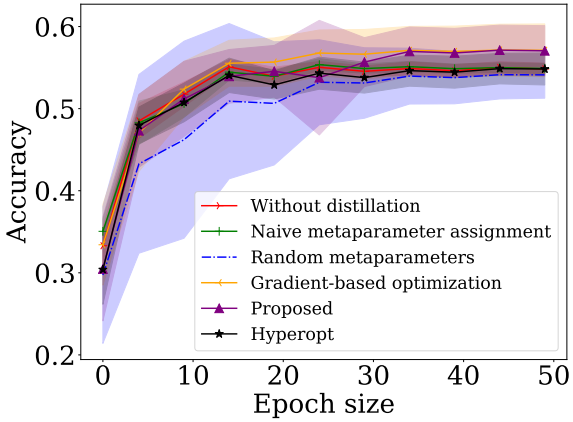
The results obtained with gradient-based optimization are close to the ones obtained after usage of linear models.

CIFAR-10 dataset

The method is evaluated on the whole and reduced ($|\mathfrak{D}_{\text{train}}| = 12800$) CIFAR-10 dataset of 60000 color images 32×32 in 10 mutually exclusive classes. Each class contains 6000 images. Dataset is divided into the train part and test part. Train part contains 5000 images. Test part contains 1000 images.



Experiment results on CIFAR-10 dataset



The model accuracy of training using distillation is higher than without it. We obtain the best accuracy if the proposed method is used.

Experiment results

Method	Synthetic dataset	Fashion-MNIST	Reduced CIFAR-10	CIFAR-10
Without distillation	0.63 (0.63)	0.87 (0.88)	0.55 (0.56)	0.65 (0.66)
Naive metaparameters	0.63 (0.63)	0.87 (0.88)	0.55 (0.56)	0.66 (0.67)
Random metaparameters	0.64 (0.72)	0.79 (0.88)	0.54 (0.57)	0.64 (0.67)
Gradient-based optimization	**0.77** (0.78)	**0.88** (0.89)	**0.57** (0.61)	**0.70** (0.72)
Hyperopt	**0.77** (0.78)	0.87 (0.88)	0.55 (0.58)	-
Proposed	0.76 (0.78)	**0.88** (0.89)	**0.57**	**0.70** (0.72)

The both the proposed method and gradient-based method give quite competitive results. The gradient-based methods are preferable because they have similar performance but require fewer optimization iterations.

Conclusion

- The usage of gradient-based methods to optimize metaparameters of the distillation task is analyzed.
- The hypothesis about the approximation of regularization path is tested.
- The computational experiment showed that the proposed approach to metaparameter optimization can be applied to the distillation task.
- The possibility of metaparameter approximation with linear models is confirmed.
- The further investigation and the analysis of the quality of metaparameter optimization trajectory approximation with more complex predictive models are planned.