Meta-Learning for Domain Generalization in Semantic Parsing

Bailin Wang[†], Mirella Lapata[†] and Ivan Titov[†§]

[†] ILCC, University of Edinburgh, [§] ILLC, University of Amsterdam

Edinburgh NLP
University of Edinburgh Natural Language Processing

UNIVERSITY OF AMSTERDAM

Semantic Parsing for Databases

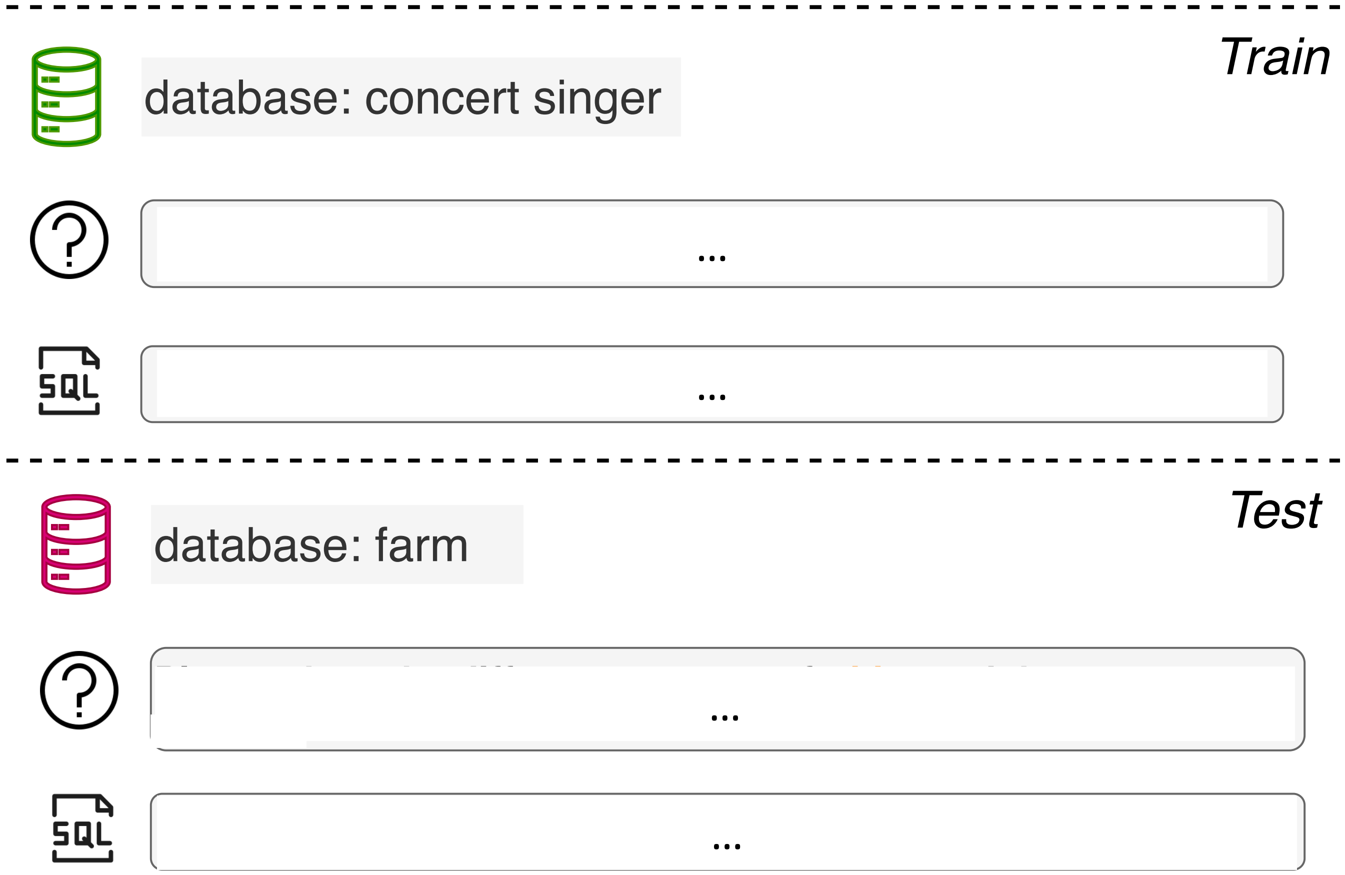
database: concert singer

Show all *countries* and the number of *singers* in each *country*.

SELECT *Country* , count(*) FROM *Singer* GROUP BY *Country*

Task: *translating natural language utterance to SQL queries.*

Cross-Domain Text-to-SQL Parsing



Domain Generalization

** a parser needs to generalize to unseen domains.*

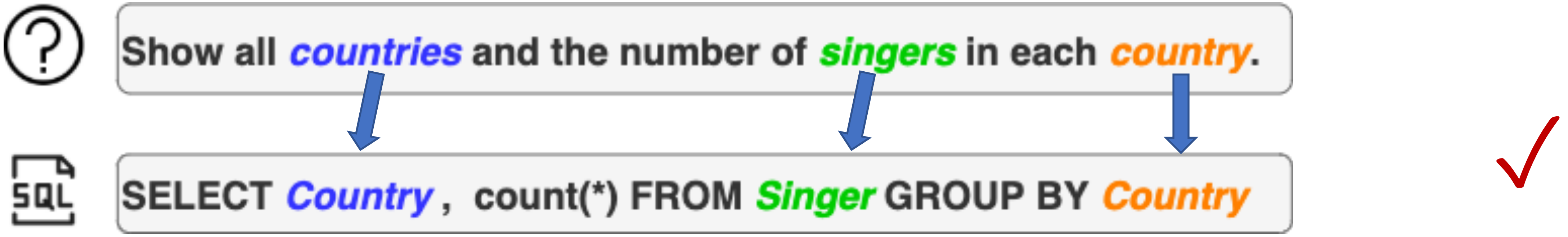
** modern parsers have a gap of more than 25% between in- and cross-domain performance*

Cross-Lingual Cross-Domain Text-to-SQL Parsing

- *Utterance and database schemas are in different languages*
- In the left figure, utterances are in Chinese whereas database schemas are in English

Previous work: specialized models for schema linking

Mono-lingual Setting



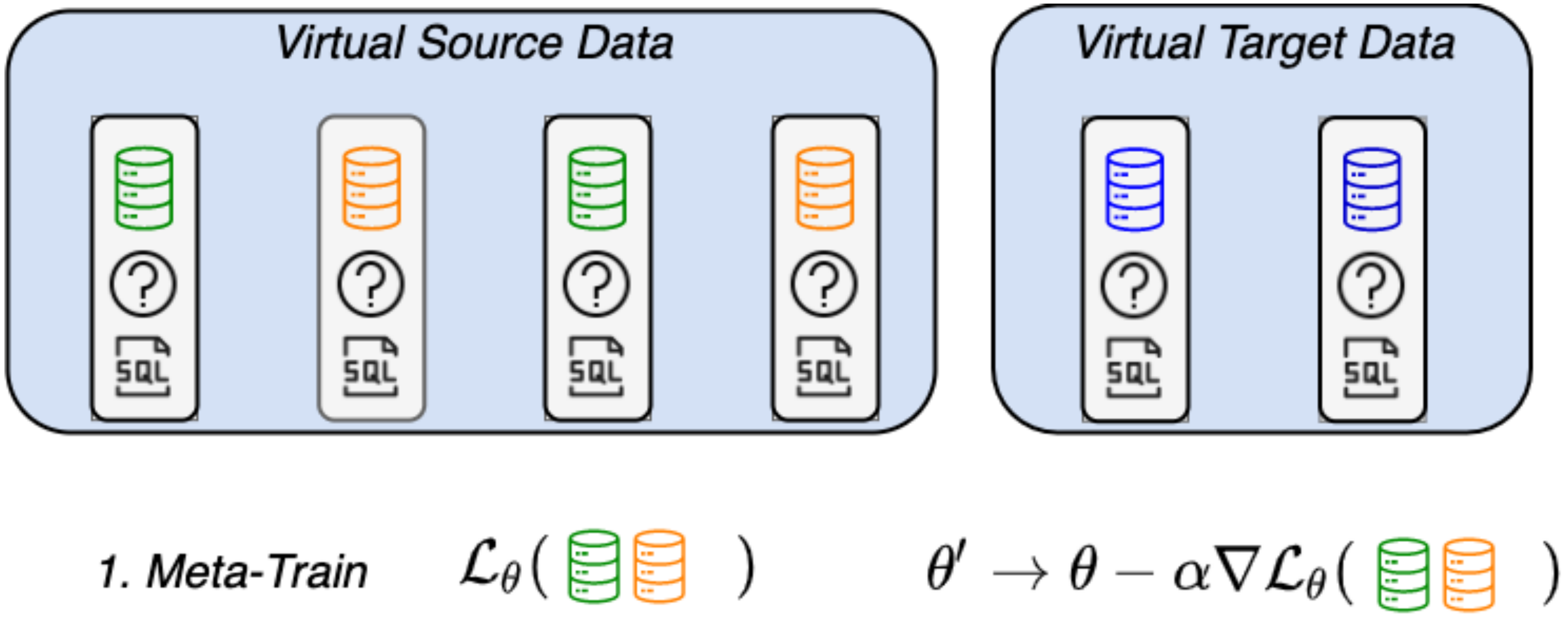
Cross-lingual Setting

Can We Optimize for Domain Generalization without Changing Models?

(for both mono- and cross-lingual settings)

Construct Virtual Tasks for Meta-Learning

Meta-Learning Objective



Meta-Learning Objective

1. *Meta-Train* $\mathcal{L}_\theta(\quad)$ $\theta' \rightarrow \theta - \alpha \nabla \mathcal{L}_\theta(\quad)$

2. *Meta-Test* $\mathcal{L}_{\theta'}(\quad)$

Meta-Learning Objective

1. *Meta-Train* $\mathcal{L}_{\theta}(\quad)$ $\theta' \rightarrow \theta - \alpha \nabla \mathcal{L}_{\theta}(\quad)$

2. *Meta-Test* $\mathcal{L}_{\theta'}(\quad)$

3. *Final Loss* $\mathcal{L}_{maml}(\theta) = \mathcal{L}_{\theta}(\quad) + \mathcal{L}_{\theta'}(\quad)$

Meta-Learning Objective: DG-MAML

1. Meta-Train $\mathcal{L}_{\theta}(\quad)$ $\theta' \rightarrow \theta - \alpha \nabla \mathcal{L}_{\theta}(\quad)$

2. Meta-Test $\mathcal{L}_{\theta'}(\quad)$

3. Final Loss $\mathcal{L}_{maml}(\theta) = \mathcal{L}_{\theta}(\quad) + \mathcal{L}_{\theta'}(\quad)$

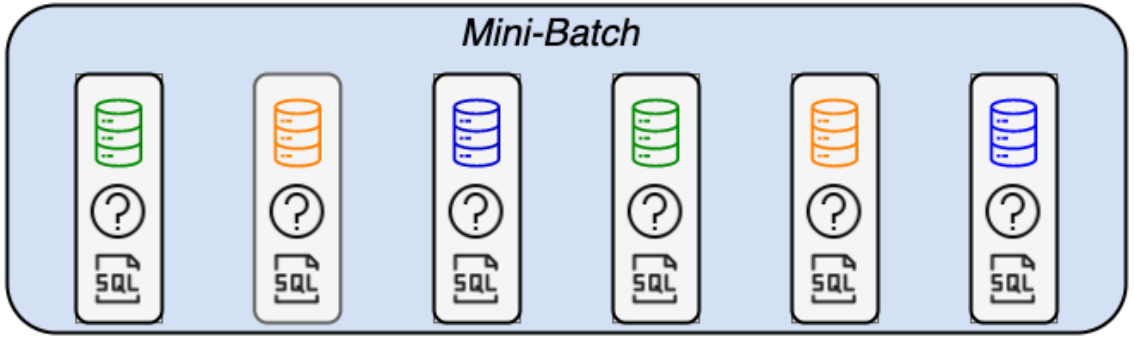
Analysis of DG-MAML

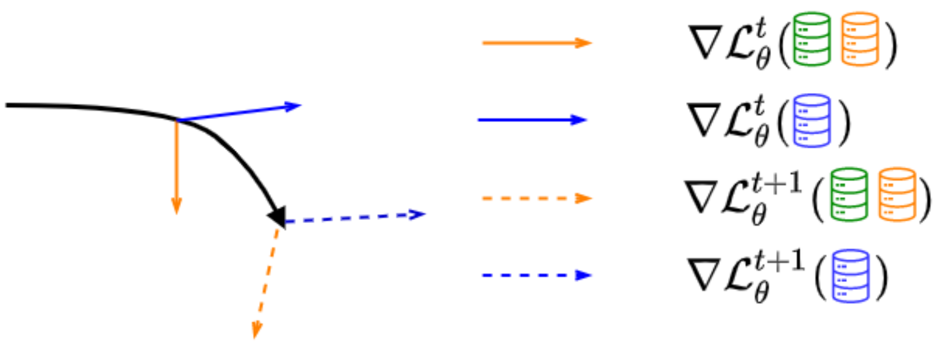
$$
\begin{aligned}
\mathcal{L}_{maml}(\theta) &= \mathcal{L}_{\theta}(\quad) + \mathcal{L}_{\theta'}(\quad) \\
&\approx \mathcal{L}_{\theta}(\quad) + \mathcal{L}_{\theta}(\quad) - \alpha \nabla \mathcal{L}_{\theta}(\quad) \cdot \nabla \mathcal{L}_{\theta}(\quad)
\end{aligned}
$$

[Li et al., 2018]

Gradient Updates of DG-MAML

Supervised Learning



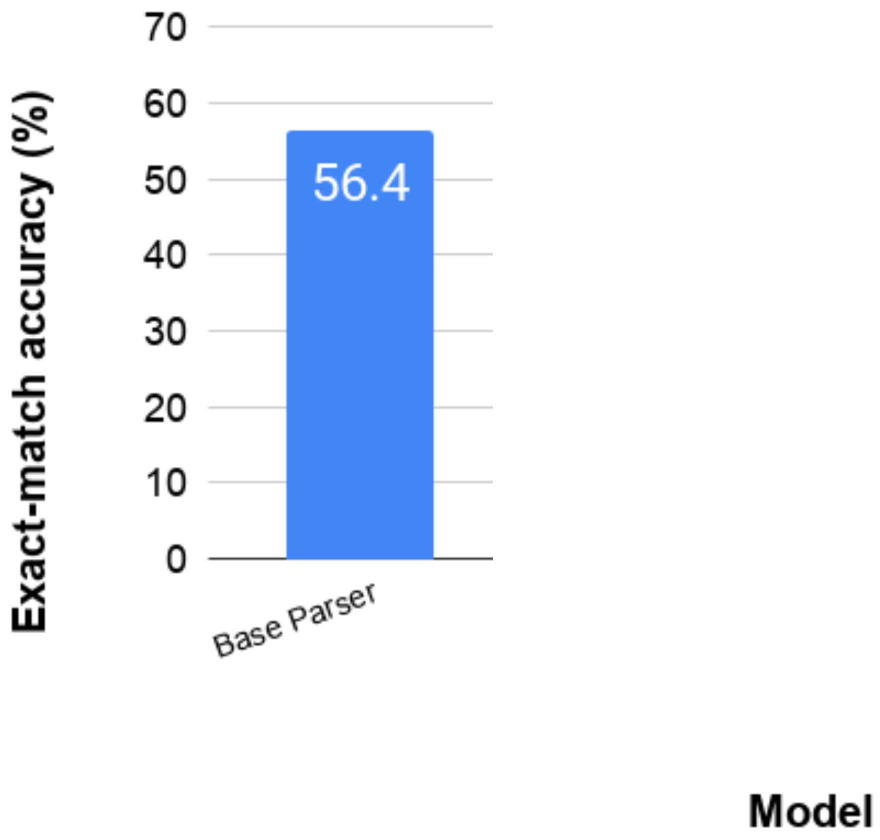


Experiments

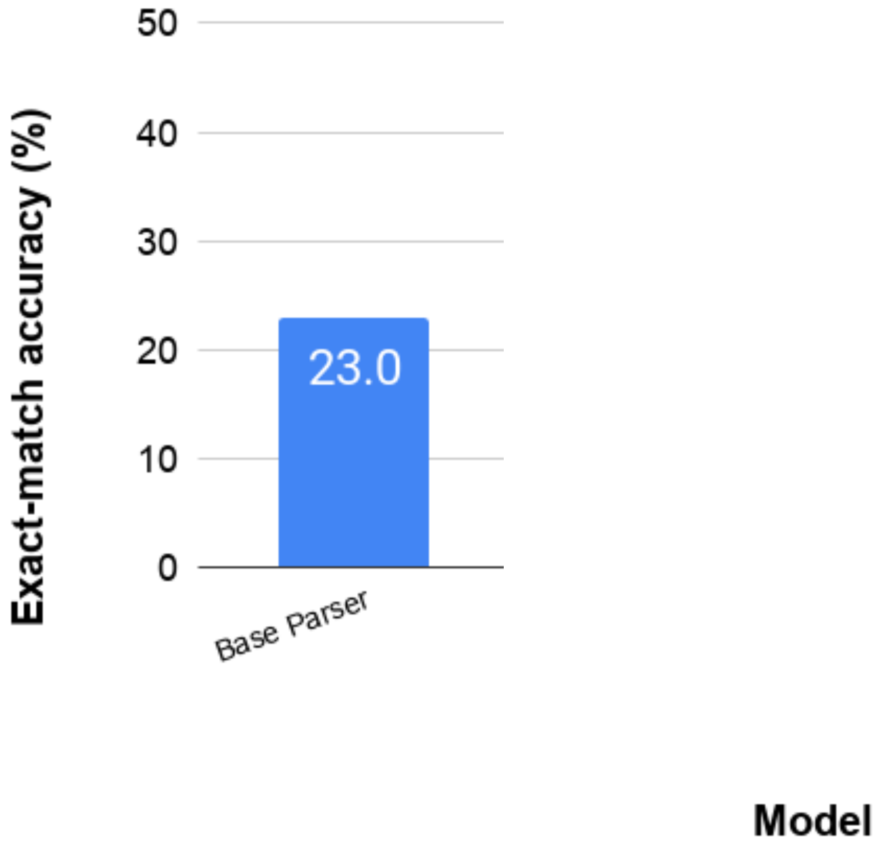
Datasets

	Cross-Domain	*Cross-Lingual*
Spider		
Chinese Spider		

Results on Spider



Results on Chinese Spider



Analysis: In-Domain vs. Out-of-Domain

We create an in-domain setting from the Spider dataset.

- *Does the parser struggle out-of-domain?*

  YES

  In-domain vs. out-of-domain performance: 56.4% vs 78.2%

- *Does DG-MAML hurt in-domain performance?*

  NO

  DG-MAML leads to a modest improvement (+1.1%)

Key Takeaways

- Meta-learning can be useful beyond few-shot learning; we show it can also be used to promote domain generalization for semantic parsing.
- Without changing model architectures, DG-MAML can boost the performance of cross-domain parsers in mono– and cross-lingual settings.
- Code: https://github.com/berlino/tensor2struct-public
- Our recent work on extending DG-MAML for compositional generalization is accepted by ACL2021

Basic Ideas

- We aim at *directly* optimizing for *domain generalization (DG)* via a meta-learning objective, dubbed DG-MAML.
- By constructing a set of *virtual cross-domain parsing* tasks, the objective encourage generalization to unseen domains in each task.

DG-MAML Training Algorithm

Algorithm 1 DG-MAML Training Algorithm

Require: Training databases $\mathcal{D}$
Require: Learning rate α
1: **for** step $\leftarrow 1$ **to** T **do**

8: **end for**

- Given a a set of examples from databases D

DG-MAML Training Algorithm

Algorithm 1 DG-MAML Training Algorithm

Require: Training databases $\mathcal{D}$
Require: Learning rate α
1: **for** step $\leftarrow$ 1 **to** T **do**
2: Sample a task τ of $(\mathcal{D}_s^\tau, \mathcal{D}_t^\tau)$ from $\mathcal{D}$
8: **end for**

- Given a a set of examples from databases D
- We first sample a virtual task from D

DG-MAML Training Algorithm

Algorithm 1 DG-MAML Training Algorithm

Require: Training databases $\mathcal{D}$
Require: Learning rate α
for step $\leftarrow 1$ **to** T **do**
    Sample a task τ of $(\mathcal{D}_s^\tau, \mathcal{D}_t^\tau)$ from $\mathcal{D}$
    Sample mini-batch $\mathcal{B}_s^\tau$ from $\mathcal{D}_s^\tau$
    Sample mini-batch $\mathcal{B}_t^\tau$ from $\mathcal{D}_t^\tau$

end for

- Given a a set of examples from databases D
- We first sample a virtual task from D
- Sample examples from virtual source and target databases

DG-MAML Training Algorithm

Algorithm 1 DG-MAML Training Algorithm

Require: Training databases $\mathcal{D}$

Require: Learning rate α

```
for step ← 1 to T do
    Sample a task τ of (D_s^τ, D_t^τ) from D
    Sample mini-batch B_s^τ from D_s^τ
    Sample mini-batch B_t^τ from D_t^τ
    Meta-train update:
        θ' ← θ − α ∇_θ L_{B_s^τ}(θ)
    Compute meta-test objective:
        L_τ(θ) = L_{B_s}(θ) + L_{B_t}(θ')
    Final Update:
        θ ← Update(θ, ∇_θ L_τ(θ))
end for
```

- Given a a set of examples from databases D
- We first sample a virtual task from D
- Sample examples from virtual source and target databases
- Update parameters using a MAML objective

MAML Objective

- Meta-Train: one step of SGD in the virtual source domains

$$\boldsymbol{\theta}' \leftarrow \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta})$$

- Meta-Test: evaluate the parameters in the virtual target domains

$$\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}')$$

- Final objective: joint loss on both virtual source and target domains

$$\mathcal{L}_{\mathcal{T}}(\boldsymbol{\theta}) = \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}) + \mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}')$$

MAML Objective

$$\mathcal{L}_{\mathcal{T}}(\boldsymbol{\theta}) = \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}) + \mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}')$$

Intuition:

- optimize towards the better source *and* target domain performance simultaneously
- gradient step in the source domain should be beneficial to the performance of the target domain as well.

MAML Objective vs. Supervised Learning

$$\mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}) + \mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}') \qquad\qquad \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}) + \mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta})$$

Comparison:

- Supervised learning objective (*right*) does not pose any constraints on the gradient update.

- MAML objective (*left*) can be viewed as a regularization of gradient updates.

Analysis of DG-MAML

First-order Taylor series expansion:

$$
\begin{aligned}
\mathcal{L}_{\tau}(\boldsymbol{\theta}) =& \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}) + \mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}') \\
=& \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}) + \mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta})) \\
\approx& \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}) + \mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}) - \alpha(\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}) \cdot \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}))
\end{aligned}
$$

DG-MAML further tries to maximize $\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}) \cdot \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta})$, the dot product between the gradients of source and target domain. That is, it encourages gradients to generalize between source and target domain within each task τ.

First-Order Approximation: DG-FMAML

The gradient of DG-MAML requires second derivatives:

$$\begin{aligned}\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\tau}(\boldsymbol{\theta}) &= \nabla_{\boldsymbol{\theta}}\boldsymbol{\theta}'\nabla_{\boldsymbol{\theta}'}\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}') + \nabla_{\boldsymbol{\theta}}\mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta}) \\ &= \left(\boldsymbol{I} - \alpha\nabla^2_{\boldsymbol{\theta}}\mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta})\right)\nabla_{\boldsymbol{\theta}'}\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}') + \nabla_{\boldsymbol{\theta}}\mathcal{L}_{\mathcal{B}_s}(\boldsymbol{\theta})\end{aligned}$$

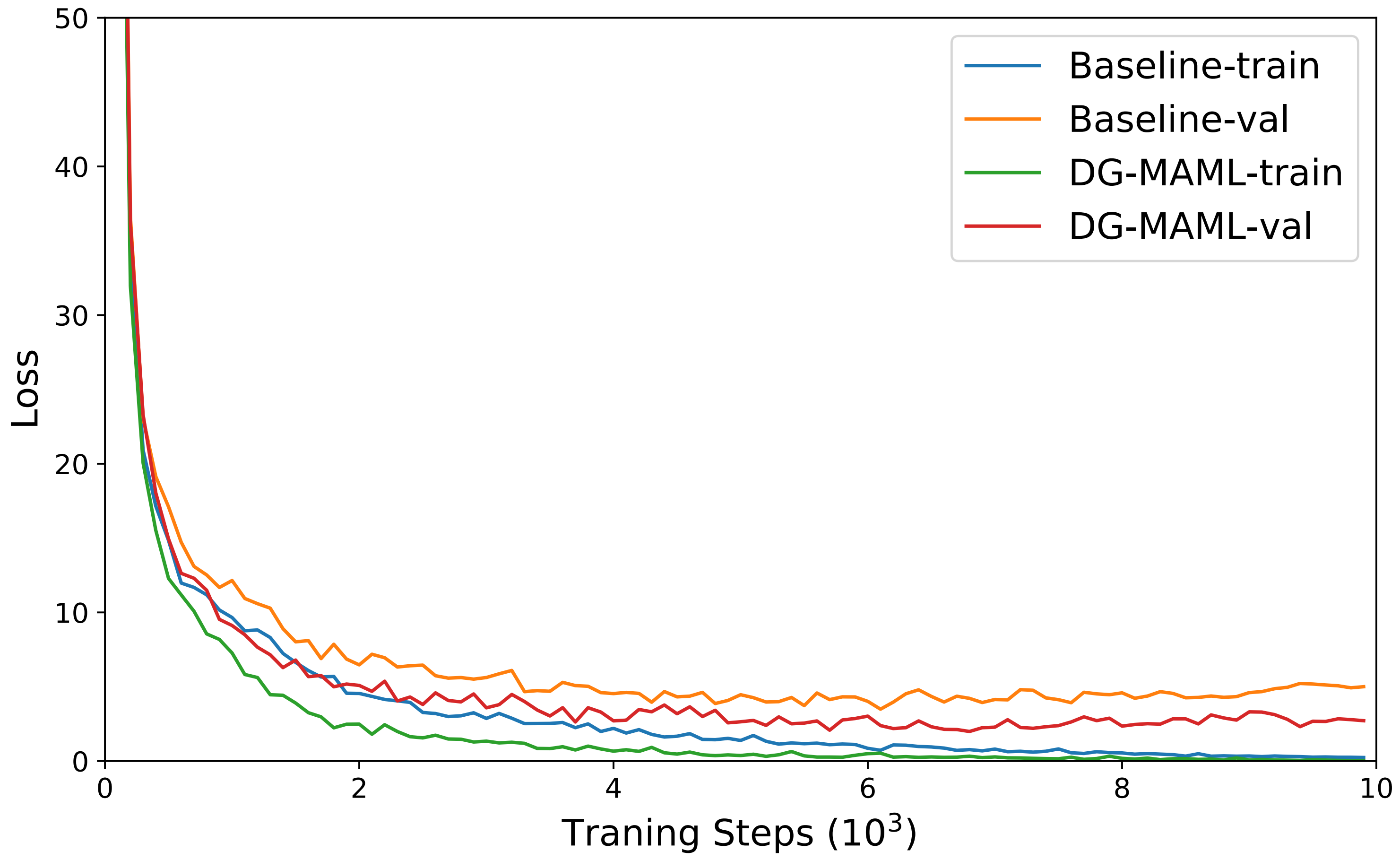
Inspired by Reptile, we consider the alternative of ignoring this second-order term and simply assume that $\nabla_{\boldsymbol{\theta}}\boldsymbol{\theta}' = \boldsymbol{I}$.

First-Order Approximation: DG-FMAML

$\mathcal{L}_{maml}(\theta) = \mathcal{L}_{\theta}($  $) + \mathcal{L}_{\theta'}($  $)$

θ' has no gradient wrt. to θ

Loss Curve

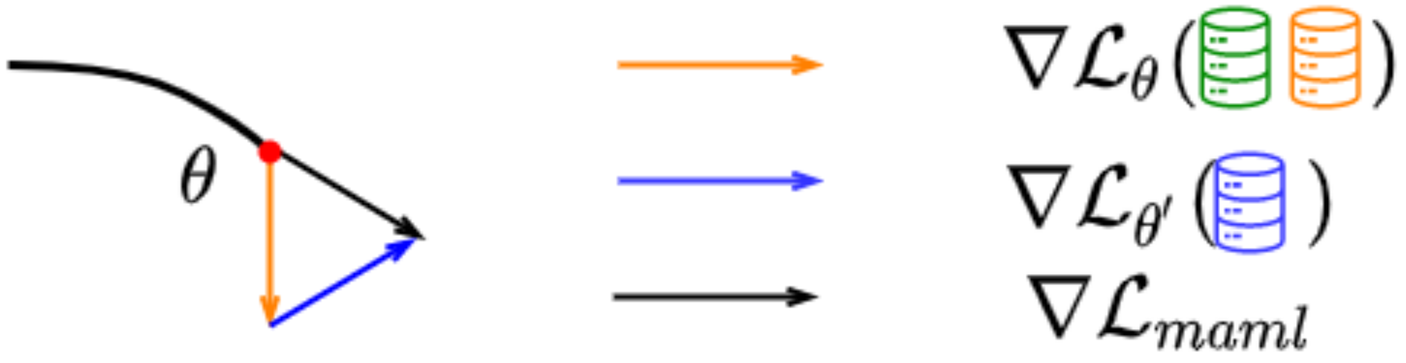


Baseline-train
Baseline-val
DG-MAML-train
DG-MAML-val
Loss
Traning Steps (10^3)
0
10
20
30
40
50
0
2
4
6
8
10

First-Order Approximation: DG-FMAML

$$\mathcal{L}_{maml}(\theta) = \mathcal{L}_{\theta}(\quad) + \mathcal{L}_{\theta'}(\quad)$$

θ' has no gradient wrt. to θ



Construct Virtual Tasks for Meta-Learning