MetaMix: Improved Meta-Learning with Interpolation-based Consistency Regularization

Yangbin Chen[1], Yun Ma[2], Tom Ko[3], Jianping Wang[1], Qing Li[2]

1. City University of Hong Kong
2. The Hong Kong Polytechnic University
3. Southern University of Science and Technology

International Conference on Pattern Recognition (ICPR) 2020

Outline

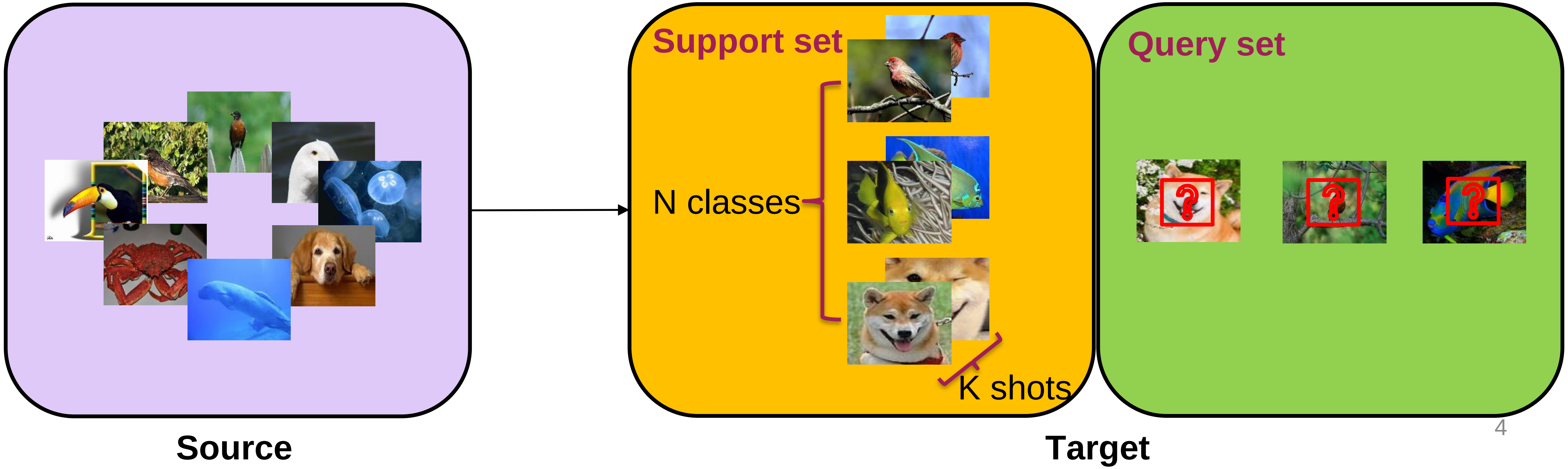
- **Background**: few-shot learning and meta-learning
- **Motivation**: to solve the meta-overfitting problem
- **Methodology**: interpolation-based consistency regularization
- **Experiment**: implementation, result, and discussion
- **Conclusion and future work**

Part I

Background

Few-shot classification

- **Few-Shot Learning (FSL) problem** is a machine learning problem that learns with limited labelled data of the target tasks by incorporating external source data, with a different distribution.
- **Few-Shot Classification** is a few-shot learning task, which is defined as **N-way, K-shot**
  - N is the number of classes in the target task
  - K is the number of labelled examples per class



Meta-Learning

- Most popular solutions of few-shot learning problems use meta-learning.
- Also known as ‘learning to learn’, aims to make a quick adaptation to new tasks with only a few examples.
- Many elegant solutions are proposed:
  - Metric-based: Matching Network, Prototypical Network, Relation Network, etc.
  - Optimization-based: Model-Agnostic Meta-Learning, Reptile, etc.
  - Model-based: Memory-Augmented Meta-Learning, Meta Networks, etc.

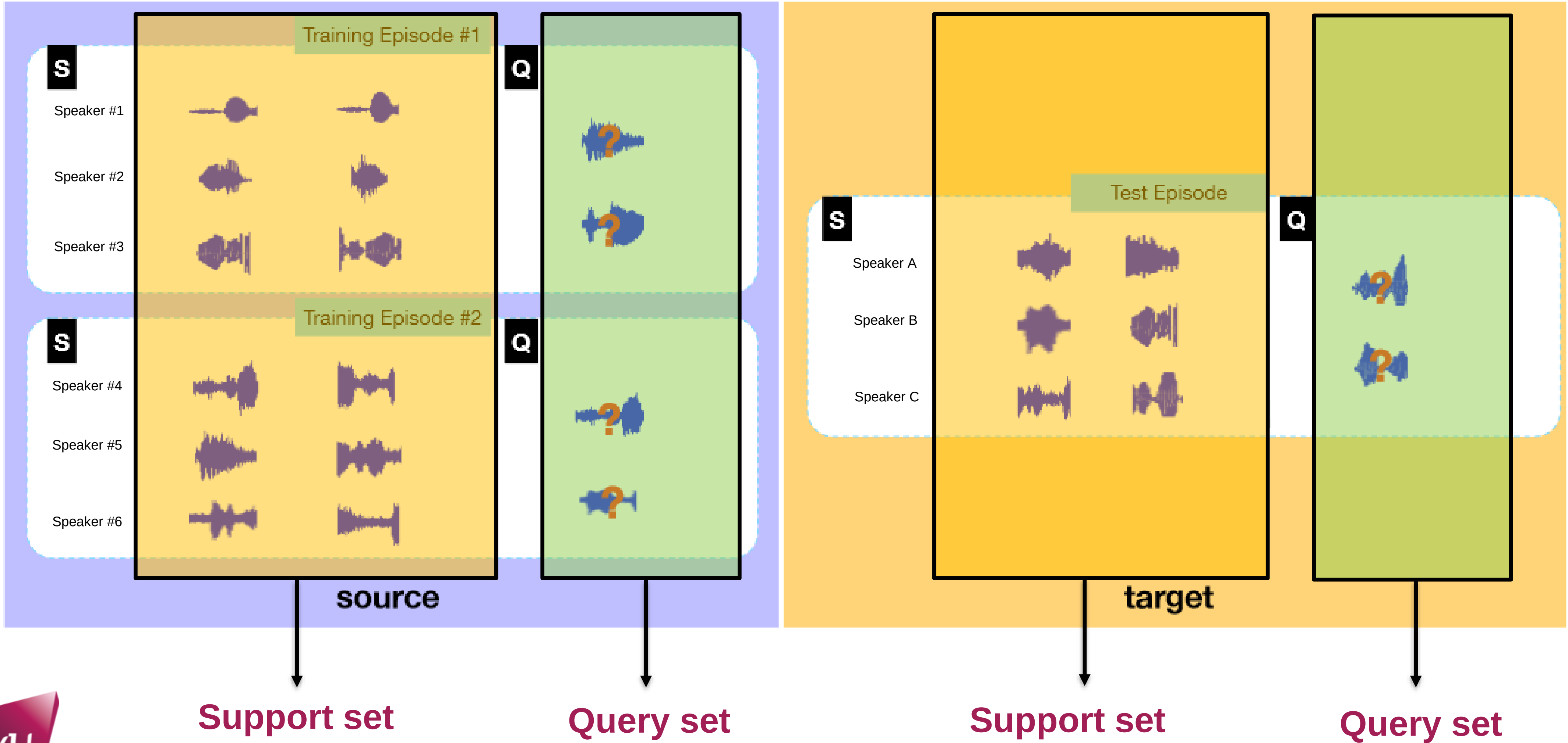
Model-Agnostic Meta-Learning (MAML)

- To train a model which can adapt to any new task using only a few labelled examples.
- The model is trained on various tasks (meta-tasks) and it treats the entire task as a training example.
- The model is forced to face different tasks so that it can get used to adapting to new tasks.

Chelsea Finn, Pieter Abbeel, Sergey Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,”in Proceedings of the 34th International Conference on Machine Learning (ICML). JMLR. 2017, pp. 1126–1135.

Episodic training in MAML

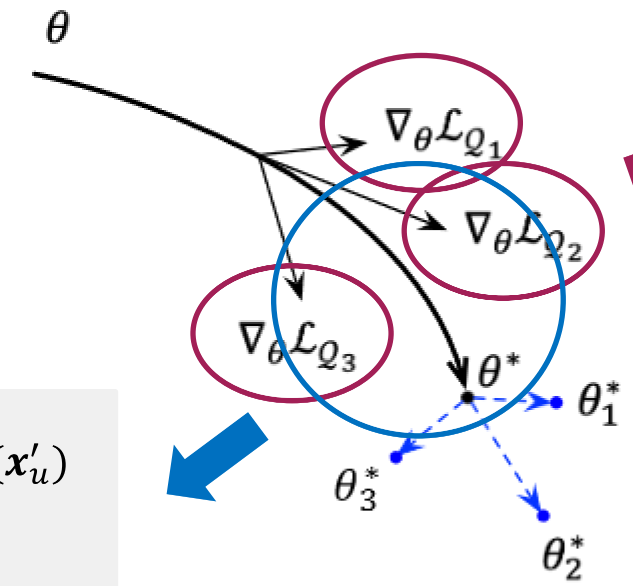
- The model is trained on various meta-tasks and it treats an entire task as a training example.



MAML – the meta-learning stage

$$\mathcal{L}_{S_i}(f_\theta) = -\sum_{(\boldsymbol{x}_j,\boldsymbol{y}_j)\in S_i} \boldsymbol{y}_j log f_\theta(\boldsymbol{x}_j)$$

$$\theta_i' = \theta - \alpha\nabla_\theta\mathcal{L}_{S_i}(f_\theta)$$ **inner loop**

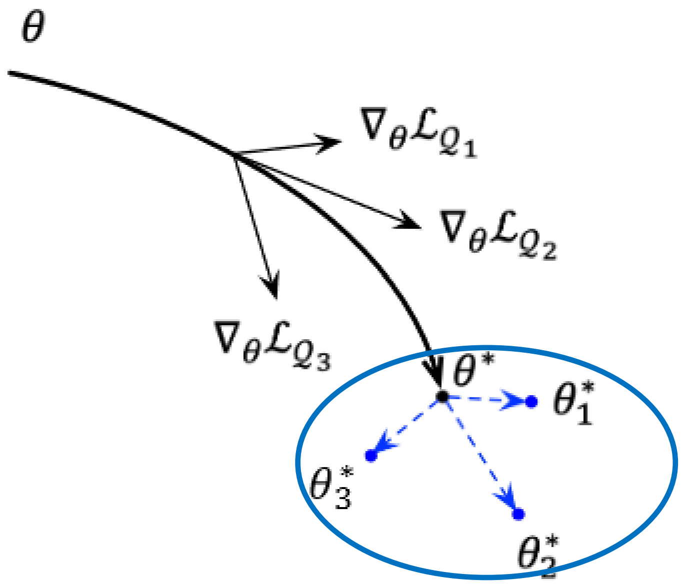


$$\mathcal{L}_{Q_i}\left(f_{\theta_i'}\right) = -\sum_{(\boldsymbol{x}_u',\boldsymbol{y}_u')\in Q_i} \boldsymbol{y}_u' log f_{\theta'}(\boldsymbol{x}_u')$$

$$\theta^* \leftarrow \theta - \beta\nabla_\theta\sum_i \mathcal{L}_{Q_i}(f_{\theta_i'})$$ **outer loop**

MAML – the fine-tuning stage

- Before evaluation, the model will be fine-tuned for a few iterations:



$$\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}_{S_i}(f_\theta)$$

Part II

Motivation

Motivation

- There exist weaknesses in current meta-learning algorithms, especially in **forming generalizable decision boundaries** (i.e., meta-overfitting).
- We aim to propose a **regularization technique** to solve the **meta-overfitting** problem.

The meta-overfitting problem

- Conventional meta-learning algorithms may face meta-overfitting problems, which form a decision boundary ***staying too close*** to the limited labelled examples in ***the few-shot tasks***.
- Empirical Risk Minimization allows large neural networks to ***memorize*** (instead of ***generalize*** from) the training data.

expected risk: $R(h) = \int \ell(h(x), y)\, dp(x, y) = \mathbb{E}[\ell(h(x), y)]$

empirical risk: $R_I(h) = \frac{1}{I} \sum_{i=1}^{I} \ell(h(x_i), y_i)$

Part III

Methodology

mixup – an interpolation-based regularization method

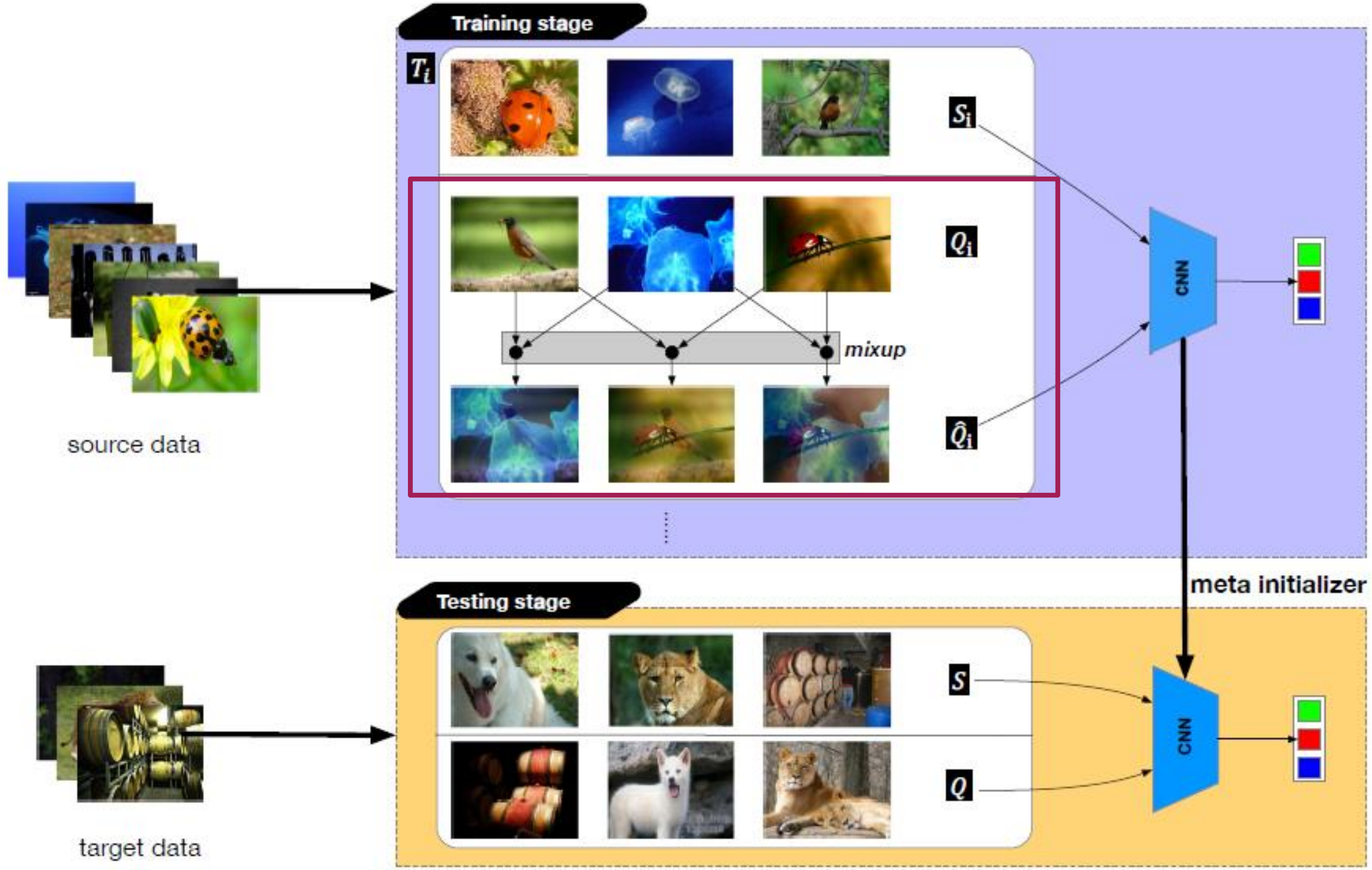
- *Mixup* [1] encourages the model to behave linearly in-between training examples, which reduces the amount of undesirable oscillations when predicting outside the training examples.
- We have adopted *mixup* in **semi-supervised learning** [2] and **unsupervised domain adaptation** [3].

$$\hat{x}_z = \lambda x_m + (1 - \lambda) x_n$$

$$\hat{y}_z = \lambda y_m + (1 - \lambda) y_n$$

[1] Zhang, H., Cisse, M., Dauphin, Y. N., & Lopez-Paz, D. mixup: Beyond Empirical Risk Minimization. In *International Conference on Learning Representations (ICLR) 2018*.
[2] Ma, Y., Mao, X., **Chen, Y.,** & Li, Q. Mixing Up Real Samples and Adversarial Samples for Semi-Supervised Learning. International Joint Conference on Neural Networks (IJCNN), IEEE, 2020.
[3] Mao, X., Ma, Y., Yang, Z**., Chen, Y.,** & Li, Q. (2019). Virtual mixup training for unsupervised domain adaptation. *arXiv preprint arXiv:1905.04215*.

MetaMix – our methodology



Algorithm 1 MetaMix with MAML

Require: $p(\mathcal{T})$: distribution over tasks
Require: $\mathcal{S}_i$: support set; $\mathcal{Q}_i$: query set
Require: α, β : learning rate
Require: $\check{\alpha}$: Beta distribution parameter
Require: $mix_\lambda(a, b) = \lambda a + (1 - \lambda)b, \lambda \sim \mathrm{B}(\check{\alpha}, \check{\alpha})$

Randomly initialize model parameters θ
while not done **do**
  Sample a batch of episodes $\mathcal{T}_i \sim p(\mathcal{T})$
  for all $\mathcal{T}_i$ **do**
    Sample a support set $\mathcal{S}_i = \{(x_j, y_j)\}_{j=1}^{J}$
    Evaluate $\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{S}_i}(f_{\boldsymbol{\theta}})$ using $\mathcal{S}_i$ and $\mathcal{L}_{\mathcal{S}_i}(f_{\boldsymbol{\theta}})$
    Compute adapted parameters with gradient descent: $\theta_i' = \theta - \alpha \cdot \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{S}_i}(f_{\boldsymbol{\theta}})$
    Sample a query set $\mathcal{Q}_i = \{(x_z, y_z)\}_{z=1}^{Z}$
    Randomly select pairs of examples $\{(x_m, y_m)\}_{m=1}^{Z}, \{(x_n, y_n)\}_{n=1}^{Z}$ from $\mathcal{Q}_i$
    $\hat{x}_z = mix_\lambda(x_m, x_n), \hat{y}_z = mix_\lambda(y_m, y_n)$
    Get new query set $\hat{\mathcal{Q}}_i = \{(\hat{x}_z, \hat{y}_z)\}_{z=1}^{Z}$
  end for
  Update $\theta \leftarrow \theta - \beta \cdot \nabla_\theta \sum_i \mathcal{L}_{\hat{\mathcal{Q}}_i}(f_{\boldsymbol{\theta}_i'})$
end while

MetaMix – our methodology

- We generate virtual examples only from the query set for two reasons:
  - The query set is responsible for optimizing the **meta-objective** across different training episodes, which is significant to the generalization of the learned initializer.
  - Virtual examples generated by interpolating examples from the query set are expected to better approximate the **real data distribution**.

Part IV

Experiment

Experimental setup

- Dataset
  - *mini*-ImageNet
    - 100 classes, 600 84 × 84 colored images per class, 64 training / 16 validation / 20 testing.
  - Caltech-UCSD Birds-200-2011 (CUB)
    - 200 classes, 11,788 84 × 84 colored images in total, 100 training / 50 validation / 50 testing.
  - Fewshot-CIFAR100 (FC100)
    - 100 classes, 600 32 × 32 colored images per class, 60 training / 20 validation / 20 testing.

Model setup

- Baselines
  - Prototypical Networks, Matching Network, Relation Network
  - MAML, First-Order MAML (FOMAML), Meta-SGD, Meta-Transfer Learning (MTL)
- Backbone model
  - Shallow CNN with 4 convolutional blocks (Conv([32, 3, 3])+ReLU+BN+MaxPooling([2, 2]))
  - ResNet-12 (in MTL)

Results

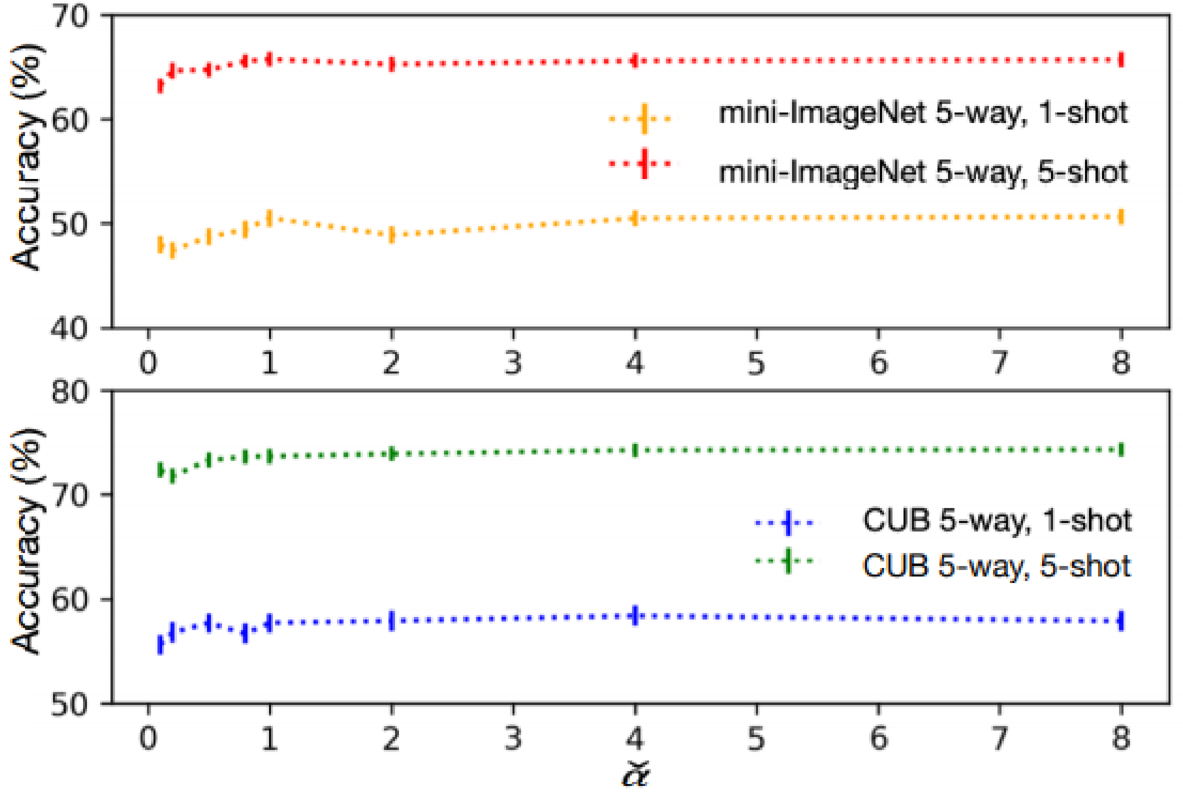
- Comparison with baselines

	mini-ImageNet		CUB		FC100	
Models	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
Matching Network	50.47 ± 0.80	64.83 ± 0.67	57.70 ± 0.87	71.42 ± 0.71	36.97 ± 0.67	49.44 ± 0.71
Prototypical Network	49.33 ± 0.82	65.71 ± 0.67	51.34 ± 0.86	67.56 ± 0.76	36.83 ± 0.69	51.21 ± 0.74
Relation Network	50.48 ± 0.80	65.39 ± 0.72	59.47 ± 0.96	73.88 ± 0.74	36.40 ± 0.69	51.35 ± 0.69
MAML	48.18 ± 0.78	63.05 ± 0.71	54.32 ± 0.91	71.37 ± 0.76	35.96 ± 0.71	48.06 ± 0.73
MetaMix+MAML	**50.51** ± **0.86**	**65.73** ± **0.72**	**57.70** ± **0.92**	**73.66** ± **0.74**	**37.09** ± **0.74**	**49.31** ± **0.72**
FOMAML	45.22 ± 0.77	60.97 ± 0.70	53.12 ± 0.93	70.90 ± 0.75	34.97 ± 0.70	47.41 ± 0.73
MetaMix+FOMAML	**47.78** ± **0.77**	**63.55** ± **0.70**	**54.81** ± **0.97**	**72.90** ± **0.74**	**36.48** ± **0.67**	**49.48** ± **0.71**
MetaSGD	49.93 ± 1.73	64.01 ± 0.90	56.19 ± 0.92	69.14 ± 0.75	36.36 ± 0.66	49.96 ± 0.72
MetaMix+MetaSGD	**50.60** ± **1.80**	**64.47** ± **0.88**	**57.64** ± **0.88**	**70.50** ± **0.70**	**37.44** ± **0.71**	**51.41** ± **0.69**
MTL	61.37 ± 0.82	78.37 ± 0.60	71.90 ± 0.86	84.68 ± 0.53	42.17 ± 0.79	56.84 ± 0.75
MetaMix+MTL	**62.74** ± **0.82**	**79.11** ± **0.58**	**73.04** ± **0.86**	**86.10** ± **0.50**	**43.58** ± **0.73**	**58.27** ± **0.73**

*Accuracy with 95% confidence intervals of **5-way, K-shot (K=1, 5)** classification tasks on **mini-ImageNet, CUB,** and **FC100** datasets.*

Results

- Analysis of hyper-parameter in Beta distribution



Effect of Beta distribution. $\breve{\alpha}$ is set to 0.1, 0.2, 0.5, 0.8, 1.0, 2.0, 4.0, 8.0.

Results

- Ablation study

	mini-ImageNet		CUB	
Set(s)	1-shot	5-shot	1-shot	5-shot
Q	**50.51 ± 0.86**	**65.73 ± 0.72**	**57.70 ± 0.92**	**73.66 ± 0.74**
S	47.87 ± 0.82	62.34 ± 0.65	54.39 ± 0.97	67.23 ± 0.74
Q+S	48.36 ± 0.81	64.06 ± 0.72	54.32 ± 0.93	70.30 ± 0.75
w/o mixup	48.18 ± 0.78	63.05 ± 0.71	54.32 ± 0.91	71.37 ± 0.76

An ablation study of doing mixup on different sets. Q denotes the query set and S denotes the support set.

Results

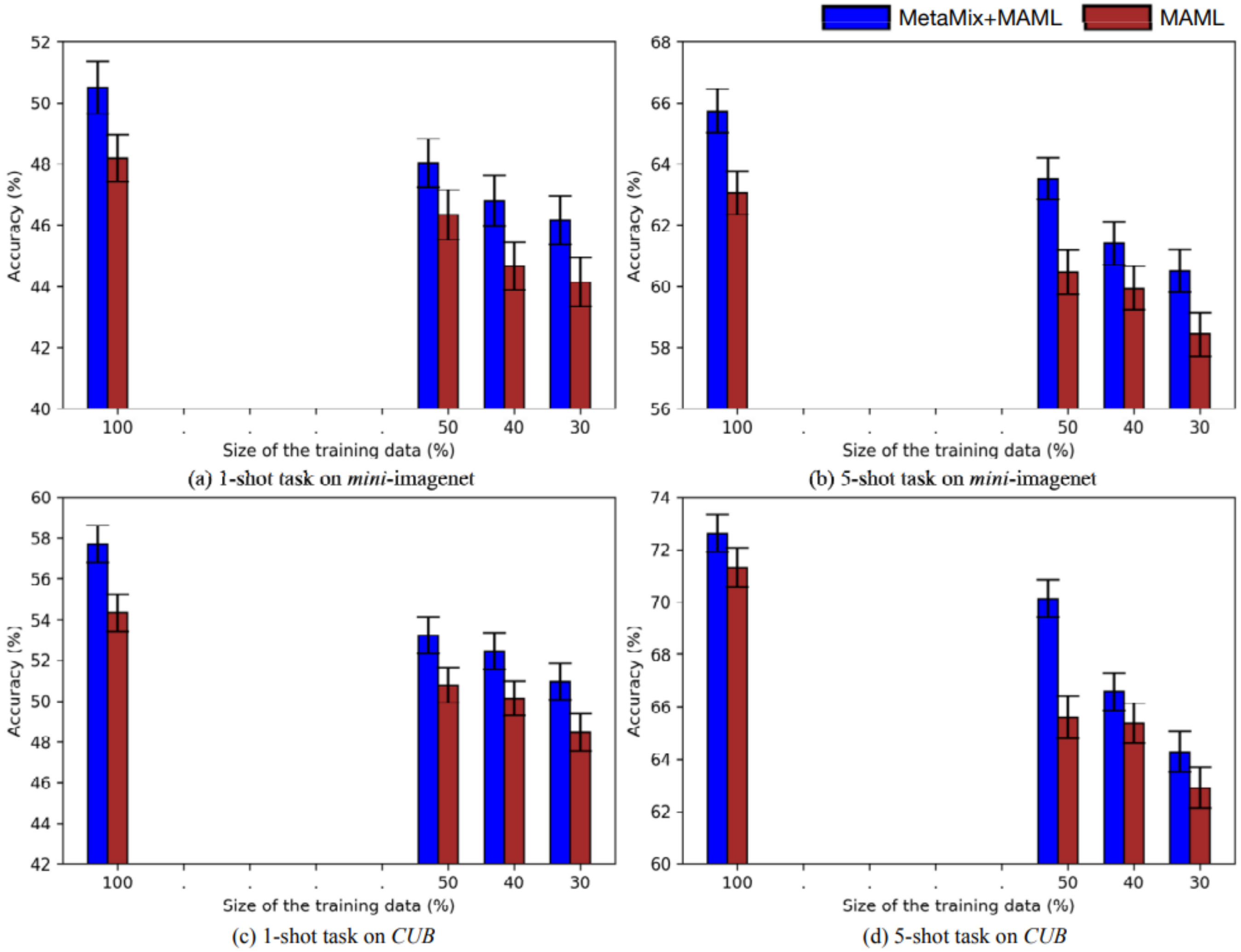
- Analysis of the effect of the size of training data

	mini-ImageNet		CUB		FC100	
Set(s)	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
MAML(100%)	48.18 ± 0.78	63.05 ± 0.71	54.32 ± 0.91	71.37 ± 0.76	35.96 ± 0.71	48.06 ± 0.73
MetaMix+MAML(100%)	**50.51 ± 0.86**	**65.73 ± 0.72**	**57.70 ± 0.92**	**73.66 ± 0.74**	**37.09 ± 0.74**	**49.31 ± 0.72**
MAML(50%)	46.34 ± 0.82	60.47 ± 0.73	50.78 ± 0.86	65.60 ± 0.81	35.38 ± 0.71	47.93 ± 0.78
MetaMix+MAML(50%)	**48.04 ± 0.79**	**63.52 ± 0.67**	**53.22 ± 0.91**	**70.13 ± 0.70**	**36.35 ± 0.74**	**48.11 ± 0.69**

*A comparison between using 100% and 50% training data; accuracy with 95% confidence intervals of **5-way, K-shot (K=1, 5)** classification tasks on **mini-ImageNet**, **CUB**, and **FC100** datasets.*

Results

- Analysis of the effect of the size of training data



A comparison among using 100%, 50%, 40%, and 30% of the training data.

Observations

- MetaMix improves the performance of all MAML-based algorithms over three datasets; meanwhile, MetaMix with MTL achieves state-of-the-art performance.
- When $\check{\alpha}$ is below 1.0, the accuracy is a little lower. When $\check{\alpha}$ is 1.0 and above, the performance maintains a good level.
- Mixing examples from only the query set performs best, compared with mixing examples from only the support set and mixing examples from both the support set and the query set.
- MetaMix performs more robust with the reduction of the size of the training data.

Part V

Conclusions

Conclusion

- We propose an improved meta-learning approach with the **interpolation-based consistency regularization** technique. It improves the performance of MAML-based algorithms.

- MetaMix achieves **state-of-the-art** results when integrated with Meta-Transfer Learning.

- MetaMix is **less sensitive to the reduction of the source training data**, compared to MAML and its variants.

Future work

- Apply MetaMix to a **broader range** of few-shot learning tasks.
- Compare **more different conditions**, under which meta-learning works, such as differences in the size of the source data, backbone models, and domains of the tasks.
- Propose **more regularization techniques** to solve the meta-overfitting problem.

Thank you!

Email: robinchen2-c@my.cityu.edu.hk
Github: https://github.com/Codelegant92