iTAML: An Incremental Task-Agnostic Meta-learning Approach

Jathushan Rajasegaran[1] Salman Khan[1] Munawar Hayat[1]
Fahad Shahbaz Khan[1,2] Mubarak Shah[3]

[1]*Inception Institute of Artificial Intelligence* [2]*CVL, Linköping University* [3]*University of Central Florida*

CVPR SEATTLE
JUNE 14-19 2020 WASHINGTON

Problem Definition

- Continual learning is essential for intelligent systems.

- Continual learning algorithms need to retain the *past knowledge* while learning new concepts on *newly revealed* data sets.

  In other words, these algorithms needs to achieve **generalization**.

- *Meta-learning* is an ideal tool for such problems.

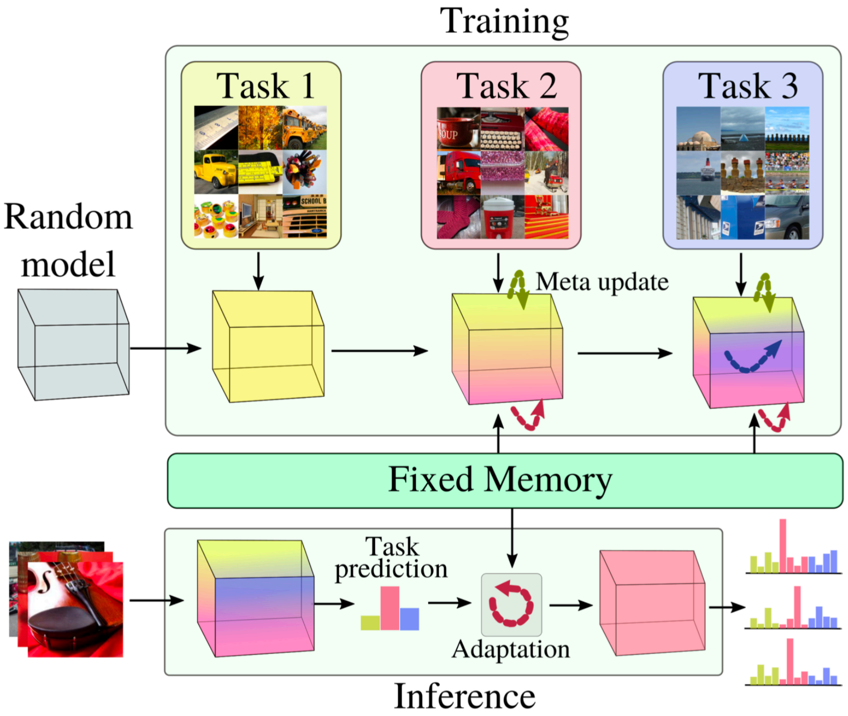
The Challenge

- Achieving generalization to new data while preserving past knowledge remains a challenge for existing *incremental learning* algorithms.

- *Meta-learning* suffers on incremental learning setting due to,
    - Out of Order Distribution (OOD).
    - Often requires fine-tuning at the end.
    - Skewed data distribution with limited memory.

iTAML tries to bridge the gap between meta-learning and incremental learning.

Incremental Task-Agnostic Meta-learning



- *The $\boxed{\text{tasks}}$ are observed sequentially*

  Each task is a set of classes

- *iTAML incrementally learns new tasks with meta-updates and tries to retain previous knowledge*

- *At inference, given a data continuum, iTAML first predicts the task and then quickly adapts to it*

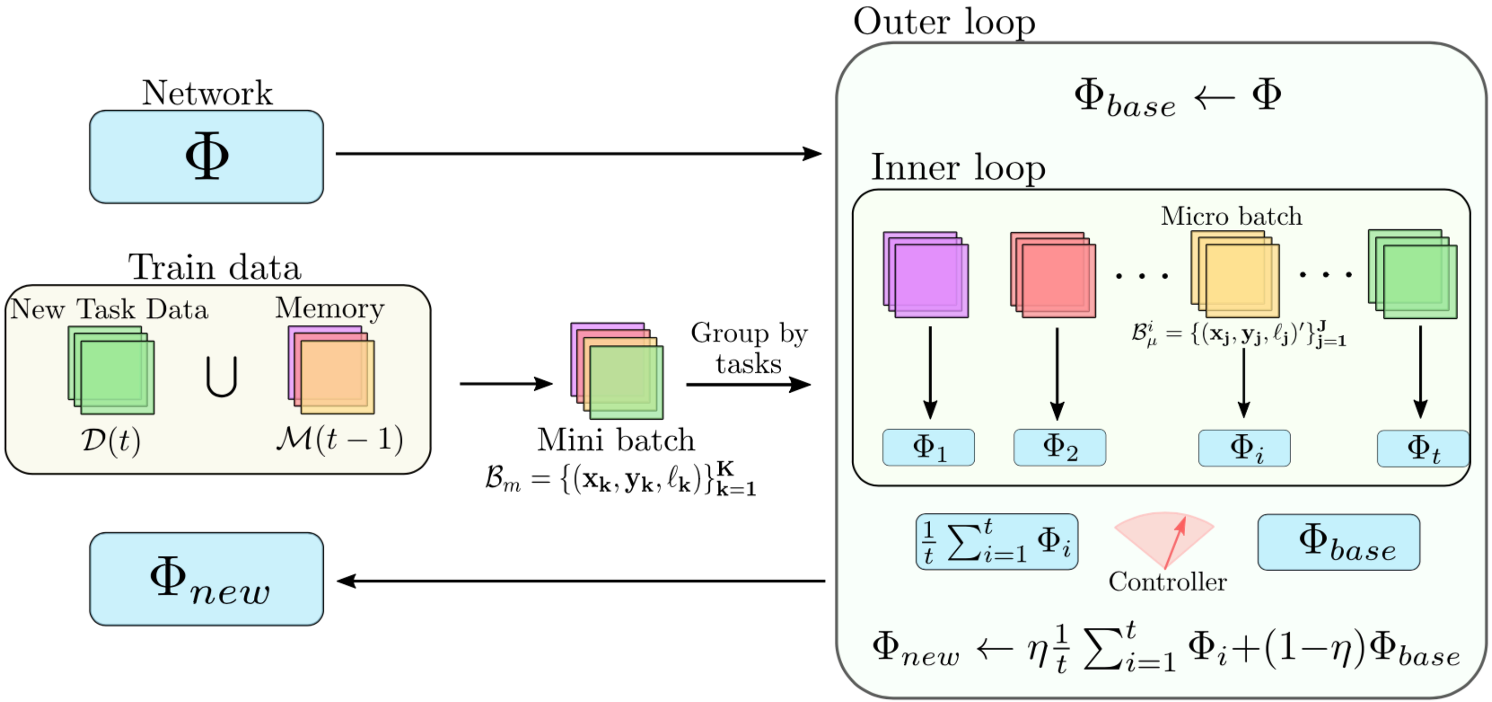
Incremental Task-Agnostic Meta-learning

- The experimental setting for iTAML:
  - Involves learning a single model which can generalize to all the tasks (*old* as well as *new*).
  - We make a weak assumption that a data continuum is available with all the samples belongs to a single task (yet the *task* is unknown).
  - Our meta-learned generic model is good enough to find the correct task.

Incremental Task-Agnostic Meta-learning

- iTAML uses the following novel learning and inference strategies:
  - A momentum based meta-update rule to avoid forgetting.
  - Disentangling the network into a generic feature extractor and task-specific classification weights.
  - A task-agnostic prediction mechanism, with two stage classification.
  - A sampling rate selection approach for data continuum.

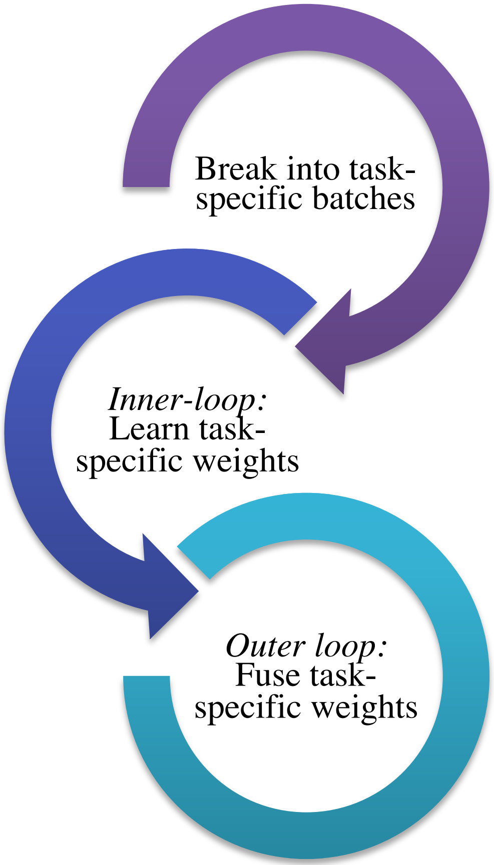
Meta Training of iTAML



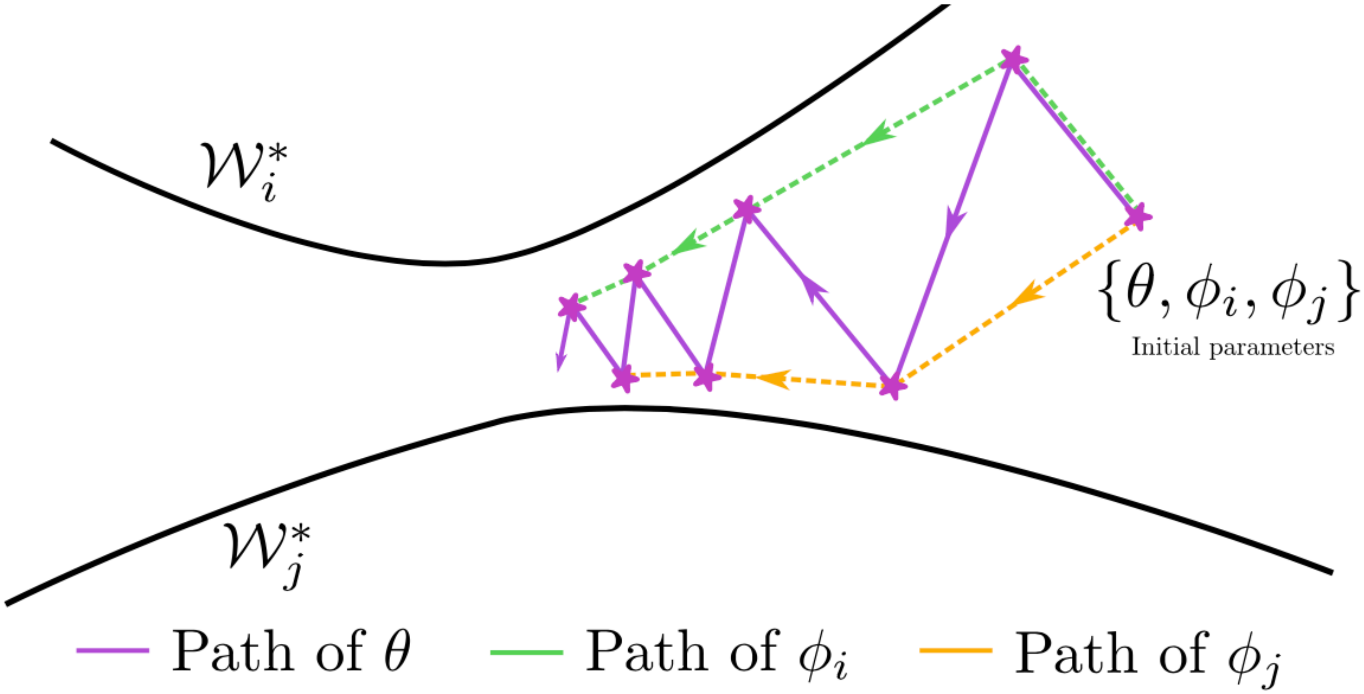
Meta Training of iTAML

- Each mini-batch is further broken into task specific micro batches.
- In the inner loop, task specific models Φ_i are trained for each seen task.
- Then, a *momentum controller* combines these task specific weights in the outer loop.

$$\Phi_{new} = \eta \frac{1}{t} \sum_{i=1}^{t} \Phi_i + (1-\eta)\Phi_{base}.$$



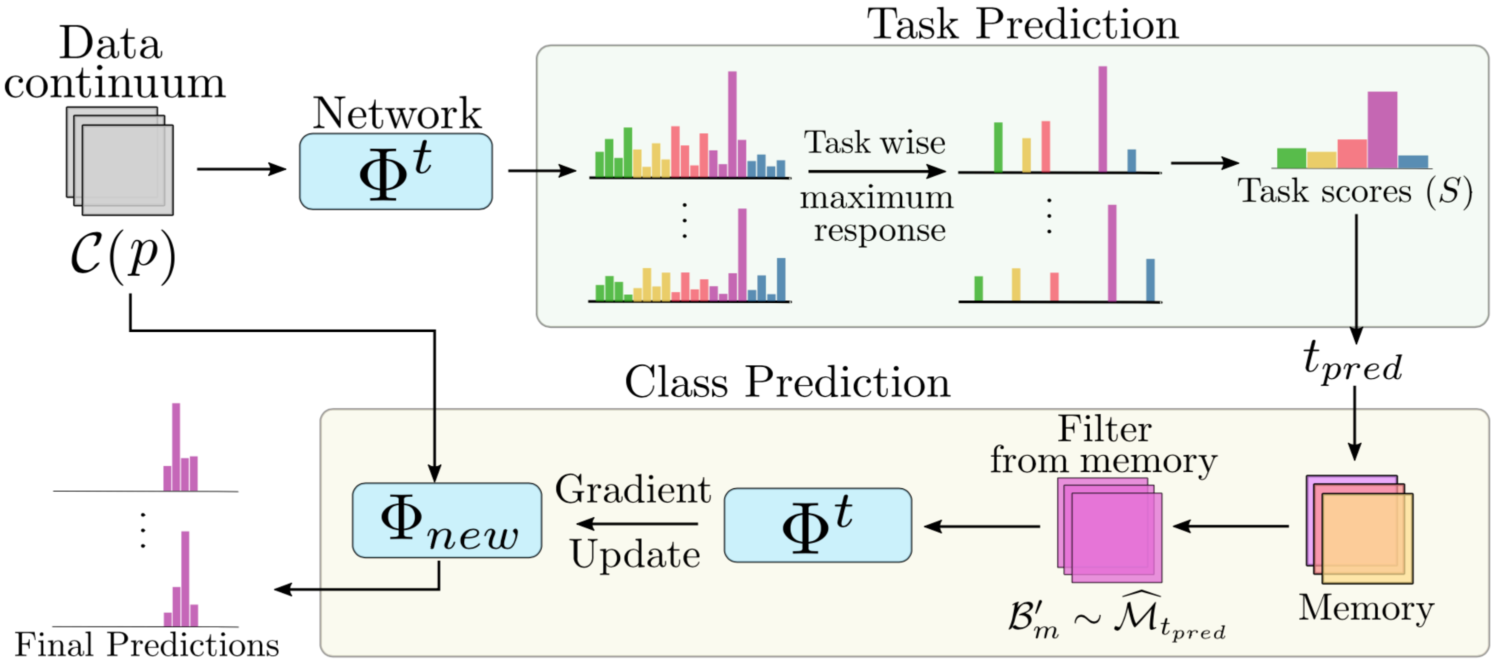
Meta Training of iTAML



Meta Training of iTAML

- Since, the feature space parameters and classification parameters are tuned separately, iTAML remains *task-agnostic*.
- Feature space parameters, are tuned for each task and combined in the outer loop, hence they remain close to optimal solution manifold of all the tasks.
- Classification parameters are tuned only for the specific task; hence they remain close to the corresponding task's optimal solution manifold.

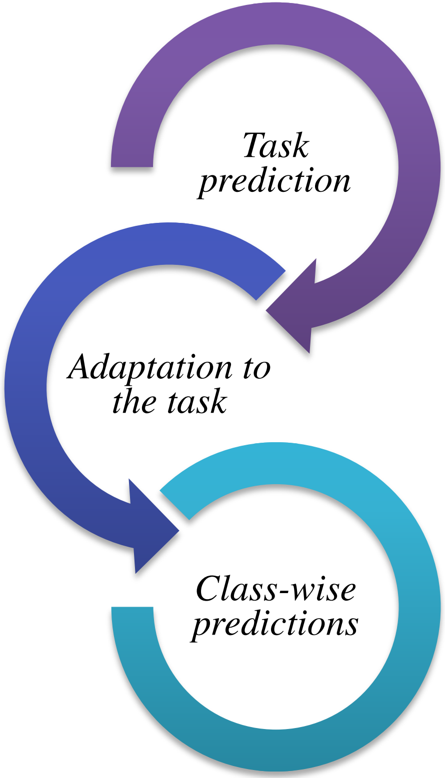
Inference of iTAML



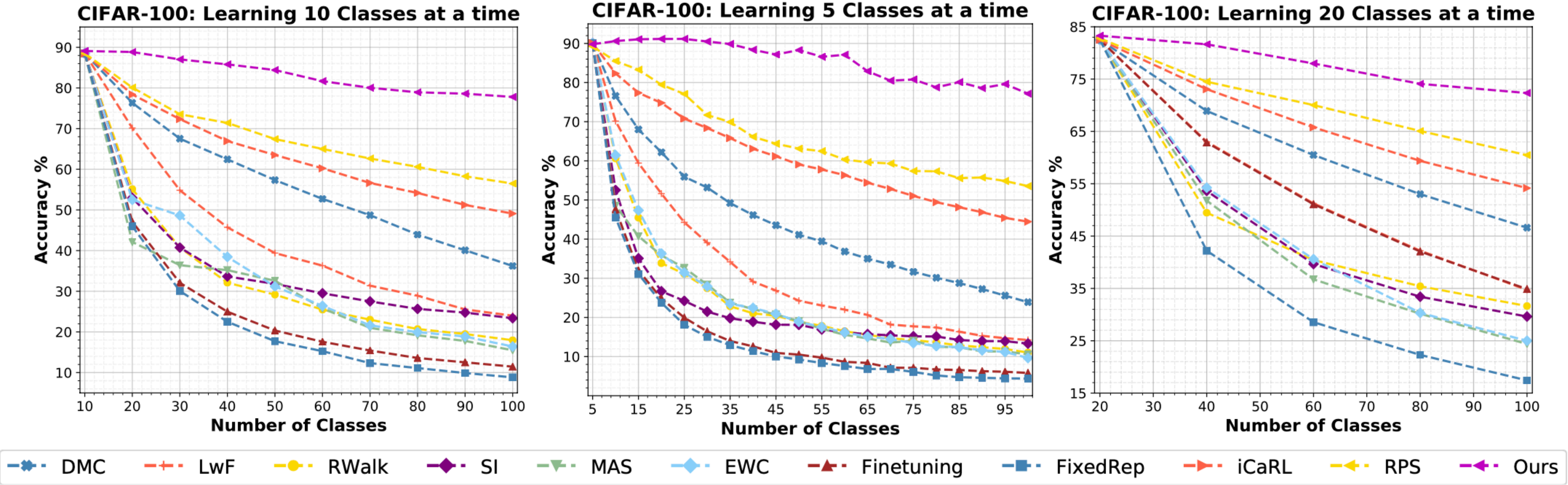
Inference of iTAML

At inference, iTAML uses a two-stage prediction.

- First, given a data continuum $\mathcal{C}(i, p)$, it predicts the task using average predictions over data samples.
- Then, it uses exemplar data to adapt for the task using a single gradient update.
- Finally, it processes the continuum and gives class-wise predictions.



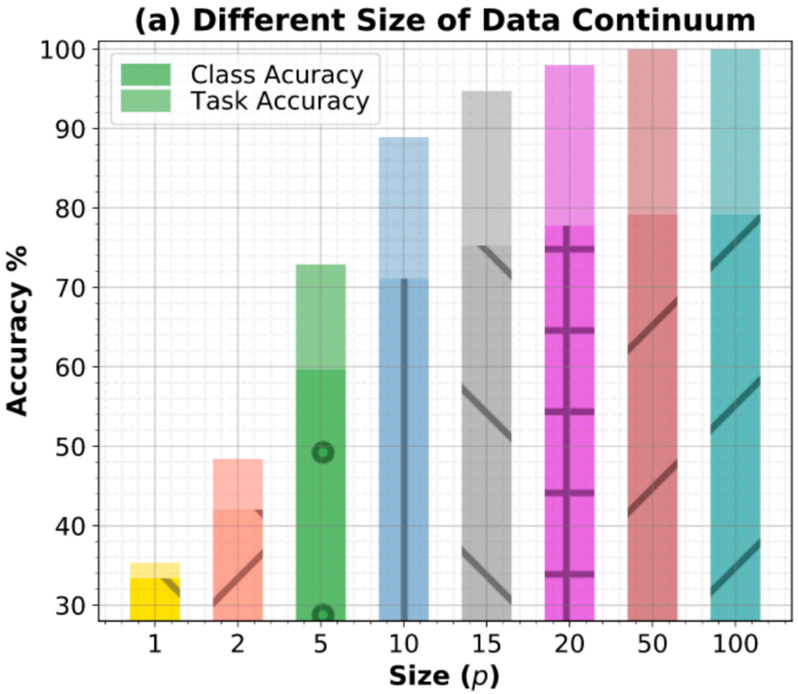
Experimental Results



CVPR SEATTLE
JUNE 14-19 2020 WASHINGTON

Experimental Results

Note that, with about 15 samples in a continuum, the model can accurately predict that correct task with 95% accuracy!

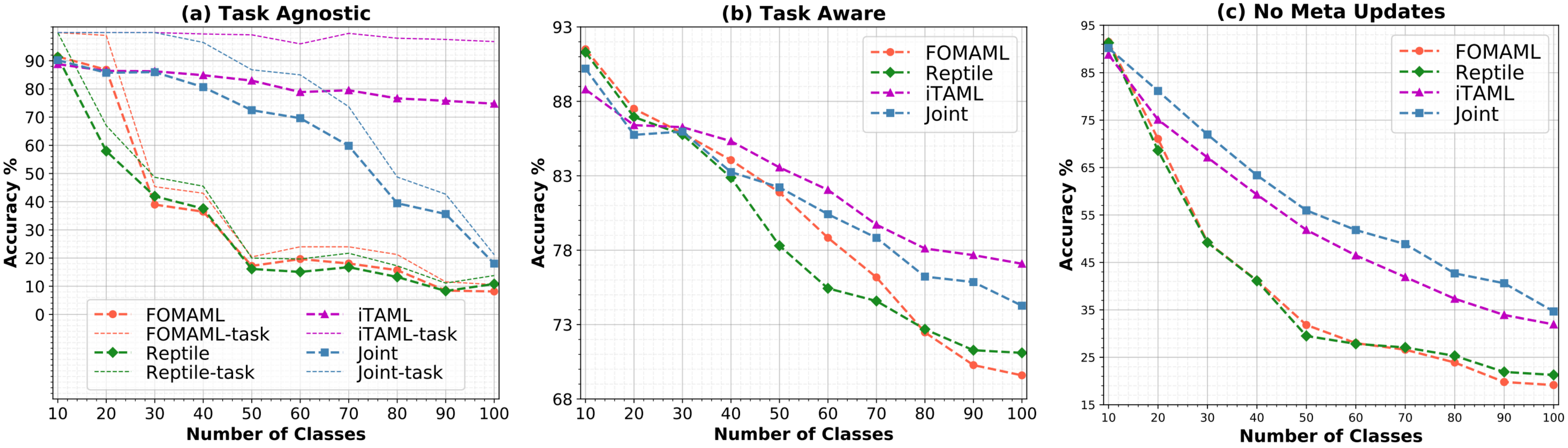


CVPR SEATTLE
JUNE 14-19 2020 WASHINGTON

Experimental Results

Datasets	Methods	1	2	3	4	5	6	7	8	9	Final
ImageNet-100/10	Finetuning	99.3	49.4	32.6	24.7	20.0	16.7	13.9	12.3	11.1	9.9
	FixedRep	99.3	88.1	73.7	62.6	55.7	50.2	42.9	41.3	39.2	35.3
	LwF(TPAMI'18)[15]	99.3	95.2	85.9	73.9	63.7	54.8	50.1	44.5	40.7	36.7
	iCaRL(CVPR'17)[23]	99.3	97.2	93.5	91.0	87.5	82.1	77.1	72.8	67.1	63.5
	RPSnet(NeurIPS'19)[22]	100.0	97.4	94.3	92.7	89.4	86.6	83.9	82.4	79.4	74.1
	Ours	99.4	96.4	94.4	93.0	92.4	90.6	89.9	90.3	90.3	**89.8**$_{+15.7}$
ImageNet-1K/10	Finetuning	90.2	43.1	27.9	18.9	15.6	14.0	11.7	10.0	8.9	8.2
	FixedRep	90.1	76.1	66.9	58.8	52.9	48.9	46.1	43.1	41.2	38.5
	LwF(TPAMI'18)[15]	90.2	77.6	63.6	51.6	42.8	35.5	31.5	28.4	26.1	24.2
	iCaRL(CVPR'17)[23]	90.1	82.8	76.1	69.8	63.3	57.2	53.5	49.8	46.7	44.1
	Ours	91.5	89.0	85.7	84.0	80.1	76.7	70.2	71.0	67.9	**63.2**$_{+19.1}$
MS-Celeb-10K/10	iCaRL(CVPR'17)[23]	94.2	93.7	90.8	86.5	80.8	77.2	74.9	71.1	68.5	65.5
	RPSnet(NeurIPS'19)[22]	92.8	92.0	92.3	90.8	86.3	83.6	80.0	76.4	71.8	65.0
	BiC(CVPR'19)[30]	95.7	96.5	96.5	95.7	95.1	94.2	93.2	91.7	90.0	87.6
	Ours	94.0	95.6	96.0	95.8	95.5	95.4	95.2	95.1	95.0	**95.0**$_{+7.4}$

Experimental Results



Thank You!

https://github.com/brjathu/iTAML