Automated Relational Meta Learning

Accepted in ICLR 2020

Kyeong Ryeol, Go
M.S. Candidate of OSI Lab

Overview

- Motivation
  - Globally shared meta-learners may not be sufficient to handle diverse tasks

- Goal
  - Customizing the globally shared meta-learners in a task specific manner

$$\text{MAML} : \theta_0 \rightarrow \theta_i = \theta_0 - \alpha \cdot \nabla_{\theta_0} \mathcal{L}(\theta_0)$$

$$\text{ARML} : \theta_0 \underset{\text{Graph}}{\rightarrow} \theta_{0i} \rightarrow \theta_i = \theta_{0i} - \alpha \cdot \nabla_{\theta_{0i}} \mathcal{L}(\theta_{0i})$$

Content

- Graph structure
  - Prototype based relational graph
  - Meta knowledge graph
  - Super graph
- ARML Algorithm
- Experiment
- Conclusion

Prototype based relational graph

- Role
  - Capture the underlying relationship behind samples

- Notation : $R_i = (C_{R_i}, A_{R_i})$
  - Vertices ($C_{R_i} \in R^{K \times d}$) : prototypes of different classes
    1. Classification : $c_i^k = \frac{1}{N_k^{tr}} \sum_{j=1}^{N_k^{tr}} \mathcal{E}(x_j)$
    2. Regression : $c_i^k = P_i[k]\mathcal{F}(X) \ \ where \ \ P_i = Softmax(W_p \mathcal{E}(X)^T + b_p)$

  - Edges ($A_{R_i} \in R^{K \times K}$) : Similarity between prototypes
    - $A_{R_i}(c_i^j, c_i^m) = \sigma\left(W_r\left(\frac{\left|c_i^j - c_i^m\right|}{\gamma_r}\right) + b_r\right)$

$where \ \ i \in [1, I], \ \ j, k, m \in [1, K], \ \ I = \#\ of\ tasks, \ \ K = \#\ of\ classes$

Meta knowledge graph

- Role
  - Organize and distill the knowledge from the historical learning process
  - Efficiently and automatically identify the relational knowledge from previous tasks

- Notation : $\mathcal{G} = (H_{\mathcal{G}}, A_{\mathcal{G}})$
  - Vertices ($H_{\mathcal{G}} \in R^{G \times d}$) : meta knowledge
    - Randomly initialized, learnable parameter
  - Edges ($A_{\mathcal{G}} \in R^{G \times G}$) : Similarity between meta knowledge
    - $A_{\mathcal{G}}(h^j, h^m) = \sigma\left(W_o\left(\frac{|h^j - h^m|}{\gamma_o}\right) + b_o\right)$

  $where\ \ j, m \in [1, G],\ \ G = \#\ of\ meta\ knowledge$

Super graph

- Role
  - For each task i, connect R_i with $\mathcal{G}$
  - Allow the training of one graph to facilitate the training of the other
  - Propagate the most relevant knowledge from $\mathcal{G}$ to R_i through GCN
    - $H_i^{(l+1)} = MP\left(A_i, H_i^{(l)}; W^{(l)}\right)\ where\ H_i^{(0)} = H_i\ and\ H_i^{(L)}[:K] = \widehat{C_{R_i}}$
- Notation : $\mathcal{S}_i = (H_i, A_i)$
  - Vertices ($H_i = \left(C_{R_i} || \mathcal{G}\right) \in R^{(K+G)\times d}$) : vertices from both R_i and $\mathcal{G}$
  - Edges ($A_i = \begin{bmatrix} A_{R_i} & A_{\mathcal{S}} \\ A_{\mathcal{S}}^T & A_{\mathcal{G}} \end{bmatrix} \in R^{(K+G)\times(K+G)}$) : edges from R_i and $\mathcal{G}$ + edges b/t R_i and $\mathcal{G}$
    - $A_{\mathcal{S}}\left(c_i^j, h^m\right) = exp\left(-\left\|\frac{c_i^j - h^m}{\gamma_s}\right\|_2^2 / 2\right) / \sum_{m'=1}^{G} exp\left(-\left\|\frac{c_i^j - h^{m'}}{\gamma_s}\right\|_2^2 / 2\right)$

$where\ i \in [1, I],\ j \in [1, K],\ m \in [1, G]$

$I = \#\ of\ tasks,\ K = \#\ of\ classes,\ G = \#\ of\ meta\ knowledge$

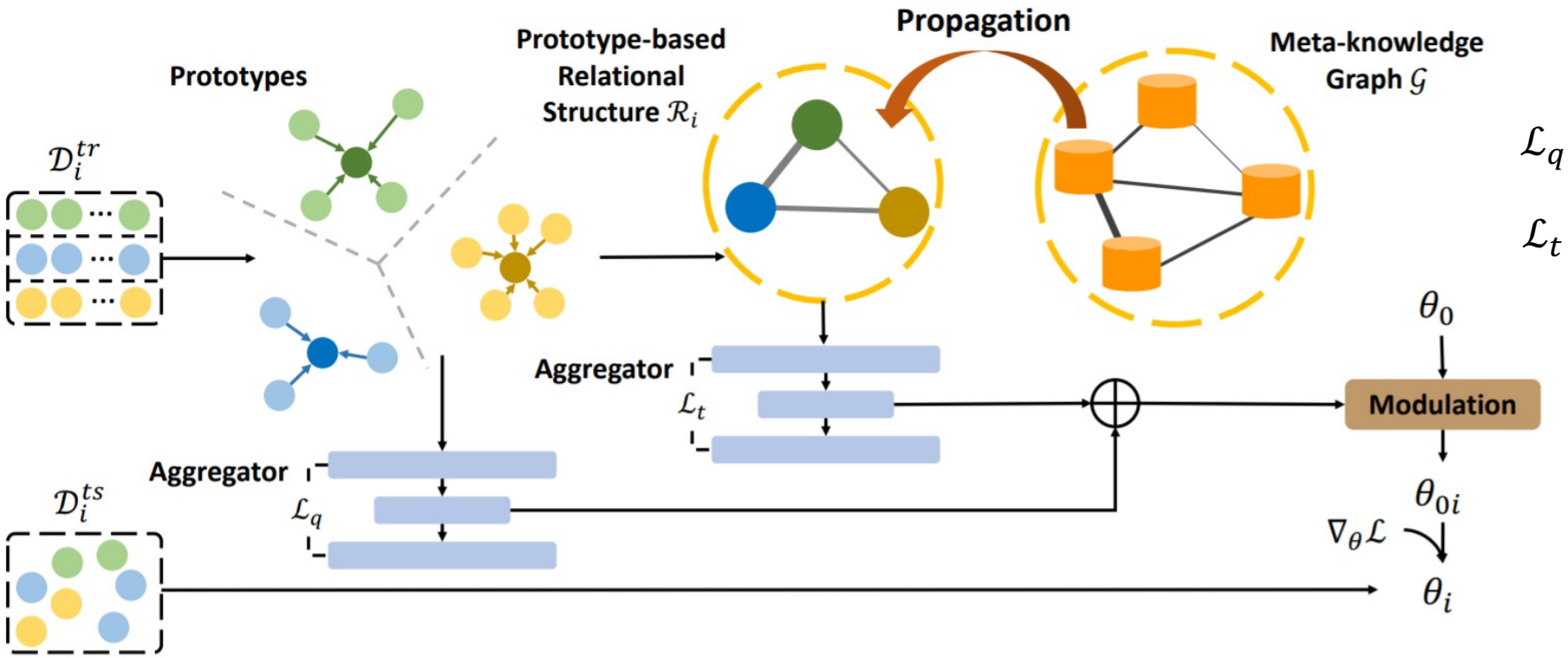
Modulation function ($\theta_0 \rightarrow \theta_{0i}$)

- Role
  - Incorporate the task-specific information to globally shared meta-learner
  - Provide customized initialization for each task utilizing C_{R_i} and $\widehat{C_{R_i}}$

- Notation
  - θ_0 : Globally shared initial parameter
  - $q_i = MeanPool\left(AG_{enc}^{q}\left(C_{R_i}\right)\right)$: task-specific dense representation of C_{R_i}
  - $t_i = MeanPool\left(AG_{enc}^{t}\left(\widehat{C_{R_i}}\right)\right)$: task-specific dense representation of $\widehat{C_{R_i}}$

  * Autoencoder-based dense representation is for the well-discriminated task representation which may be hard to learn by merely utilizing the loss signal from D_i^{ts}

- Modulation
  - $\theta_{0i} = \sigma\left(W_g(t_i||q_i) + b_g\right) \circ \theta_0$



$$\mathcal{L}_q = \left\| C_{R_i} - AG_{dec}^{q}\left(AG_{enc}^{q}(C_{R_i})\right)\right\|_F^2$$

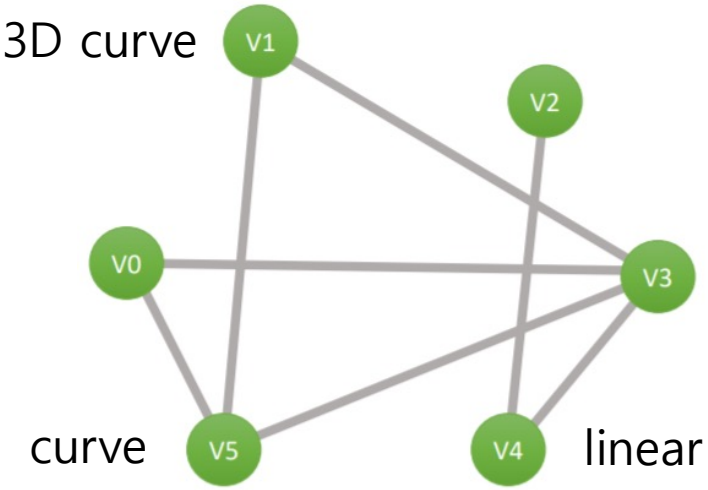
$$\mathcal{L}_t = \left\| \widehat{C_{R_i}} - AG_{dec}^{t}\left(AG_{enc}^{t}(\widehat{C_{R_i}})\right)\right\|_F^2$$

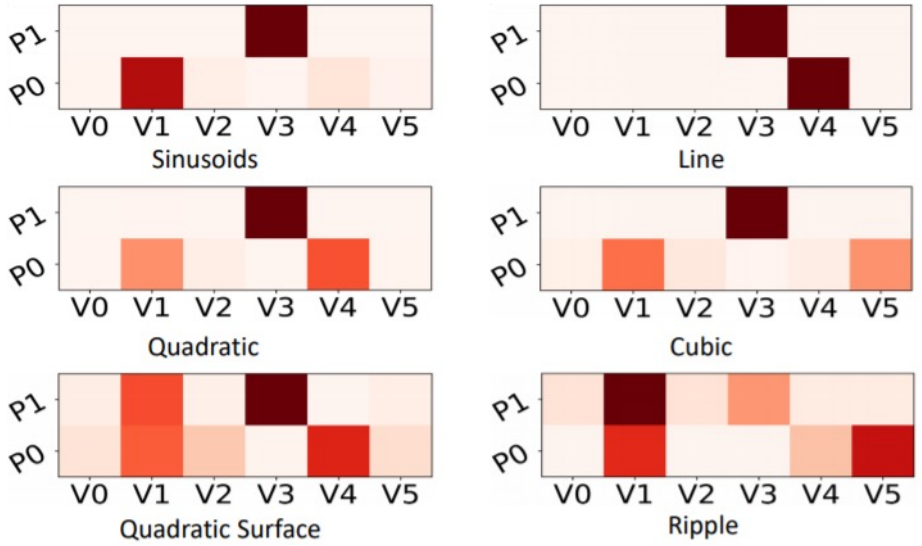
1. For each task i
   1. Construct the prototype-based relational graph $R_i = (C_{R_i}, A_{R_i})$
   2. Compute the super graph $S_i = (H_i, A_i)$
   3. Apply GNN on super graph to get enriched representation $\widehat{C_{R_i}}$
   4. Aggregate and compute dense representation of C_{R_i} and $\widehat{C_{R_i}}$ using the autoencoders
   5. Modulate the globally shared initialization to derive task-specific initial parameter θ_{0i}
   6. Apply adaptation process $\theta_i = \theta_{0i} - \alpha \nabla_{\theta_{oi}} \mathcal{L}(\theta_{0i}, \mathcal{D}_i^{tr})$
2. Apply meta-update $\phi \leftarrow \phi - \beta \nabla_\phi \sum_{i=1}^{I} [\mathcal{L}(\theta_i, \mathcal{D}_i^{ts}) + \mu_1 \mathcal{L}_t + \mu_2 \mathcal{L}_q]$

Experiment – 2D regression

- 2 prototypes, 6 meta knowledges

- $x \sim U[0,5], y \sim U[0,5]$
- Dataset
  - $z(x,y) = a_s \sin(w_s x + b_s)$
  - $z(x,y) = a_l x + b_l$
  - $z(x,y) = a_q x^2 + b_q x + c_q$
  - $z(x,y) = a_c x^3 + b_c x^2 + c_c x + d_c$
  - $z(x,y) = a_{qs} x^2 + b_{qs} y^2$
  - $z(x,y) = \sin\left(-a_r (x^2 + y^2)\right) + b_r$

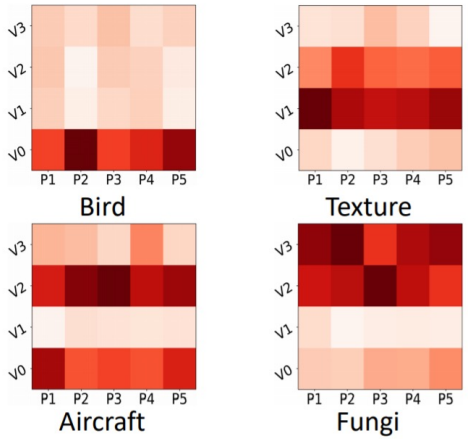


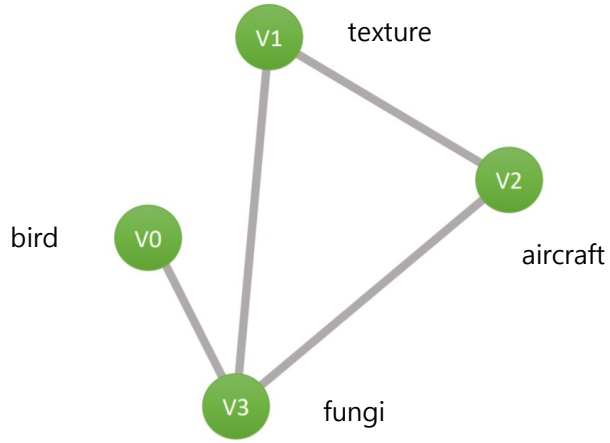


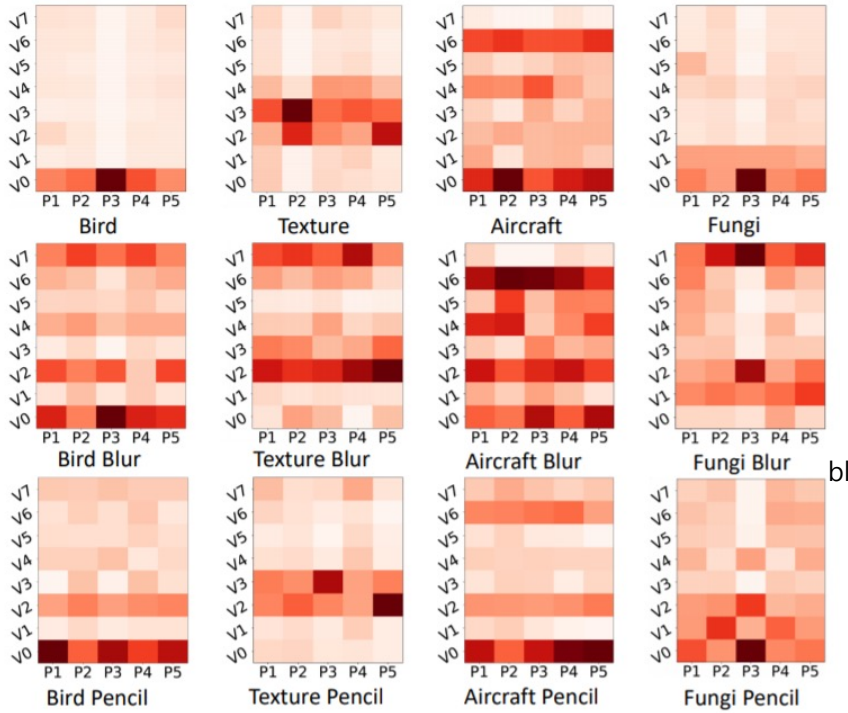
Model	MAML	Meta-SGD	BMAML	MT-Net	MUMOMAML	HSML	ARML
10-shot	2.29 ± 0.16	2.91 ± 0.23	1.65 ± 0.10	1.76 ± 0.12	0.52 ± 0.04	0.49 ± 0.04	$\mathbf{0.44 \pm 0.03}$

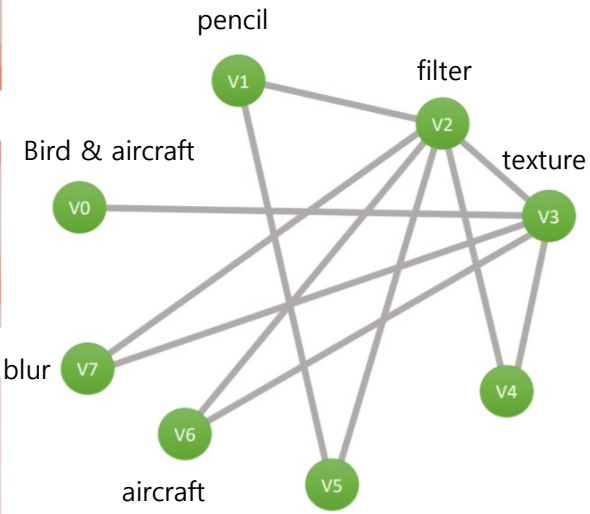
Experiment – classification

- 5 prototypes, 4 or 8 meta knowledges
- Dataset
  - Plain-Multi
    - CUB-200-2011
    - Describable Textures Dataset
    - FGVC of Aircraft
    - FGVCx-Fungi
  - Art-Multi
    - Plain-Multi
    - Plain-Multi + Blur filter
    - Plain-Multi + Pencil filter









(a) : Plain Image

(b) : with blur filter

(c) : with pencil filter

Experiment – quantitative analysis

Plain-Multi

Settings	Algorithms	Data: Bird	Data: Texture	Data: Aircraft	Data: Fungi
5-way 1-shot	VERSA	$53.40 \pm 1.41\%$	$30.43 \pm 1.30\%$	$50.60 \pm 1.34\%$	$40.40 \pm 1.40\%$
	ProtoNet	$54.11 \pm 1.38\%$	$32.52 \pm 1.28\%$	$50.63 \pm 1.35\%$	$41.05 \pm 1.37\%$
	TapNet	$54.90 \pm 1.34\%$	$32.44 \pm 1.23\%$	$51.22 \pm 1.34\%$	$42.88 \pm 1.35\%$
	TADAM	$56.58 \pm 1.34\%$	$33.34 \pm 1.27\%$	$53.24 \pm 1.33\%$	$43.06 \pm 1.33\%$
	MAML	$53.94 \pm 1.45\%$	$31.66 \pm 1.31\%$	$51.37 \pm 1.38\%$	$42.12 \pm 1.36\%$
	MetaSGD	$55.58 \pm 1.43\%$	$32.38 \pm 1.32\%$	$52.99 \pm 1.36\%$	$41.74 \pm 1.34\%$
	BMAML	$54.89 \pm 1.48\%$	$32.53 \pm 1.33\%$	$53.63 \pm 1.37\%$	$42.50 \pm 1.33\%$
	MT-Net	$58.72 \pm 1.43\%$	$32.80 \pm 1.35\%$	$47.72 \pm 1.46\%$	$43.11 \pm 1.42\%$
	MUMOMAML	$56.82 \pm 1.49\%$	$33.81 \pm 1.36\%$	$53.14 \pm 1.39\%$	$42.22 \pm 1.40\%$
	HSML	$60.98 \pm 1.50\%$	$35.01 \pm 1.36\%$	$57.38 \pm 1.40\%$	$44.02 \pm 1.39\%$
	ARML	$\mathbf{62.33 \pm 1.47\%}$	$\mathbf{35.65 \pm 1.40\%}$	$\mathbf{58.56 \pm 1.41\%}$	$\mathbf{44.82 \pm 1.38\%}$
5-way 5-shot	VERSA	$65.86 \pm 0.73\%$	$37.46 \pm 0.65\%$	$62.81 \pm 0.66\%$	$48.03 \pm 0.78\%$
	ProtoNet	$68.67 \pm 0.72\%$	$45.21 \pm 0.67\%$	$65.29 \pm 0.68\%$	$51.27 \pm 0.81\%$
	TapNet	$69.07 \pm 0.74\%$	$45.54 \pm 0.68\%$	$67.16 \pm 0.66\%$	$51.08 \pm 0.80\%$
	TADAM	$69.13 \pm 0.75\%$	$45.78 \pm 0.65\%$	$69.87 \pm 0.66\%$	$53.15 \pm 0.82\%$
	MAML	$68.52 \pm 0.79\%$	$44.56 \pm 0.68\%$	$66.18 \pm 0.71\%$	$51.85 \pm 0.85\%$
	MetaSGD	$67.87 \pm 0.74\%$	$45.49 \pm 0.68\%$	$66.84 \pm 0.70\%$	$52.51 \pm 0.81\%$
	BMAML	$69.01 \pm 0.74\%$	$46.06 \pm 0.69\%$	$65.74 \pm 0.67\%$	$52.43 \pm 0.84\%$
	MT-Net	$69.22 \pm 0.75\%$	$46.57 \pm 0.70\%$	$63.03 \pm 0.69\%$	$53.49 \pm 0.83\%$
	MUMOMAML	$70.49 \pm 0.76\%$	$45.89 \pm 0.69\%$	$67.31 \pm 0.68\%$	$53.96 \pm 0.82\%$
	HSML	$71.68 \pm 0.73\%$	$48.08 \pm 0.69\%$	$73.49 \pm 0.68\%$	$56.32 \pm 0.80\%$
	ARML	$\mathbf{73.34 \pm 0.70\%}$	$\mathbf{49.67 \pm 0.67\%}$	$\mathbf{74.88 \pm 0.64\%}$	$\mathbf{57.55 \pm 0.82\%}$

Experiment – quantitative analysis

Art-Multi

Settings	Algorithms	B Plain	B Blur	B Pencil	T Plain	T Blur	T Pencil	A Plain	A Blur	A Pencil	F Plain	F Blur	F Pencil
5-way 1-shot	VERSA	52.46%	51.65%	48.80%	30.03%	29.10%	28.74%	51.03%	47.89%	41.07%	42.13%	39.26%	36.20%
	ProtoNet	53.67%	50.98%	46.66%	31.37%	29.08%	28.48%	45.54%	43.94%	35.49%	37.71%	38.00%	34.36%
	TapNet	53.30%	51.14%	47.76%	31.56%	29.48%	29.08%	46.18%	44.50%	36.66%	37.54%	39.50%	35.46%
	TADAM	54.76%	52.18%	48.85%	32.03%	29.90%	30.82%	50.42%	47.59%	40.17%	41.73%	40.09%	36.27%
	MAML	55.27%	52.62%	48.58%	30.57%	28.65%	28.39%	45.59%	42.24%	34.52%	39.37%	38.58%	35.38%
	MetaSGD	55.23%	53.08%	48.18%	29.28%	28.70%	28.38%	51.24%	47.29%	35.98%	41.08%	40.38%	36.30%
	BMAML	56.71%	52.87%	47.83%	31.02%	29.11%	29.69%	46.83%	42.68%	36.08%	40.09%	39.66%	35.51%
	MT-Net	56.99%	54.21%	50.25%	32.13%	29.63%	29.23%	43.64%	40.08%	33.73%	43.02%	42.64%	37.96%
	MUMOMAML	57.73%	53.18%	50.96%	31.88%	29.72%	29.90%	49.95%	43.36%	39.61%	42.97%	40.08%	36.52%
	HSML	58.15%	53.20%	51.09%	32.01%	30.21%	30.17%	49.98%	45.79%	40.87%	42.58%	41.29%	37.01%
	ARML	**59.67%**	**54.89%**	**52.97%**	**32.31%**	**30.77%**	**31.51%**	**51.99%**	**47.92%**	**41.93%**	**44.69%**	**42.13%**	**38.36%**
5-way 5-shot	VERSA	66.28%	65.12%	60.76%	38.85%	35.49%	33.83%	64.82%	62.73%	53.60%	51.18%	50.30%	43.54%
	ProtoNet	70.42%	67.90%	61.82%	44.78%	38.43%	38.40%	65.84%	63.41%	54.08%	51.45%	50.56%	46.33%
	TapNet	68.60%	68.03%	62.69%	43.41%	37.86%	38.60%	65.16%	64.29%	54.73%	53.92%	50.66%	46.69%
	TADAM	70.08%	69.05%	65.45%	44.93%	**41.80%**	40.18%	70.35%	68.56%	59.09%	56.04%	54.04%	47.85%
	MAML	71.51%	68.65%	63.93%	42.96%	39.59%	38.87%	64.68%	62.54%	49.20%	54.08%	52.02%	46.39%
	MetaSGD	71.31%	68.73%	64.33%	41.89%	37.79%	37.91%	64.88%	63.36%	52.31%	53.18%	52.26%	46.43%
	BMAML	71.66%	68.51%	64.99%	43.18%	39.83%	39.76%	66.57%	63.33%	51.91%	53.96%	53.18%	48.21%
	MT-Net	71.18%	69.29%	68.28%	43.23%	39.42%	39.20%	63.39%	58.29%	46.12%	54.01%	51.70%	47.02%
	MUMOMAML	71.57%	70.50%	64.57%	44.57%	40.31%	40.07%	63.36%	61.55%	52.17%	54.89%	52.82%	47.79%
	HSML	71.75%	69.31%	65.62%	44.68%	40.13%	41.33%	70.12%	67.63%	59.40%	55.97%	54.60%	49.40%
	ARML	**73.05%**	**71.31%**	**67.14%**	**45.32%**	40.15%	**41.98%**	**71.89%**	**68.59%**	**61.41%**	**56.83%**	**54.87%**	**50.53%**

Conclusion

- Goal
  - Effective meta-learning for handling heterogeneous task
- Contribution
  - Automatically construct the meta-knowledge graph
  - Well capture the relationship among tasks and thus improve interpretability
  - Empirically outperform the state-of-the-art meta-learning algorithms
- Proposed future work
  - Extend ARML to the continual learning scenario
  - Further investigate the semantical meaning of meta knowledge graph
  - Further customize the feature space and the label space

Thank you for your attention