ACL 2020

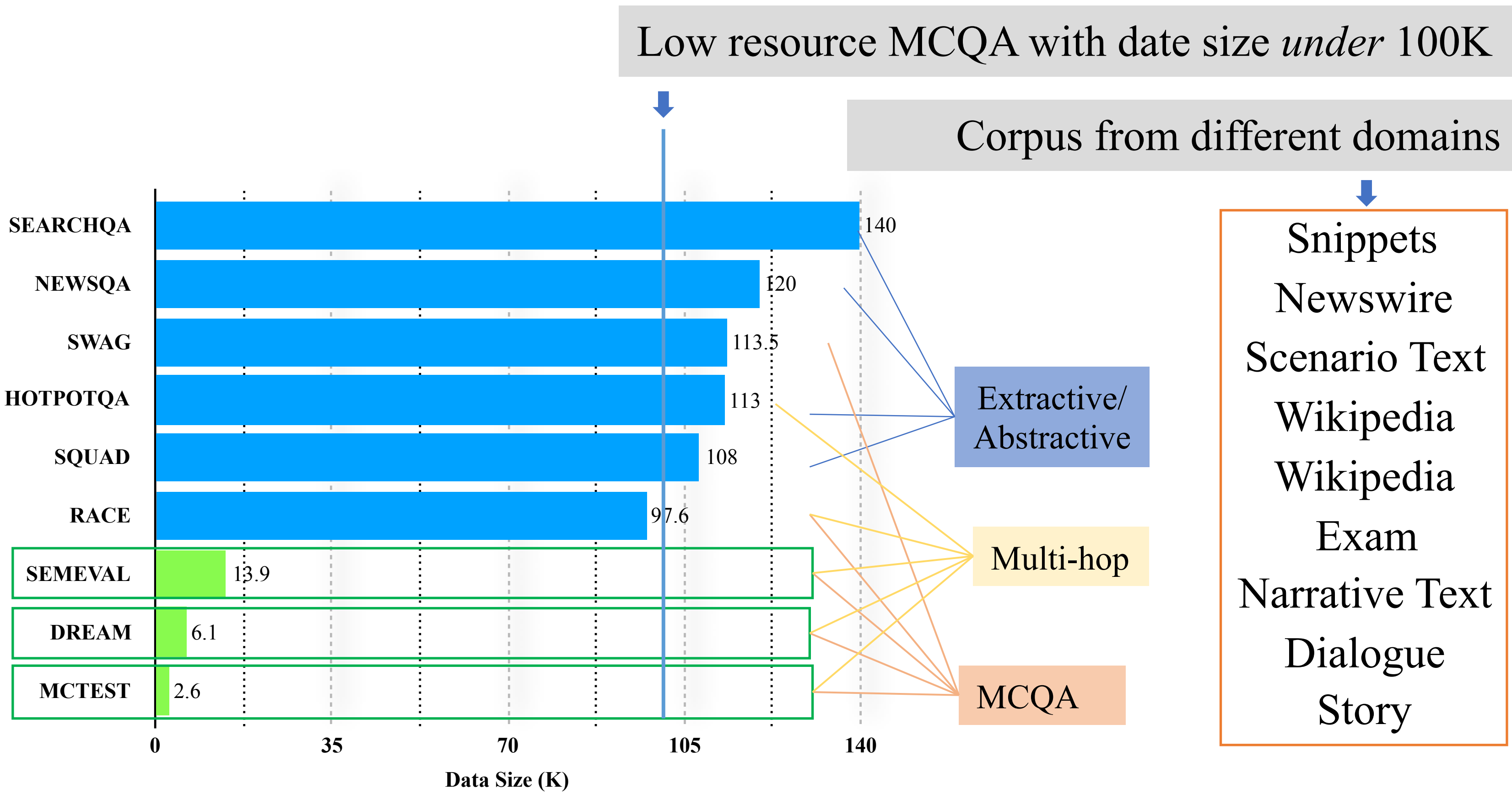
Multi-source Meta Transfer for Low Resource MCQA

Ming Yan[1], Hao Zhang[1,2], Di Jin[3], Joey Tianyi Zhou[1]

[1] IHPC, A*STAR, Singapore
[2] CSCE, NTU, Singapore
[3] CSAIL, MIT, USA

Agency for Science, Technology and Research SINGAPORE
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE
MIT CSAIL

Background



Low resource MCQA with date size *under* 100K
Corpus from different domains
SEARCHQA
NEWSQA
SWAG
HOTPOTQA
SQUAD
RACE
SEMEVAL
DREAM
MCTEST
140
120
113.5
113
108
97.6
13.9
6.1
2.6
0
35
70
105
140
Data Size (K)
Extractive/ Abstractive
Multi-hop
MCQA
Snippets
Newswire
Scenario Text
Wikipedia
Wikipedia
Exam
Narrative Text
Dialogue
Story

How does meta learning work?

- Low resource setting
- Domains discrepancy

Transfer learning, multi-task learning

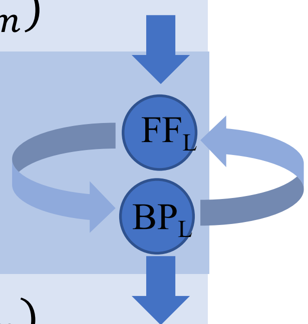
Fine-tuning on the target domain

$init\ w_m$ from $backbone\ model\ \ J: cost\ function$

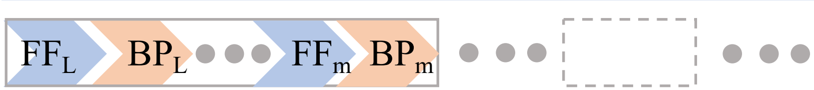
$Support\ tasks\colon x_l \sim X\ Enquiry\ tasks\colon x_m \sim X$

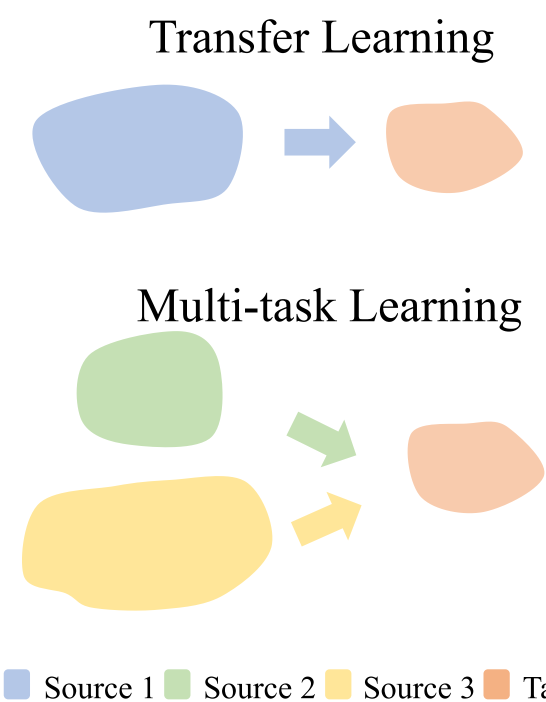
$$\text{model}_l =: \boldsymbol{copy}(model_m)$$

FF_L	$y_l = model_l(w_l, x_l)$
BP_L fast adaption	$w_l =: w_l + \alpha \dfrac{\partial J_l}{\partial w_l}$
FF_m	$y_m = model_l(w_l, x_m)$
BP_m meta-learning	$w_m =: w_m + \alpha \dfrac{\partial J_m}{\partial w_m}$



FF: Feedforward
BP: Backpropagation



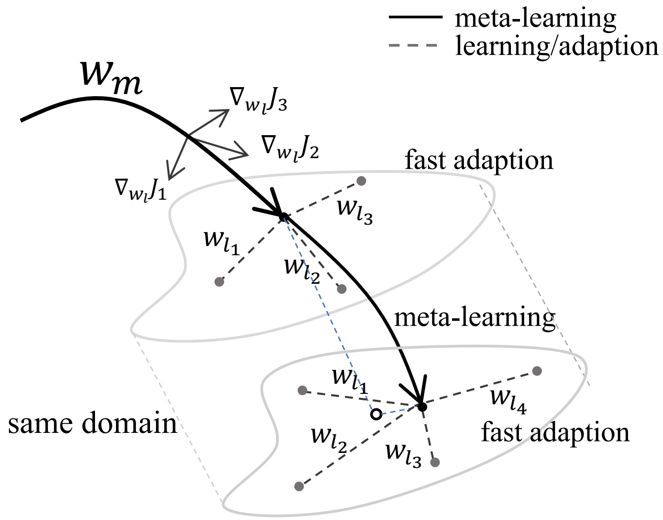


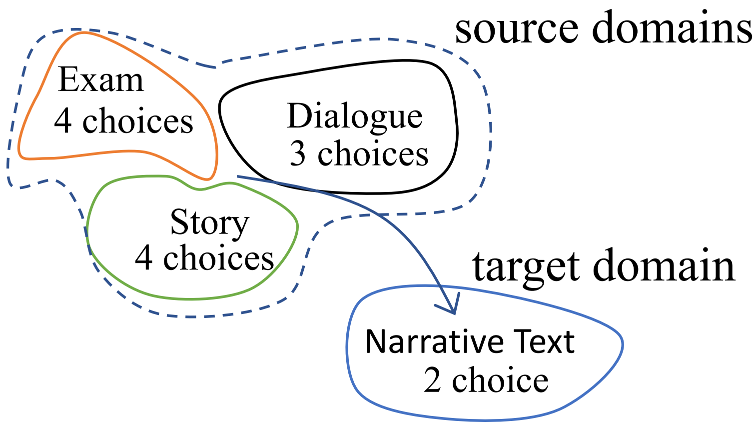
[Finn et al. 17] Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks, ICML 2017

How does meta learning work?

$init\ w_m$ from $pretrained\ model$

$Support\ T:\ x_l \sim X$ $Enquiry\ T:\ x_m \sim X$ $J:\ cost\ function$

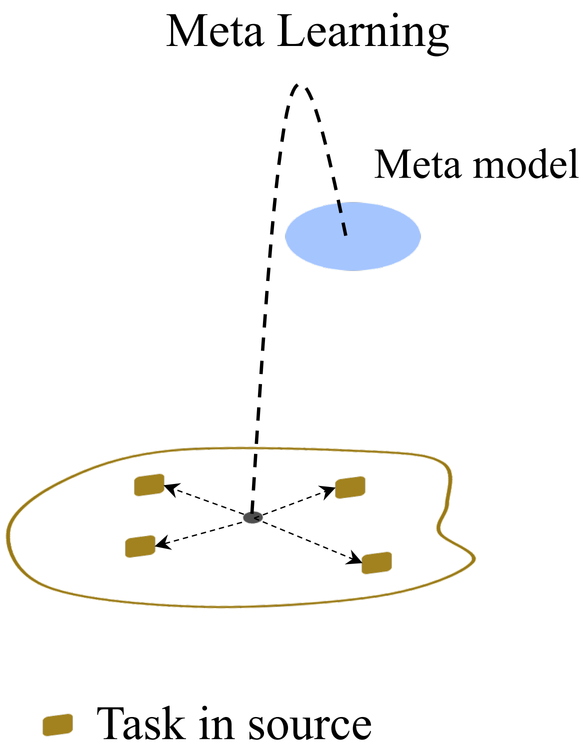


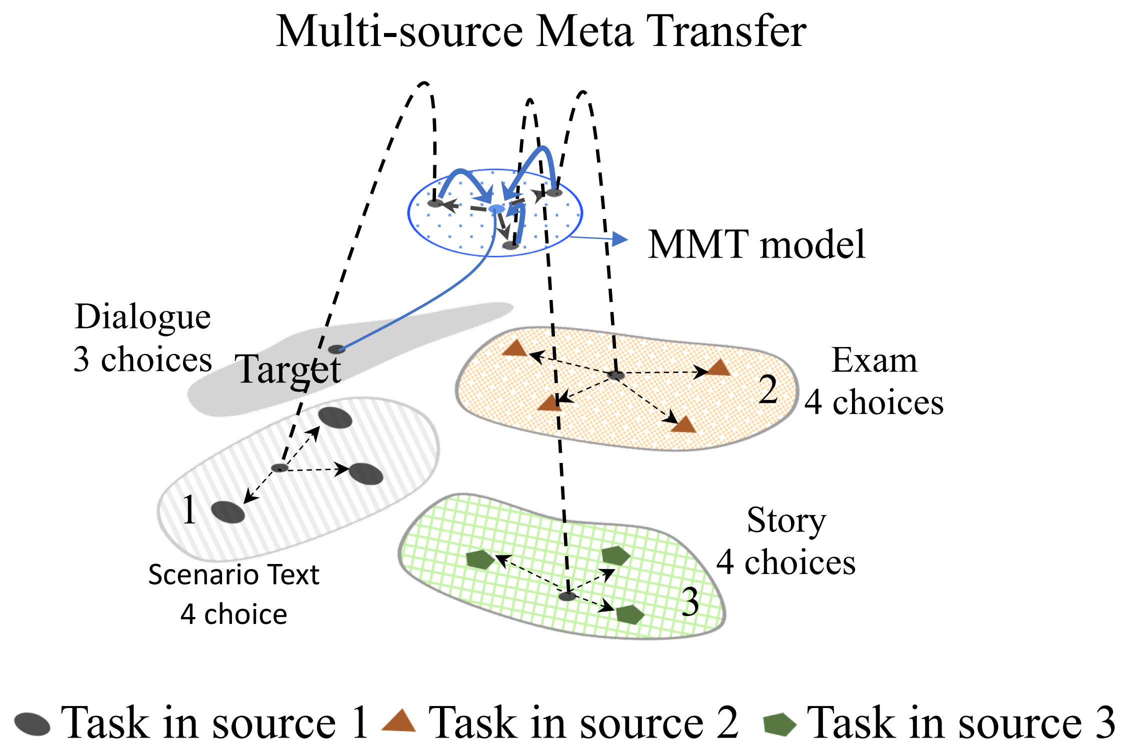


Learn a model that can generalize over the task distribution.

[Finn et al. 17] Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks, ICML 2017

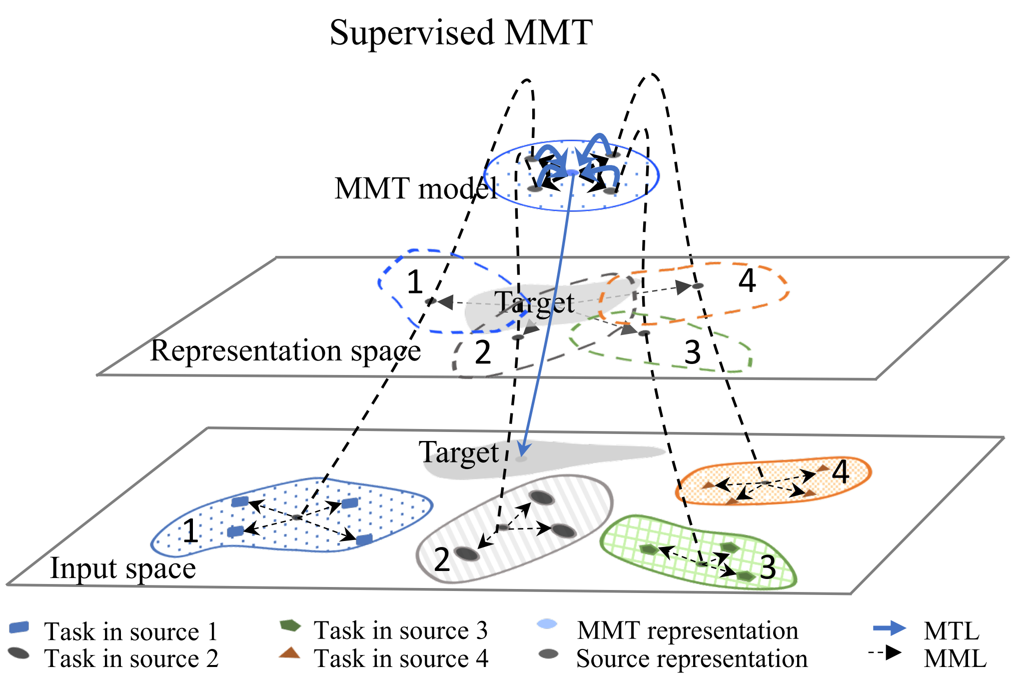
Multi-source Meta Transfer





- Learn knowledge from multiple sources
- Reduce discrepancy between sources and target.

Multi-source Meta Transfer



Multi-source Meta Learning (MML)

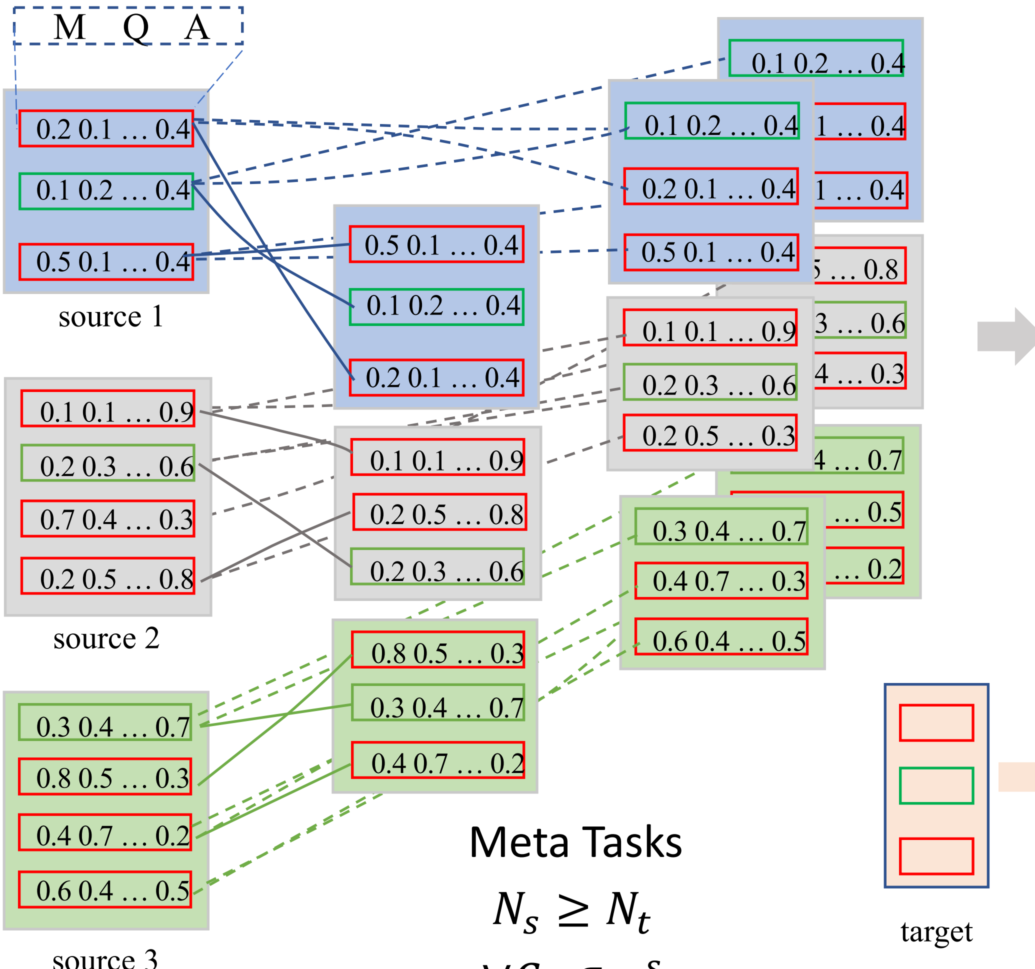
Learn knowledge from multiple sources.

Learn a representation near to the target.

Multi-source Transfer Learning (MTL)

Finetune meta-model to the target source.

How MMT samples the task?



Algorithm 1: The procedure of MMT

Input: Task distribution over source $p^s(\tau)$, data distribution over target $P^t(\tau)$, backbone model $f(\theta)$, learning rates in MMT α, β, λ

Output: Optimized parameters θ

Initial the value of θ

While *not done* **do**

  for *all source S* **do**

    Sample batch of tasks $\tau_i^s \sim p^s(\tau)$

    for all τ_i^s do

      Evaluate $\nabla_\theta L_{\tau_i^s}(f(\theta))$ with respect to k examples

      Compute gradient for fast adaption:

$$\theta' =: \theta - \alpha \nabla_\theta L_{\tau_i^s}(f(\theta))$$

    end

    Meta model update:

$$\theta =: \theta - \beta \nabla_\theta \sum_{\tau_i^s \sim p^s(\tau)} L_{\tau_i^s}(f(\theta'))$$

    Get batch of data $\tau_i^t \sim p^t(\tau)$

    for all τ_i^t **do**

      Evaluate $\nabla_\theta L_{\tau_i^t}(f(\theta))$ with respect to k examples

      Gradient for target fine-tuning:

$$\theta =: \theta - \beta \nabla_\theta L_{\tau_i^t}(f(\theta))$$

    end

  end

end

Get all batches of data $\tau_i^t \sim p^t(\tau)$

for all τ_i^t **do**

  Evaluate with respect to batch size;

  Gradient for meta transfer learning:

$$\theta =: \theta - \beta \nabla_\theta L_{\tau_i^t}(f(\theta))$$

end

Multi-source Meta Transfer

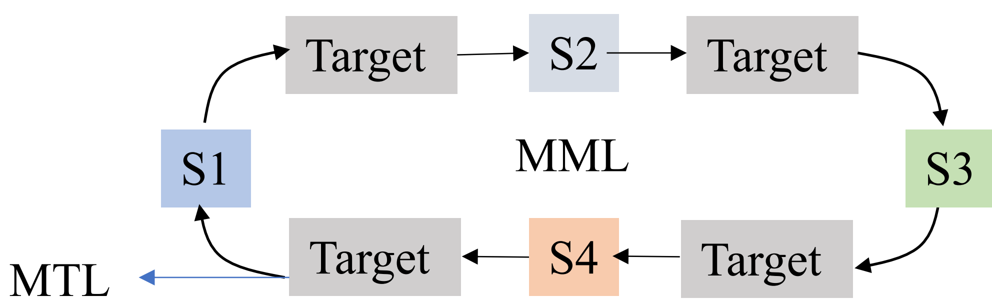
MMT is agnostic
to backbone models

Support task and Query task sampled
from the same distribution

Updated the learner (θ') on support task

Updated the meta model (θ) on query task

Updated the meta model (θ) on target data



Transfer meta model to the target

Algorithm 1: The procedure of MMT

Input: Task distribution over source $p^s(\tau)$, data distribution over target $P^t(\tau)$, backbone model $f(\theta)$, learning rates in MMT α, β, λ
Output: Optimized parameters θ

Initial the value of θ

While *not done* **do**

  for *all source S* **do**

    Sample batch of tasks $\tau_i^s \sim p^s(\tau)$

    for all τ_i^s do

      Evaluate $\nabla_\theta L_{\tau_i^s}(f(\theta))$ with respect to k examples

      Compute gradient for fast adaption:

      $$\theta' =: \theta - \alpha \nabla_\theta L_{\tau_i^s}(f(\theta))$$

    end

    Meta model update:

    $$\theta =: \theta - \beta \nabla_\theta \sum_{\tau_i^s \sim p^s(\tau)} L_{\tau_i^s}(f(\theta'))$$

    Get batch of data $\tau_i^t \sim p^t(\tau)$

    for all τ_i^t **do**

      Evaluate $\nabla_\theta L_{\tau_i^t}(f(\theta))$ with respect to k examples

      Gradient for target fine-tuning:

      $$\theta =: \theta - \beta \nabla_\theta L_{\tau_i^t}(f(\theta))$$

    end

    MML

  end

end

Get all batches of data $\tau_i^t \sim p^t(\tau)$

for all τ_i^t **do**

  Evaluate with respect to batch size;

  Gradient for meta transfer learning:

  $$\theta =: \theta - \beta \nabla_\theta L_{\tau_i^t}(f(\theta))$$

end

MTL

Results

Methods	DREAM		MCTEST		SemEval	
	Dev	Test	Dev	Test	Dev	Test
CoMatching (Wang et al., 2018)	45.6	45.5	-	-	-	-
HFL (Chen et al., 2018)	-	-	-	-	86.46	84.13
QACNN (Chung et al., 2018)	-	-	-	72.66	-	-
IMC (Yu et al., 2019)	-	-	-	76.59	-	-
XLNet (Yang et al., 2019)	-	72.0	-	-	-	-
GPT+Strategies (2×) (Sun et al., 2019b)	-	-	-	81.9	-	89.5
BERT-Base	60.05	61.58	70.0	67.98	86.03	87.53
RoBERTa†	82.16	84.37	88.37	87.26	93.76	94.00
MMT (BERT-Base)	**68.38**	**68.89**	**81.56**	**82.02**	**88.52**	**88.85**
MMT (RoBERTa)†	**83.87**	**85.55**	**88.66**	**88.80**	**94.33**	**94.24**

Performance of Supervised MMT

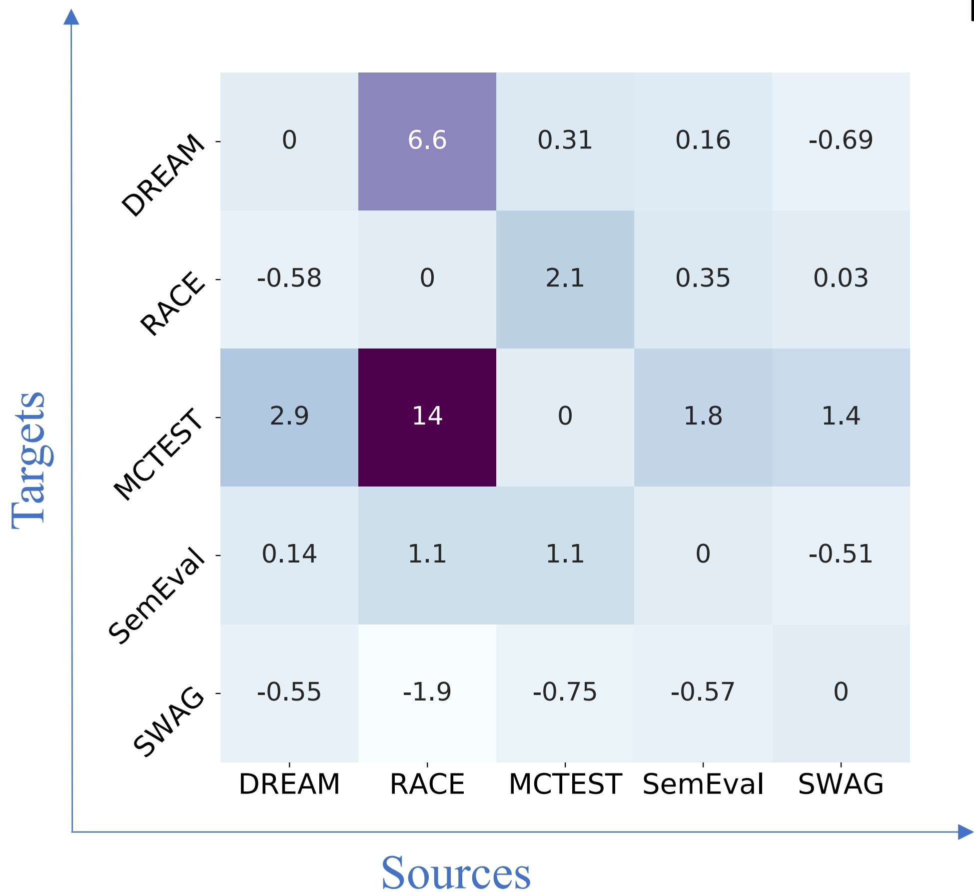
Method	**Sup.**	**Test**
Bert-Base	Yes	67.98
QACNN (Chung et al., 2018)	Yes	72.66
IMC (Yu et al., 2019)	Yes	76.59
MemN2N (Chung et al., 2018)	No	53.39
QACNN (Chung et al., 2018)	No	63.10
TL(S)	No	50.02
TL(R)	No	77.02
TL(R-S)	No	62.97
TL(S-R)	No	77.38
TL(R+S)	No	79.17
Unsupervised MMT(S+R)	No	**81.55**

MCTEST Performance of Unsupervised MMT

Dream	**Dev**	**Test**
BERT-Base	60.05	61.58
+MML(M)	49.85	52.87
+MML(R)	49.56	51.69
+MML(M∪R)	29.60	29.20
+TL(M)	60.31	60.14
+TL(R)	68.72	67.72
+TL(R-M)	68.97	67.38
+TL(M+R)	68.61	68.15
+MMT(M)	67.99	68.54
+MMT(R)	68.04	68.69
+MMT(M∪R)	61.72	60.12
MMT(M+R)	**68.38**	**68.89**

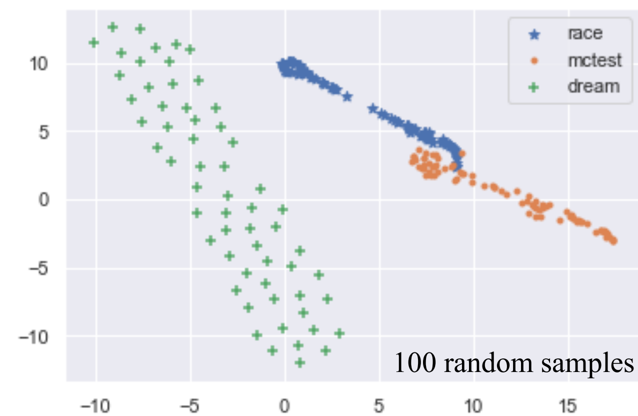
MMT Ablation Study

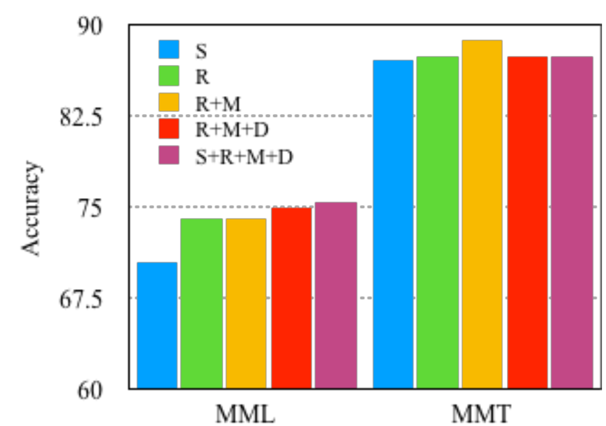
How to select sources?



Transferability Matrix

T-SNE Visualization of BERT Feature





Test on SemEval 2018

Takeaways

- MMT extends to meta learning to multi-source on MCQA task
- MMT provided an algorithm both for supervised and unsupervised meta training
- MMT give a guideline to source selection