Paper sharing

——ICML非正式+ICLR

- XTREME: A Massively Multilingual Multi-task Benchmark for Evaluating Cross-lingual Generalization
- Learning to Branch for Multi-Task Learning
- Efficient Continuous Pareto Exploration in Multi-Task Learning
- Understanding and Improving Information Transfer in Multi-Task Learning

SHARING KNOWLEDGE IN MULTI-TASK DEEP REINFORCEMENT LEARNING

2020/8/9 ZhuJingdan

XTREME: A Massively Multilingual Multi-task Benchmark for Evaluating Cross-lingual Generalization

Junjie Hu *[1] Sebastian Ruder *[2] Aditya Siddhant [3] Graham Neubig [1] Orhan Firat [3] Melvin Johnson [3]

In NLP, there is a pressing urgency to build systems that serve all of the world's approximately 6,900 languages to overcome language barriers and enable universal information access for the world's citizens .

This paper introduce the Cross-lingual TRansfer Evaluation of Multilingual Encoders (XTREME) benchmark.XTREME covers 40 typologically diverse languages spanning 12 language families and includes 9 tasks that require reasoning about different levels of syntax or semantics.

——如：英语的“ desk ”和德语的“ Tisch ” 均源自拉丁文 “ discus ”

□Cross-lingual representations

- parallel corpora or bilingual dictionary to learn a linear transformation
- self-training or unsupervised strategies

↓

multilingual extensions of pretrained encoders

□Cross-lingual evaluation

- related languages and similar domains, it does not capture differences in classification performance that are due to cultural differences
- cross-lingual approaches have been evalu_x0002_ated on a wide range of tasks

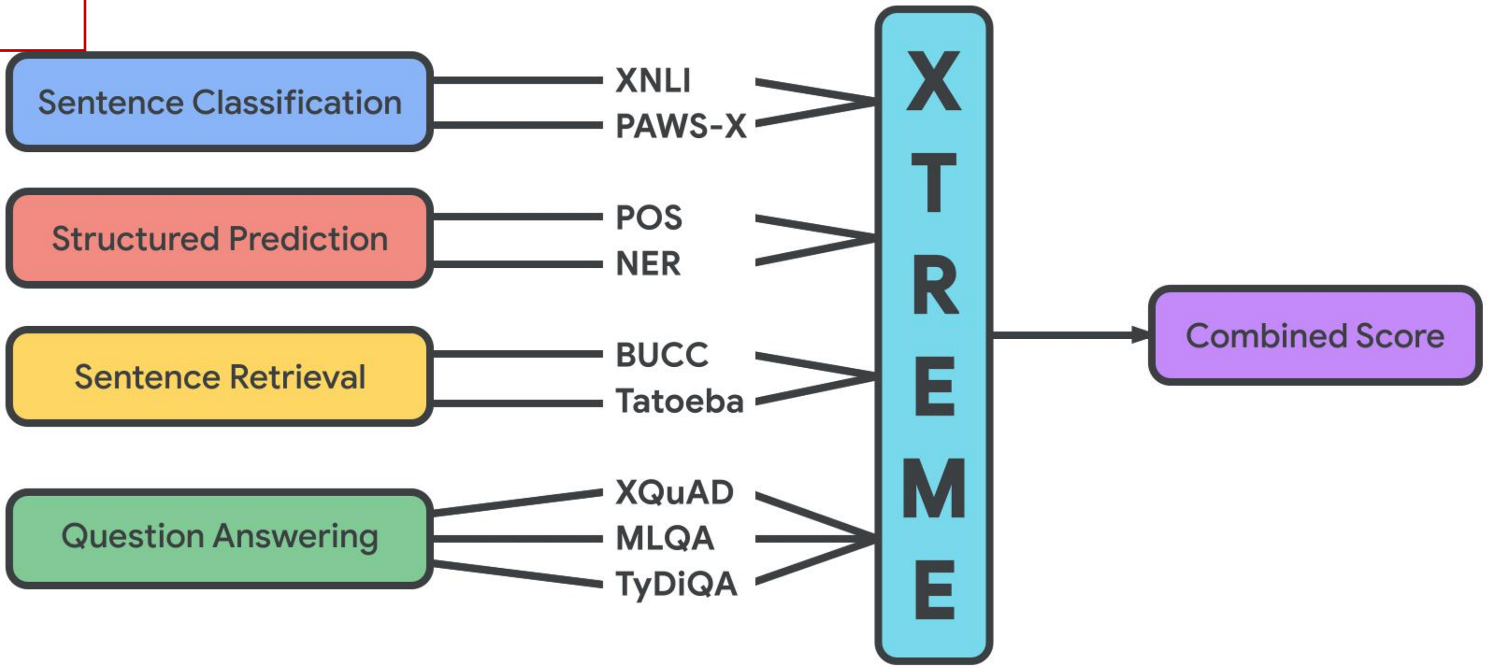
↓

a benchmark not only needs to cover a diverse set of tasks but also languages

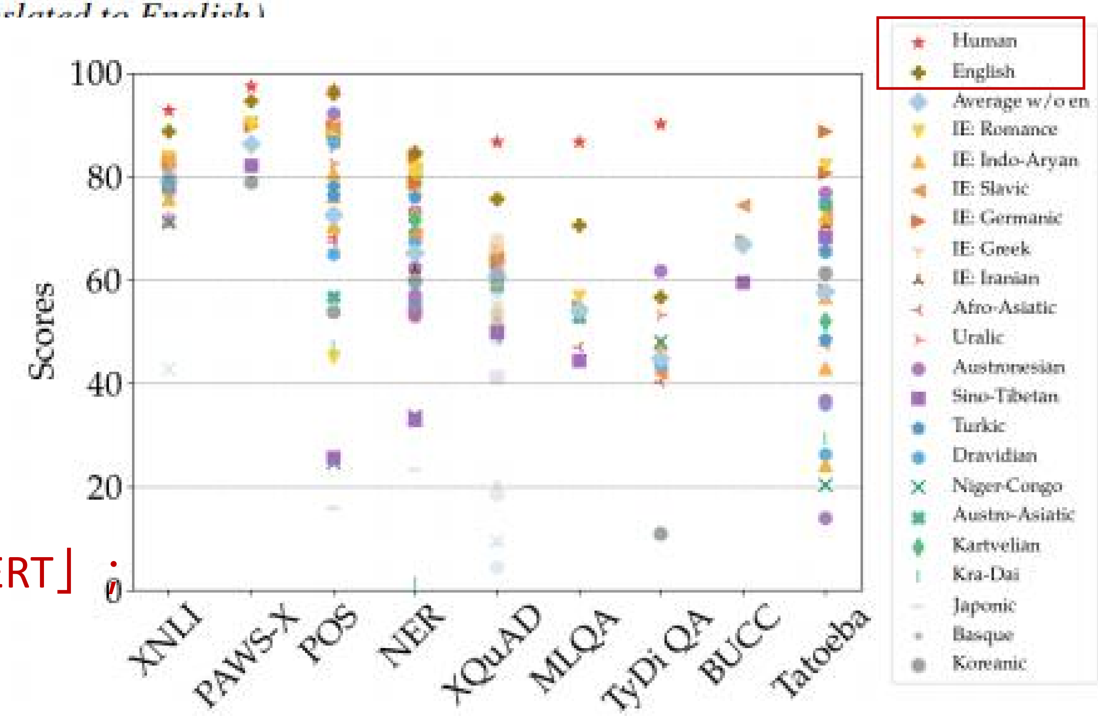
Characteristics of the datasets in XTREME for the zero-shot transfer setting

Task	Corpus	|Train|	|Dev|	|Test|	Test sets	|Lang.|	Task	Metric	Domain
Classification	XNLI	392,702	2,490	5,010	translations	15	NLI	Acc.	Misc.
	PAWS-X	49,401	2,000	2,000	translations	7	Paraphrase	Acc.	Wiki / Quora
Struct. pred.	POS	21,253	3,974	47-20,436	ind. annot.	33 (90)	POS	F1	Misc.
	NER	20,000	10,000	1,000-10,000	ind. annot.	40 (176)	NER	F1	Wikipedia
QA	XQuAD	87,599	34,726	1,190	translations	11	Span extraction	F1 / EM	Wikipedia
	MLQA			4,517–11,590	translations	7	Span extraction	F1 / EM	Wikipedia
	TyDiQA-GoldP	3,696	634	323–2,719	ind. annot.	9	Span extraction	F1 / EM	Wikipedia
Retrieval	BUCC	-	-	1,896–14,330	-	5	Sent. retrieval	F1	Wiki / news
	Tatoeba	-	-	1,000	-	33 (122)	Sent. retrieval	Acc.	misc.

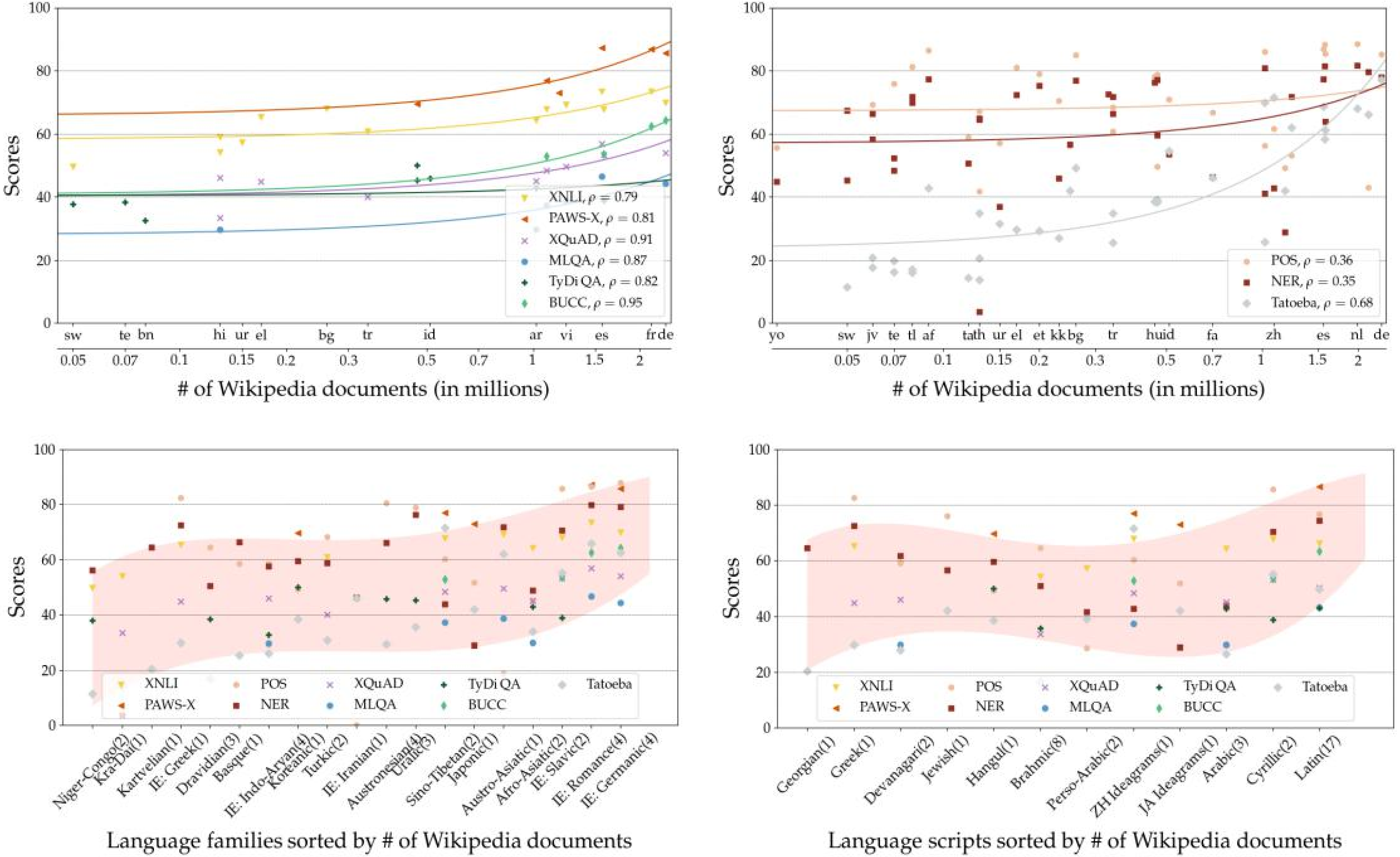
9 tasks



Model	Avg	Pair sentence XNLI	Pair sentence PAWS-X	Structured prediction POS	Structured prediction NER	Question answering XQuAD	Question answering MLQA	Question answering TyDiQA-GoldP	Sentence retrieval BUCC	Sentence retrieval Tatoeba
Metrics		Acc.	Acc.	F1	F1	F1 / EM	F1 / EM	F1 / EM	F1	Acc.
Cross-lingual zero-shot transfer (models are trained on English data)										
mBERT	59.6	65.4	81.9	70.3	62.2	64.5 / 49.4	61.4 / 44.2	59.7 / 43.9	56.7	38.7
XLM	55.5	69.1	80.9	70.1	61.2	59.8 / 44.3	48.5 / 32.6	43.6 / 29.1	56.8	32.6
XLM-R Large	68.1	79.2	86.4	72.6	65.4	76.6 / 60.8	71.6 / 53.2	65.1 / 45.0	66.0	57.3
MMTE	59.3	67.4	81.3	72.3	58.3	64.4 / 46.2	60.3 / 41.4	58.1 / 43.8	59.8	37.9
Translate-train (models are trained on English training data translated to the target language)										
mBERT	-	74.0	86.3	-	-	70.0 / 56.0	65.6 / 48.0	55.1 / 42.1	-	-
mBERT, multi-task	-	75.1	88.9	-	-	72.4 / 58.3	67.6 / 49.8	64.2 / 49.3	-	-
Translate-test (models are trained on English data and evaluated on target language data translated to English)										
BERT-large	-	76.5	84.4	-	-	76.3 / 62.1				
In-language models (models are trained on the target language training data)										
mBERT, 1000 examples	-	-	-	87.6	77.9	-				
mBERT	-	-	-	89.8	88.3	-				
mBERT, multi-task	-	-	-	91.5	89.1	-				
Human	-	92.8	97.5	97.0	-	91.2 / 82.3				



mBERT：BERT 的多语言扩展版本；
XLM 和 XLM-R Large：规模更大、数据处理量更多版本的「多语言 BERT」；
MMTE：大规模多语言机器翻译模型。



Scores
# of Wikipedia documents (in millions)
XNLI, ρ = 0.79
PAWS-X, ρ = 0.81
XQuAD, ρ = 0.91
MLQA, ρ = 0.87
TyDi QA, ρ = 0.82
BUCC, ρ = 0.95
sw te bn hi ur el bg tr id ar vi es fr de
POS, ρ = 0.36
NER, ρ = 0.35
Tatoeba, ρ = 0.68
yo sw jv te tl af ta th ur el et kk bg tr hu id fa zh es nl de
Language families sorted by # of Wikipedia documents
Niger-Congo(2)
Kra-Dai(1)
Kartvelian(1)
IE: Greek(1)
Dravidian(3)
Basque(1)
IE: Indo-Aryan(4)
Koreanic(1)
Turkic(2)
IE: Iranian(1)
Austronesian(4)
Uralic(3)
Sino-Tibetan(2)
Japonic(1)
Austro-Asiatic(1)
Afro-Asiatic(2)
IE: Slavic(2)
IE: Romance(4)
IE: Germanic(4)
XNLI
PAWS-X
POS
NER
XQuAD
MLQA
TyDi QA
BUCC
Tatoeba
Language scripts sorted by # of Wikipedia documents
Georgian(1)
Greek(1)
Devanagari(2)
Jewish(1)
Hangul(1)
Brahmic(8)
Perso-Arabic(2)
ZH Ideagrams(1)
JA Ideagrams(1)
Arabic(3)
Cyrillic(2)
Latin(17)

Learning to Branch for Multi-Task Learning

Pengsheng Guo [1] Chen-Yu Lee [1] Daniel Ulbricht [1]

Over-sharing a network could erroneously enforce over-generalization, causing negative knowledge transfer across tasks.

This paper introduce a novel tree-structured design space that casts a tree branching operation as a gumbelsoftmax sampling procedure. This enables differentiable network splitting that is end-to-end trainable.

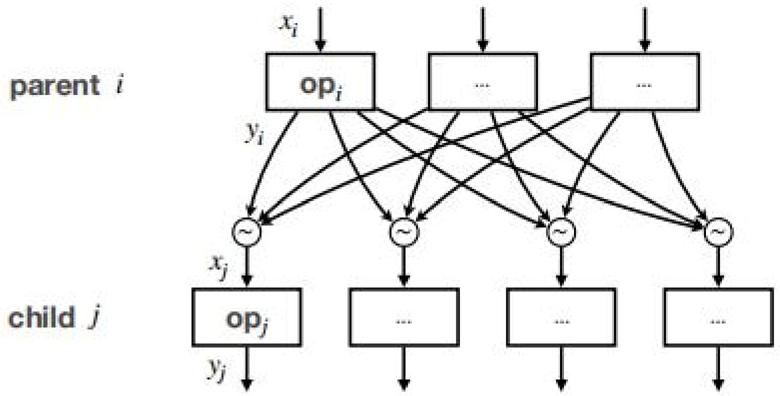
Given a set of N tasks $\mathcal{T} = \{t_1, t_2, ..., t_N\}$, the goal of the proposed method is to learn a tree-structured (Lee et al., 2016) network architecture Ω and the weight values ω of the network that minimize the overall loss $\mathcal{L}_{\text{total}}$ across all tasks,

$$\begin{aligned} \omega^*, \Omega^* &= \arg\min_{\omega,\Omega} \mathcal{L}_{\text{total}}(\omega, \Omega) \\ &= \arg\min_{\omega,\Omega} \sum_k \boxed{\alpha_k} \boxed{\mathcal{L}_k}(\omega, \Omega) \end{aligned}$$

Loss of task k

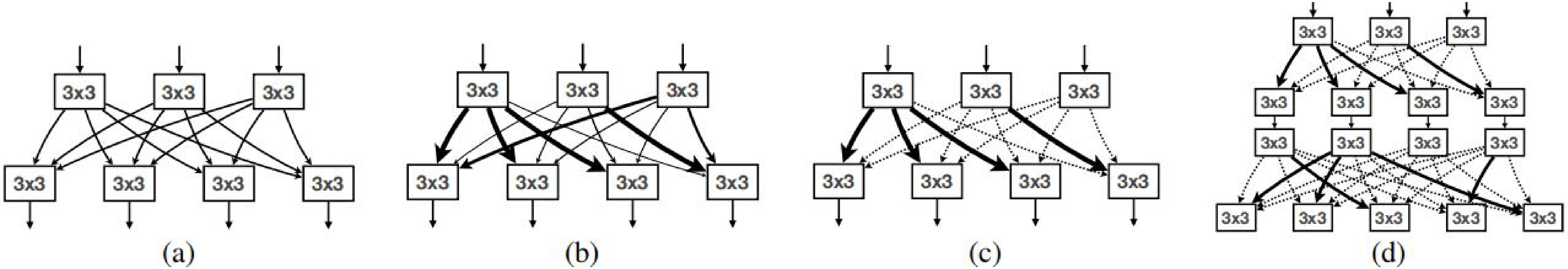
Weights of task k

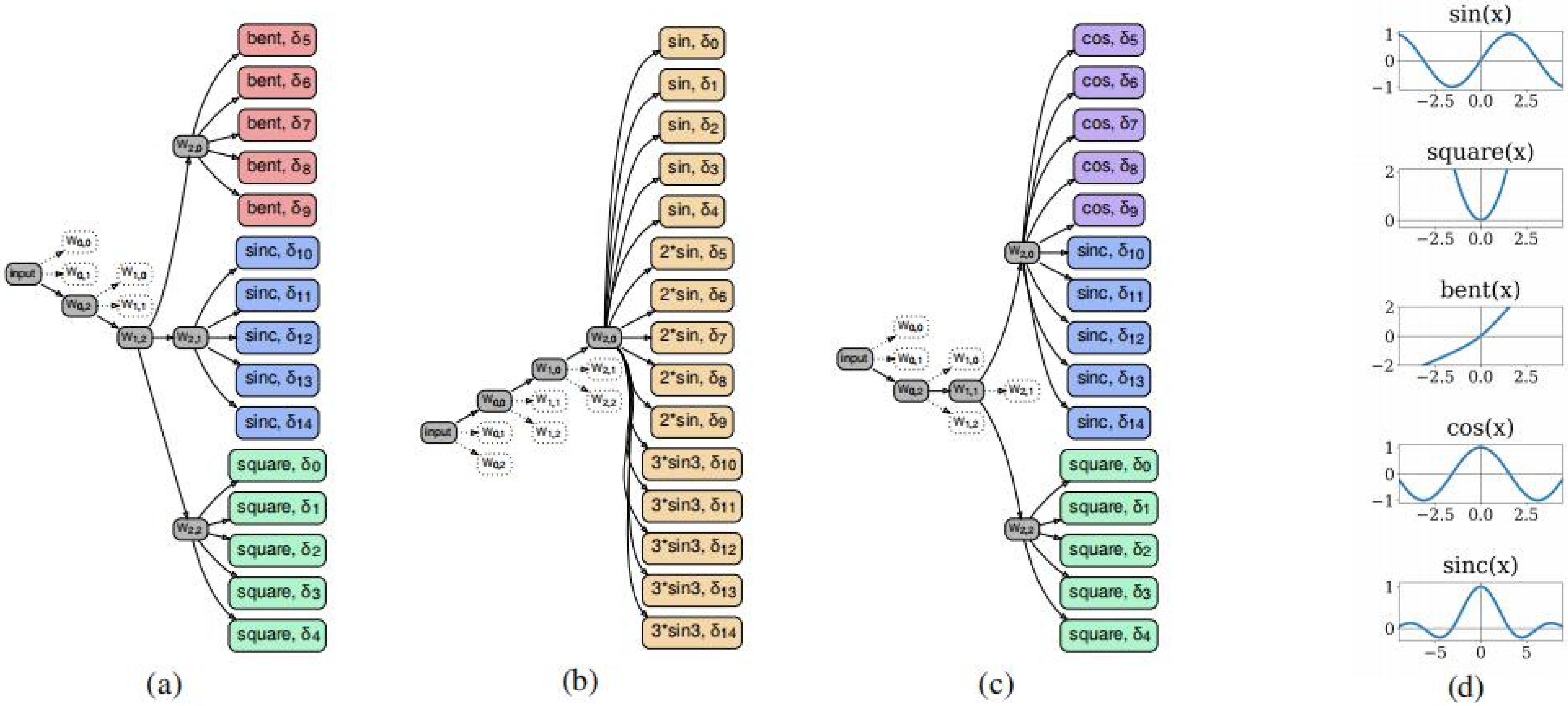
The key ingredient for effective and efficient network configuration sampling is our proposed differentiable tree-structured network topology. The topological space is represented as a Directed Acyclic Graph (DAG) where the nodes represent computational operations and the edges denote data flows. Figure 1 illustrates a certain block of a DAG



Specifically, we construct multiple parent nodes and child nodes for each block and allow a child node to sample a path from all the paths between it and all its parent nodes. The

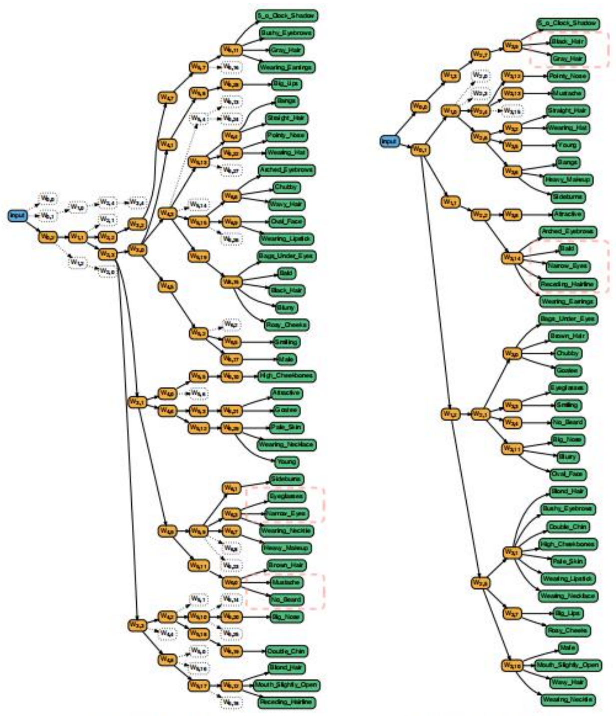
$$x_j^{l+1} = \mathbb{E}_{d_j \sim p_{\theta_j}}[d_j \cdot Y^l]$$


3x3
(a)
(b)
(c)
(d)


bent, δ5
bent, δ6
bent, δ7
bent, δ8
bent, δ9
sinc, δ10
sinc, δ11
sinc, δ12
sinc, δ13
sinc, δ14
square, δ0
square, δ1
square, δ2
square, δ3
square, δ4
input
sin, δ0
sin, δ1
sin, δ2
sin, δ3
sin, δ4
2*sin, δ5
2*sin, δ6
2*sin, δ7
2*sin, δ8
2*sin, δ9
3*sin3, δ10
3*sin3, δ11
3*sin3, δ12
3*sin3, δ13
3*sin3, δ14
cos, δ5
cos, δ6
cos, δ7
cos, δ8
cos, δ9
sin(x)
square(x)
bent(x)
cos(x)
sinc(x)
(a)
(b)
(c)
(d)

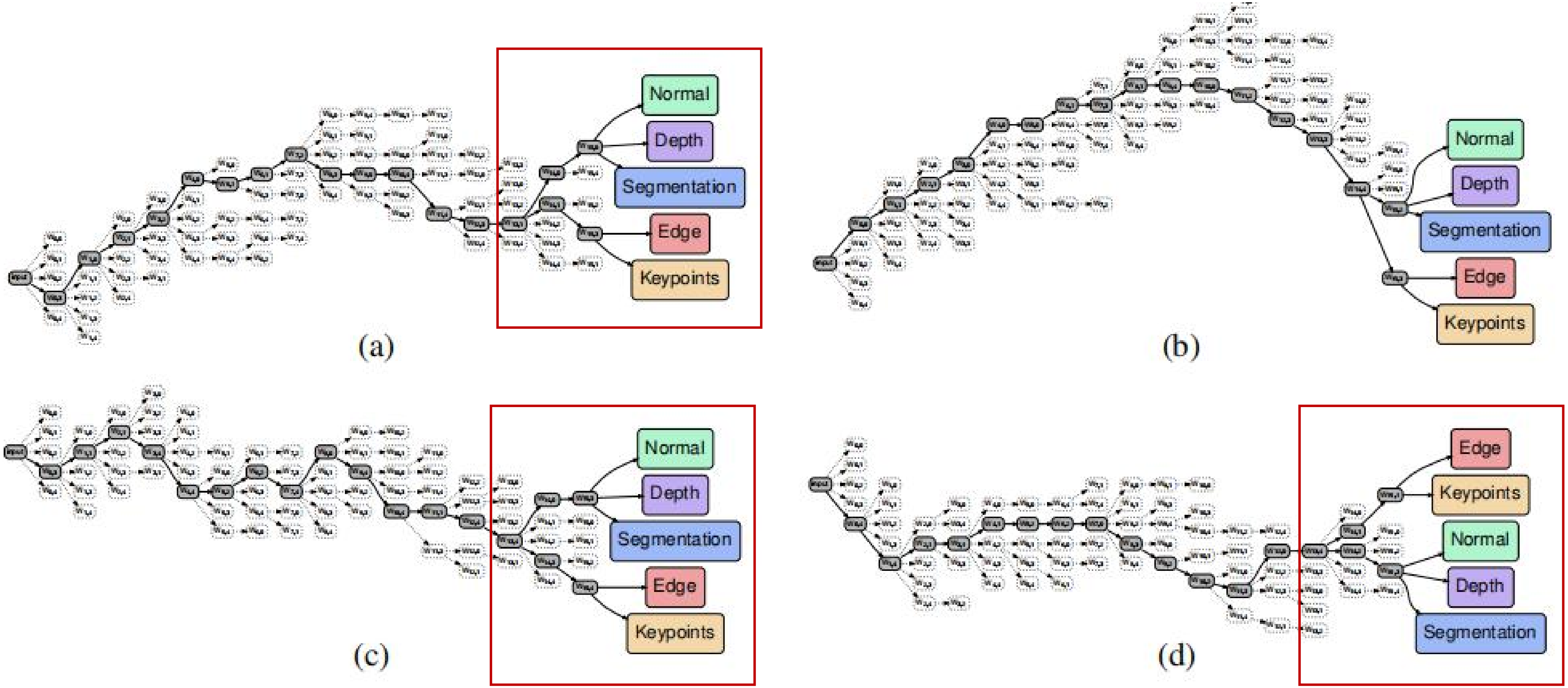
CelebA

METHOD	ACC (%)	PARAMS (M)
LNET+ANET (WANG ET AL., 2016)	87	-
WALK AND LEARN (WANG ET AL., 2016)	88	-
MOON (RUDD ET AL., 2016)	90.94	119.73
INDEP GROUP (HAND & CHELLAPPA, 2017)	91.06	-
MCNN-AUX (HAND & CHELLAPPA, 2017)	91.29	-
VGG-16 BASELINE (LU ET AL., 2017)	91.44	134.41
BRANCH-VGG (LU ET AL., 2017)	90.79	2.09
LEARNTOBRANCH-VGG (OURS)	**91.55**	**1.94**
GNAS-DEEP-WIDE (HUANG ET AL., 2018)	91.36	6.41
LEARNTOBRANCH-DEEP-WIDE (OURS)	**91.62**	**6.33**



(a) LearnToBranch-VGG (b) LearnToBranch-Deep-Wide

METHOD	PARAMS (M)	SEGMENTATION ↓	NORMAL ↑	DEPTH ↓	KEYPOINT ↓	EDGE ↓
SINGLE-TASK (SUN ET AL., 2019)	124	0.575	0.707	0.022	0.197	0.212
MULTI-TASK (SUN ET AL., 2019)	41	0.587	0.702	0.024	0.194	0.201
CROSS-STITCH (MISRA ET AL., 2016)	124	0.560	0.684	0.022	0.202	0.219
SLUICE (RUDER ET AL., 2017)	124	0.610	0.702	0.023	0.192	0.198
NDDR-CNN (GAO ET AL., 2019)	133	0.539	0.705	0.024	0.194	0.206
MTAN (LIU ET AL., 2019B)	114	0.637	0.702	0.023	0.193	0.203
ADASHARE (SUN ET AL., 2019)	41	0.566	0.707	0.025	0.192	0.193
LEARNTOBRANCH (OURS)	51	**0.462**	**0.709**	**0.018**	**0.122**	**0.136**


Normal
Depth
Segmentation
Edge
Keypoints
(a)
Normal
Depth
Segmentation
Edge
Keypoints
(b)
Normal
Depth
Segmentation
Edge
Keypoints
(c)
Edge
Keypoints
Normal
Depth
Segmentation
(d)

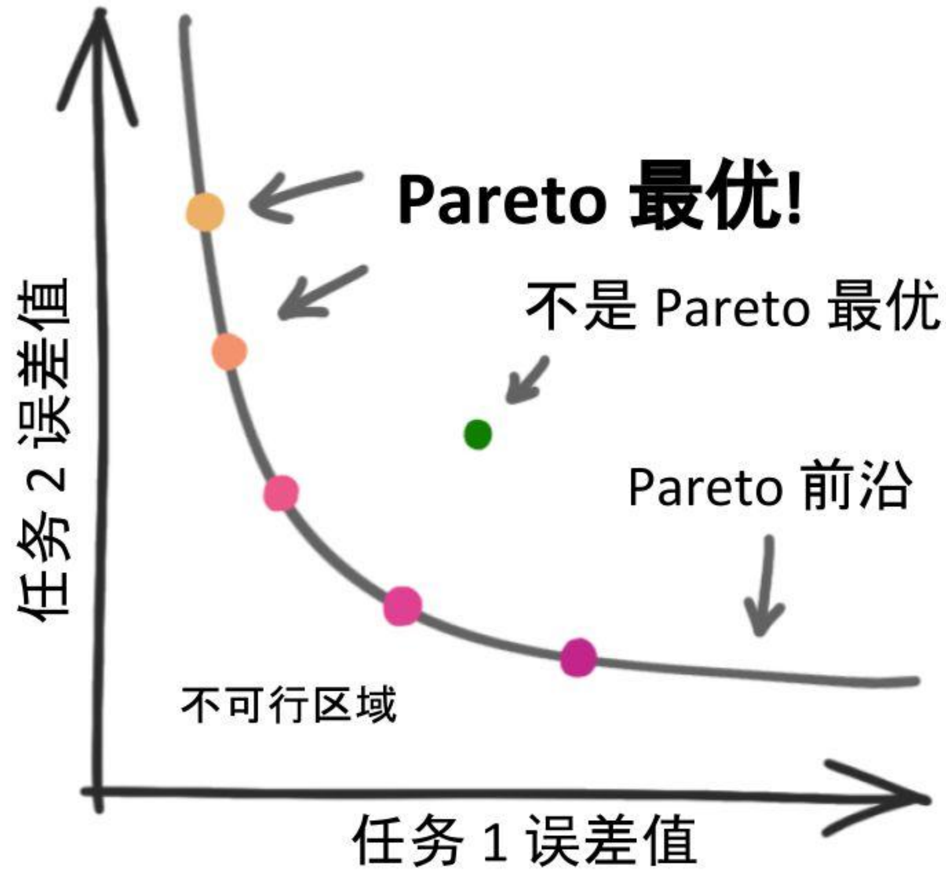
Efficient Continuous Pareto Exploration in Multi-Task Learning

Pingchuan Ma *[1] Tao Du *[1] Wojciech Matusik [1]

Tasks in multi-task learning often correlate, conflict, or even compete with each other. As a result,a single solution that is optimal for all tasks rarely exists.

Author present an efficient method that generates locally continuous Pareto sets and Pareto fronts, which opens up the possibility of continuous analysis of Pareto optimal solutions in machine learning problems.

Pareto（Pareto optimality，帕累托最优）



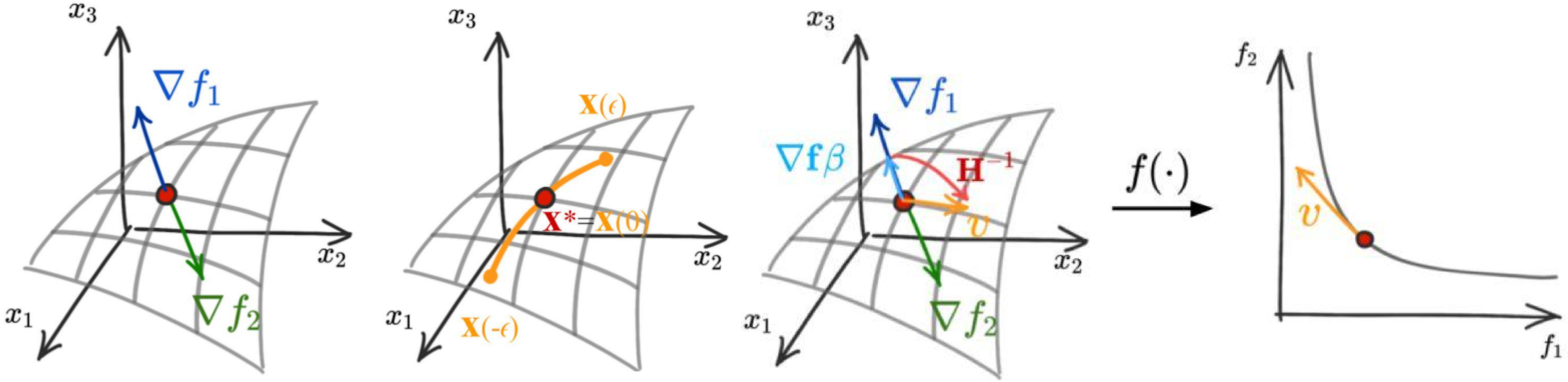
optimization (e.g., training a neural network). In order to obtain an efficient algorithm for computing a continuous Pareto set, it is necessary to exploit local information. Our technical method is inspired by second-order methods in multi-objective optimization (MOO) (Hillermeier, 2001; Martín & Schütze, 2018; Schulz et al., 2018) which connect the local tangent plane, the gradient information, and the Hessian matrices at a Pareto optimal solution all in one concise linear equation. This theorem allows us to construct a

例：在学校里很饿的时候有3个选择：
- **吃炸鸡**（好吃，但是不健康）
- **吃沙拉**（健康，但是不好吃）
- 吃食堂（又不好吃，又不健康）

此时炸鸡和沙拉都是 Pareto 最优解， 但食堂就不是 Pareto 最优解

$$(\sum \alpha_i \nabla^2 f_i)\mathbf{x}'(0) = -\sum \alpha_i' \nabla f_i(\mathbf{x}^*)\,\mathbf{r}$$

$$\mathbf{H}v = \nabla \mathbf{f}\beta \in \mathrm{colspan}\{\nabla f_i\}$$

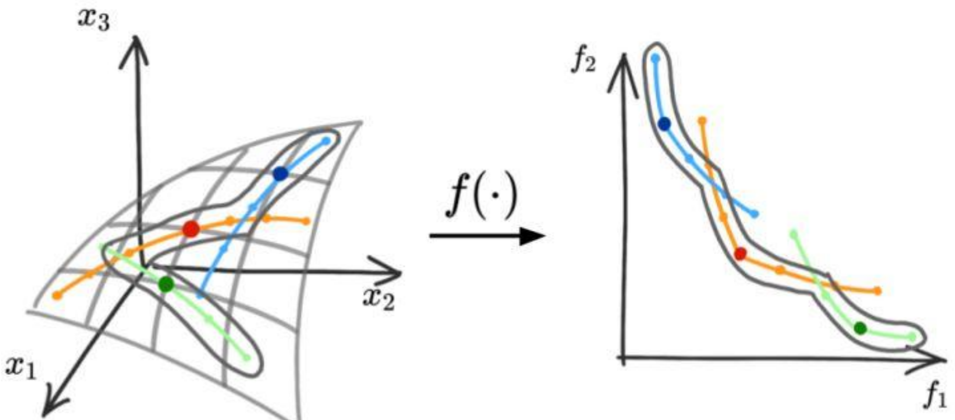


Definition 3.1 (Hillermeier et al. 2001). *Assuming each $f_i(\boldsymbol{x})$ is continuously differentiable, a point $\boldsymbol{x}$ is called Pareto stationary if there exists $\boldsymbol{\alpha} \in \mathbb{R}^m$ such that $\alpha_i \geq 0$, $\sum_{i=1}^m \alpha_i = 1$, and $\sum_{i=1}^m \alpha_i \nabla f_i(\boldsymbol{x}) = \mathbf{0}$.*

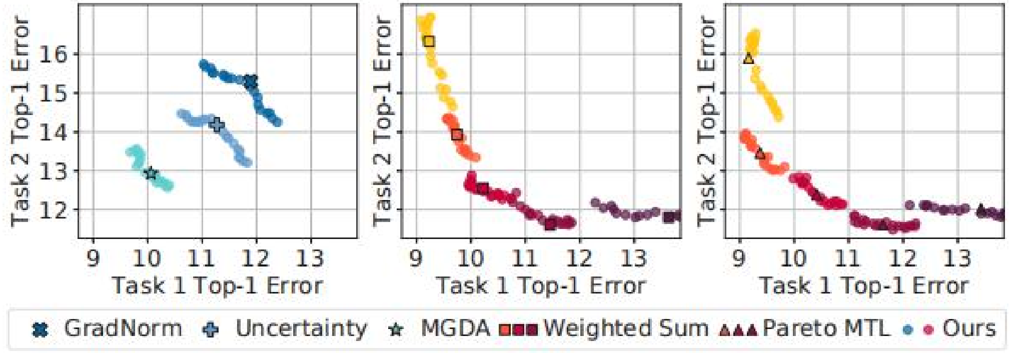
Proposition 3.2 (Hillermeier 2001). *Assuming that $\boldsymbol{f}(\boldsymbol{x})$ is smooth and $\boldsymbol{x}^*$ is Pareto optimal, consider any smooth curve $\boldsymbol{x}(t) : (-\epsilon, \epsilon) \to \mathbb{R}^n$ in the Pareto set and passing $\boldsymbol{x}^*$ at $t = 0$, i.e., $\boldsymbol{x}(0) = \boldsymbol{x}^*$, then $\exists \boldsymbol{\beta} \in \mathbb{R}^m$ such that:*

$$\boldsymbol{H}(\boldsymbol{x}^*)\boldsymbol{x}'(0) = \nabla \boldsymbol{f}(\boldsymbol{x}^*)^\top \boldsymbol{\beta} \tag{1}$$

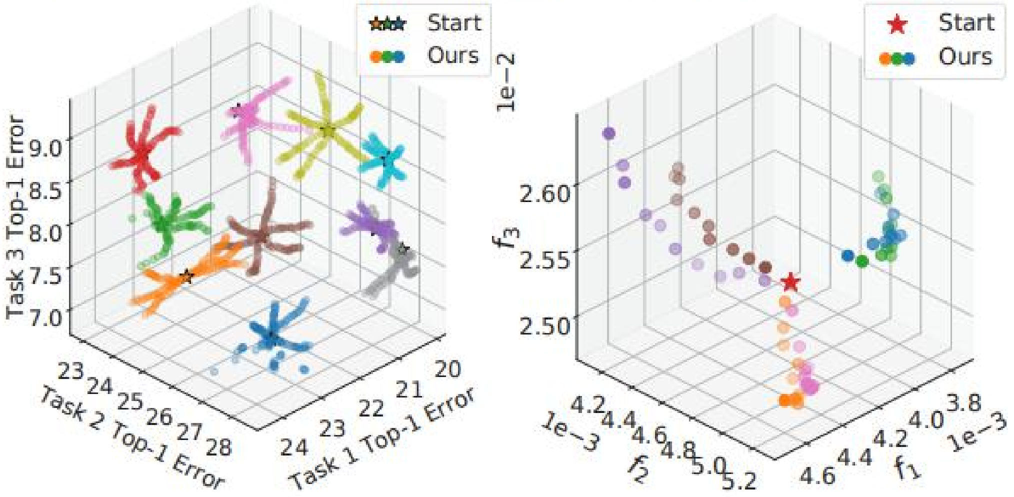
Krylov子空间迭代法



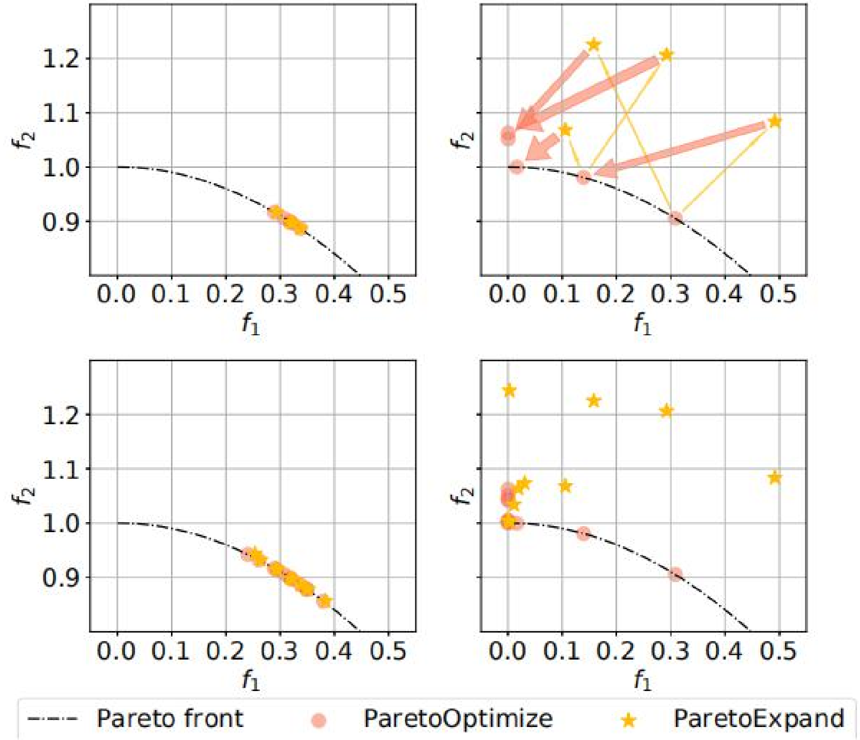
x_3
x_2
x_1
$f(\cdot)$
f_2
f_1



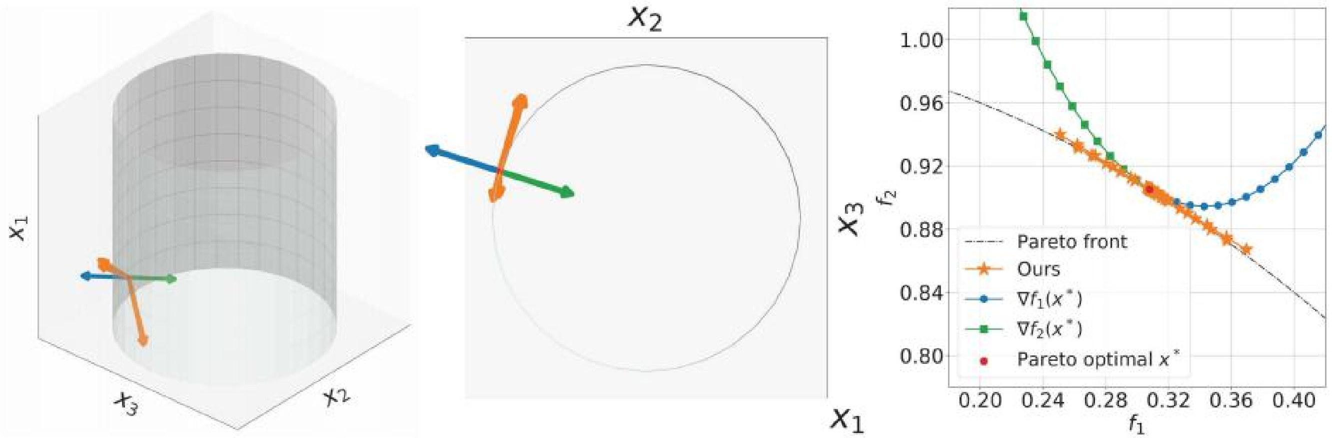
Task 2 Top-1 Error
Task 1 Top-1 Error
GradNorm
Uncertainty
MGDA
Weighted Sum
Pareto MTL
Ours



Start
Ours
Task 3 Top-1 Error
Task 2 Top-1 Error
Task 1 Top-1 Error
1e−2
1e−3
f_3
f_2
f_1



f_2
f_1
Pareto front
ParetoOptimize
ParetoExpand



x_1
x_2
x_3
f_2
f_1
Pareto front
Ours
$\nabla f_1(x^*)$
$\nabla f_2(x^*)$
Pareto optimal x^*

Understanding and Improving Information Transfer in Multi-Task Learning

Sen Wu*
Stanford University

Hongyang R. Zhang*
University of Pennsylvania

Christopher Ré
Stanford University

Multi-task learning has recently emerged as a powerful paradigm in deep learning. While in some cases, great improvements have been reported compared to single-task learning , practitioners have also observed problematic outcomes, where the performances of certain tasks have decreased due to task interference.

They develop methods to improve the effectiveness and robustness of multi-task training.

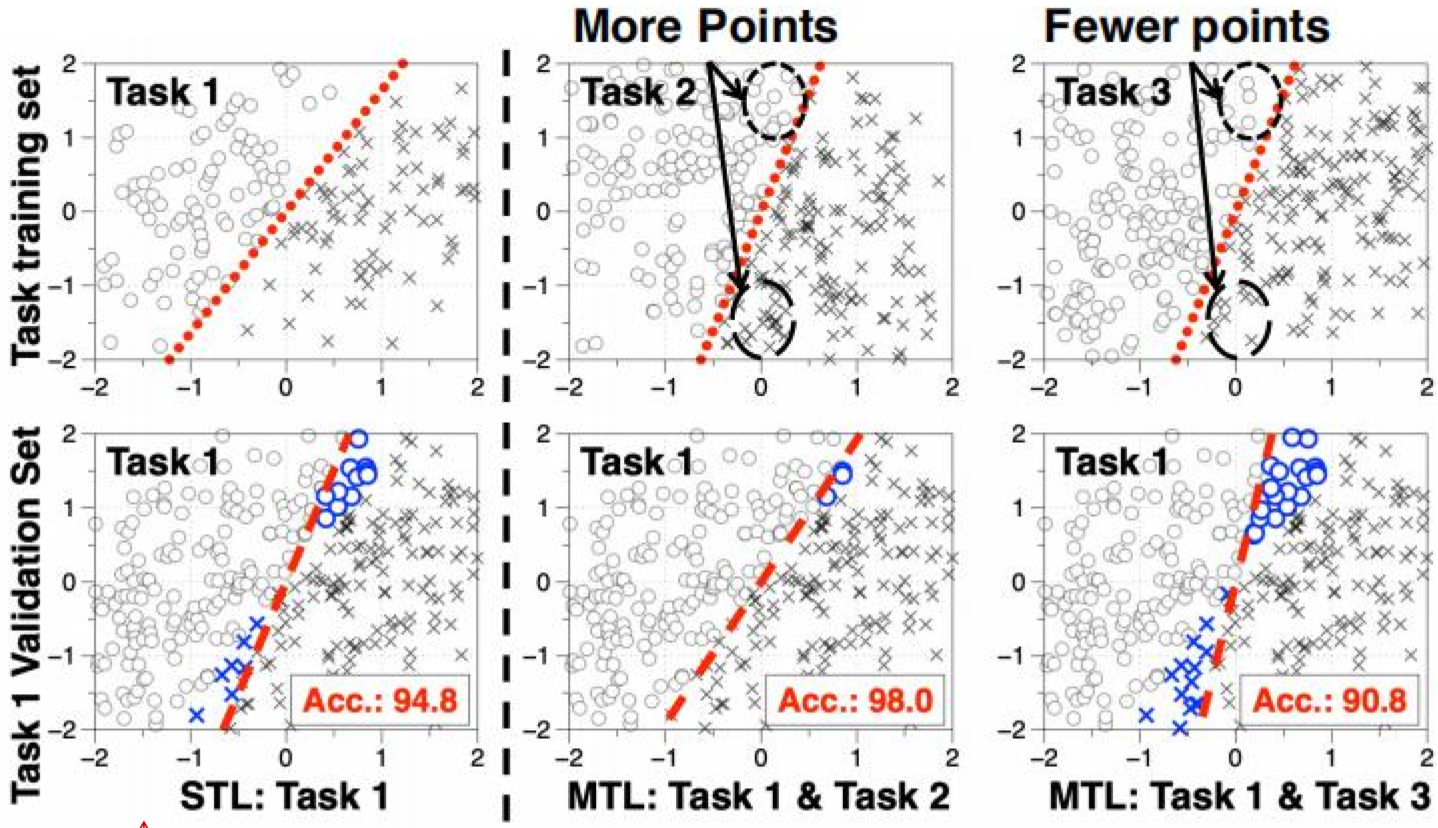
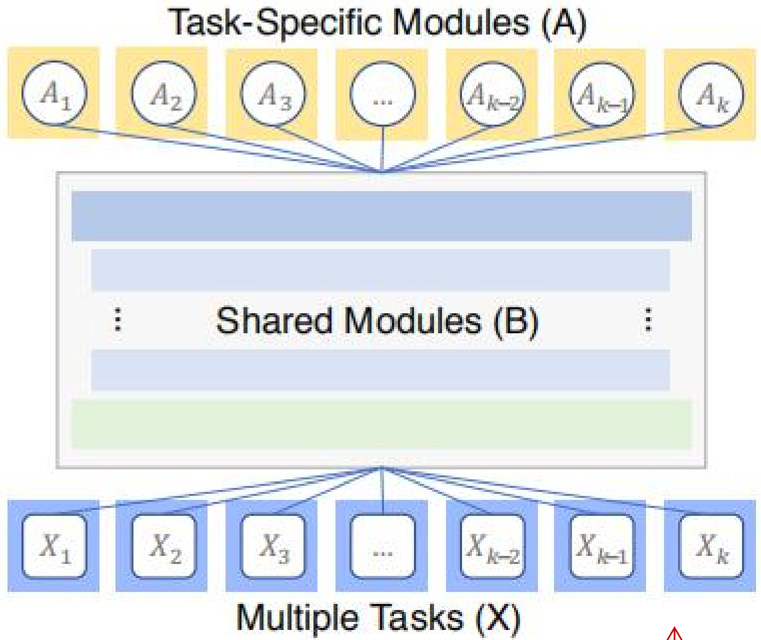


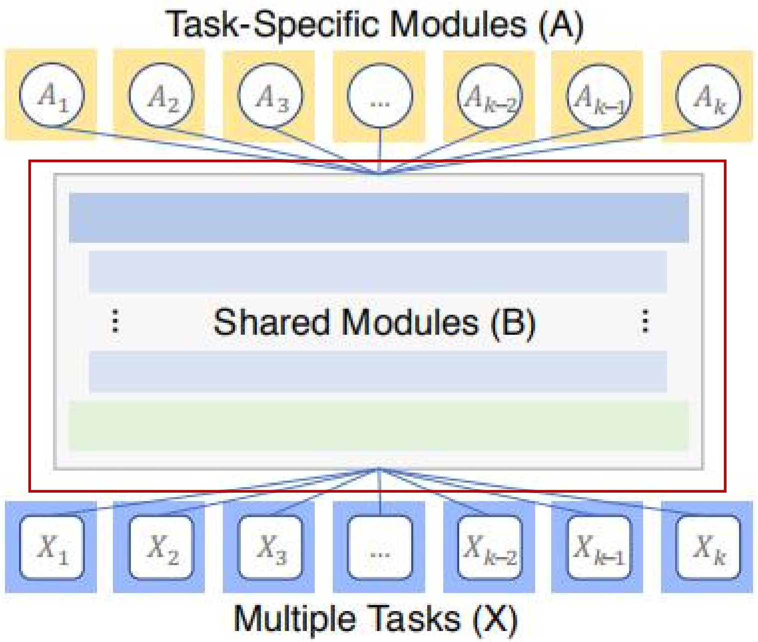
Figure 2 for an illustration). We observe that training task 1 with task 2 or task 3 can either improve or hurt task 1's performance, depending on the amount of contributing data along the decision

the multi-task learning architecture with a shared lower module B and k task-specific modules $\{A_i\}_{i=1}^k$.



See Figure 1 for an illustration. Our motivating observation is that in addition to model similarity which affects the type of interference, task data similarity plays a second-order effect after controlling model similarity. To illustrate the idea, we consider three tasks with the same number of data

MODEL CAPACITY



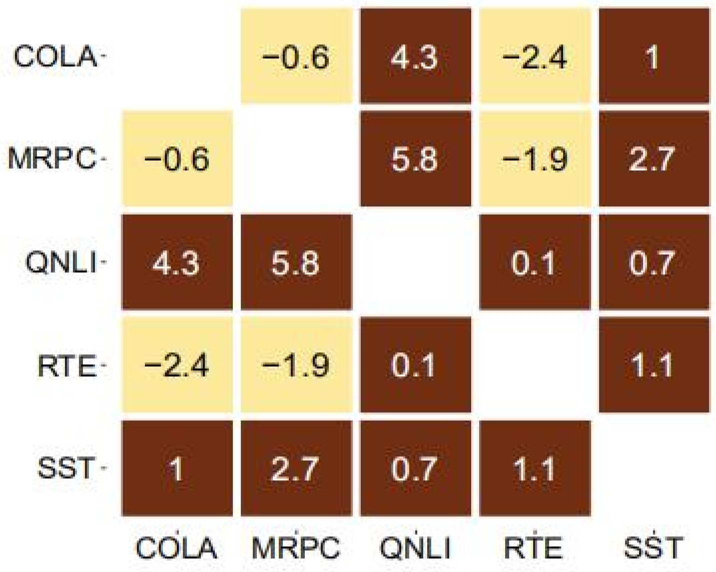
r

the output dimension of B

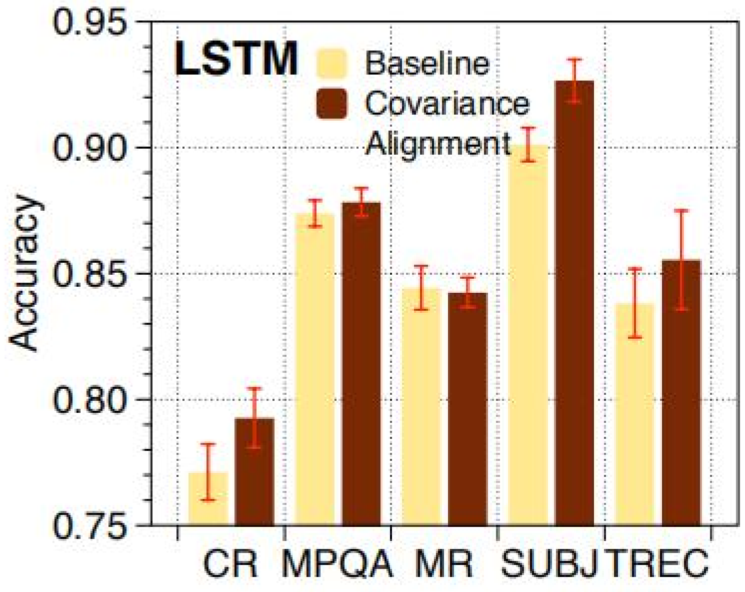
Task	STL		MTL	
	Cap.	Acc.	Cap.	Acc.
SST	200	82.3	100	**90.8**
MR	200	76.4		**96.0**
CR	5	73.2		**78.7**
SUBJ	200	**91.5**		89.5
MPQA	500	86.7		**87.0**
TREC	100	**85.7**		78.7
Overall	1205	82.6	100	85.1

We show that if $r \geq k$, then there is no transfer between any two tasks.

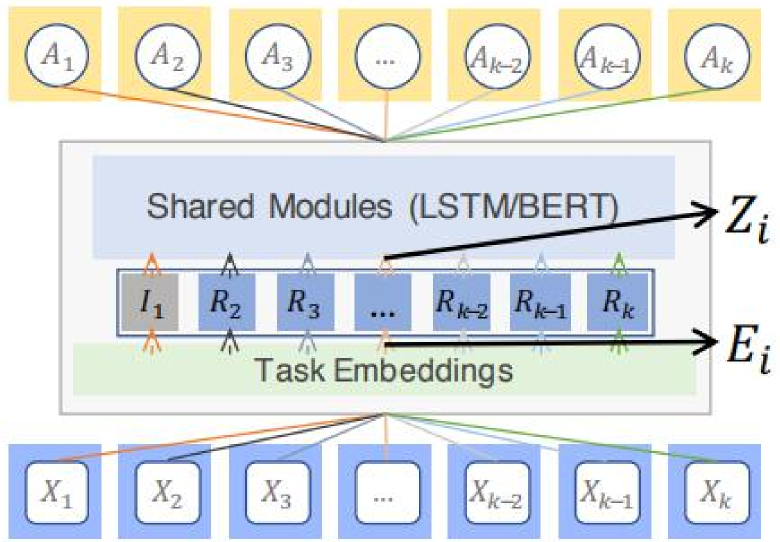
Task covariance

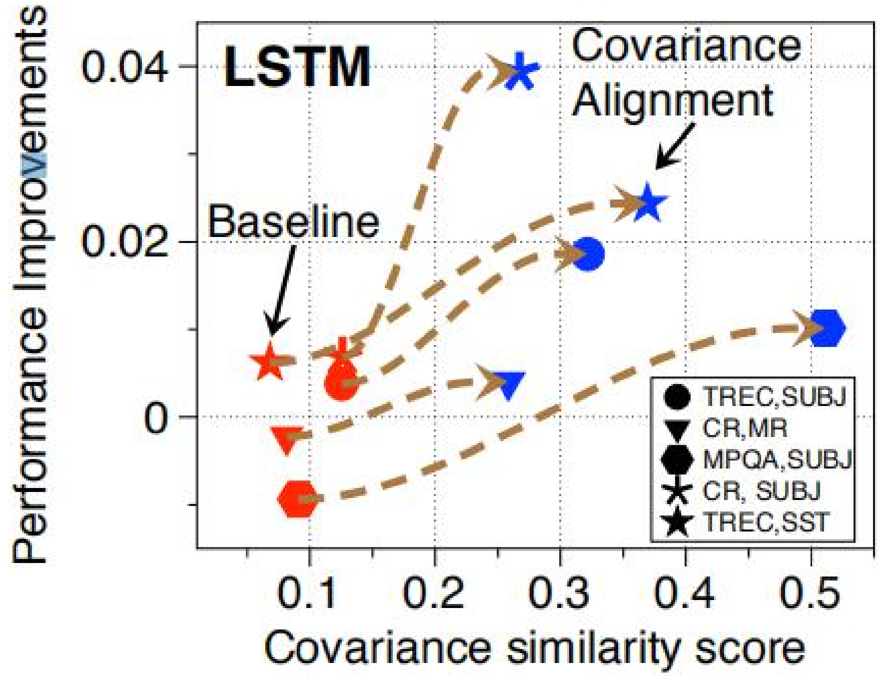


(a) MTL on GLUE over 10 task pairs



(b) Transfer learning on six sentiment analysis tasks





SHARING KNOWLEDGE IN MULTI-TASK DEEP REINFORCEMENT LEARNING

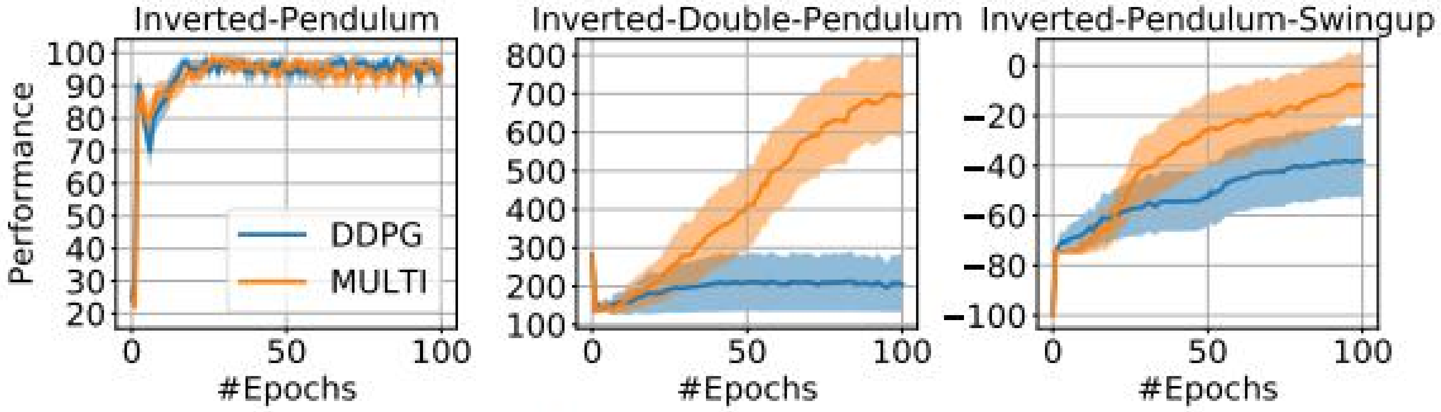
Carlo D'Eramo & Davide Tateo
Department of Computer Science
TU Darmstadt, IAS
Hochschulstraße 10, 64289, Darmstadt, Germany
{carlo.deramo,davide.tateo}@tu-darmstadt.de

Andrea Bonarini & Marcello Restelli
Politecnico di Milano, DEIB
Piazza Leonardo da Vinci 32, 20133, Milano
{andrea.bonarini,marcello.restelli}@polimi.it

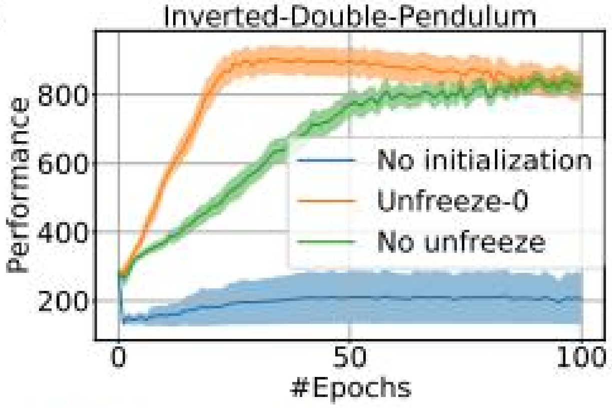
Jan Peters
TU Darmstadt, IAS
Hochschulstraße 10, 64289, Darmstadt, Germany
Max Planck Institute for Intelligent Systems
Max-Planck-Ring 4, 72076, Tübingen, Germany
jan.peters@tu-darmstadt.de

They study the benefit of sharing representations among tasks to enable the effective use of deep neural networks in Multi-Task Reinforcement Learning.

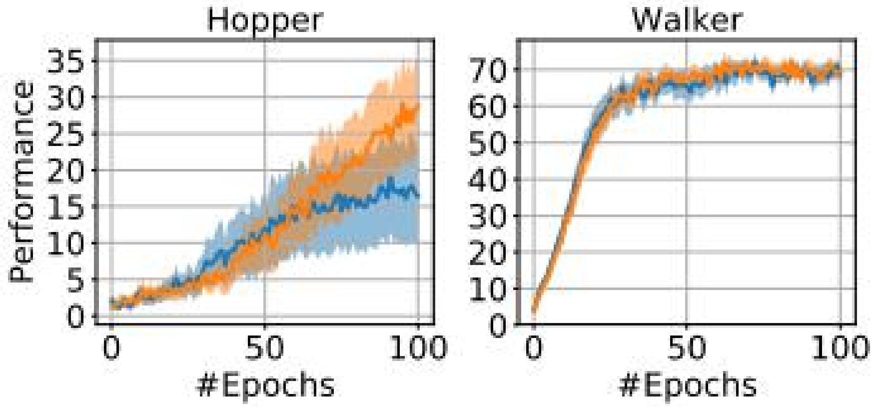
In addition, they complement their analysis by proposing multi-task extensions of three Reinforcement Learning algorithms.



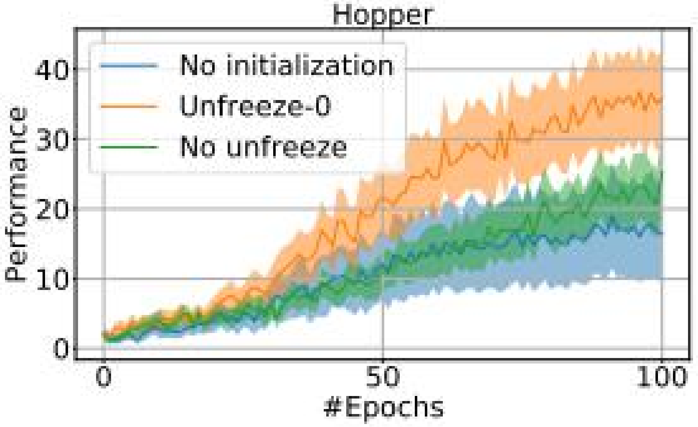
(a) Multi-task for pendulums



(b) Transfer for pendulums



(c) Multi-task for walkers



(d) Transfer for walkers

Thanks