Multi-Task Cooperative Learning via Searching for Flat Minima

Introduction

Multi-task learning (MTL) has shown great potential in medical image analysis, improving the generalizability of the learned features and the performance in individual tasks. However, most of the work on MTL focuses on either architecture design or gradient manipulation, while in both scenarios, features are learned in a competitive manner. In this work, we propose to formulate MTL as a multi/bi-level optimization problem, and therefore force features to learn from each task in a cooperative approach. Specifically, we update the sub-model for each task alternatively taking advantage of the learned sub-models of the other tasks. To alleviate the negative transfer problem during the optimization, we search for flat minima for the current objective function with regard to features from other tasks. To demonstrate the effectiveness of the proposed approach, we validate our method on three publicly available datasets. The proposed method shows the advantage of cooperative learning, and yields promising results when compared with the state-of-the-art MTL approaches.

(Methodology): Bi-Level Optimization for Cooperative Two-Task Learning

Formally, let $x_i \in \mathbb{R}^{W\times H\times C}$ denotes an image with the width W, height H and channel C, $y_i \in \mathbb{R}^{C_0}$ is a label for classification, (or $y_i \in \mathbb{R}^{W\times H\times C_0}$ for segmentation) and C_0 is the number of classes, $F_i(\cdot;\theta_i)$ is a feature extractor, $G_i(\cdot;\phi_i)$ is a prediction function for task $i = 1,\ldots,T$ where T is a number of tasks, and here $T = 2$. θ_i and ϕ_i are corresponding parameters to be learned. Our task is to predict label $\widehat{y}_i = G_i(F_i(x_i))$.

For MTL, instead of using shared backbone, *i.e.*, $F_1 = F_2$, and updating them simultaneously with a single loss ℓ, we propose to optimize them in a cooperative manner, that is learning (F_1, G_1) conditioned on a fixed and informative F_2, and versa vice. Generally, it can be formulated as a bi-level optimization problem:

$$(U)\ \min_{\theta_1,\phi_1} \mathcal{L}_1(\theta_1,\phi_1,\theta_2) = \ell_1(G_1(\mathcal{M}(F_1(x_1;\theta_1),F_2(x_1;\theta_2));\phi_1),\widehat{y}_1), \quad (1)$$

$$(L)\ \min_{\theta_2,\phi_2} \mathcal{L}_2(\theta_2,\phi_2,\theta_1) = \ell_2(G_2(\mathcal{M}(F_1(x_2;\theta_1),F_2(x_2;\theta_2));\phi_2),\widehat{y}_2), \quad (2)$$

where ℓ_i is the loss function, e.g. cross-entropy loss for classification. $\mathcal{M}$ denotes a feature fusion to facilitate the current task learning by incorporating useful information from other tasks.

(Methodology): Finding Flat minima via Injecting Noise

At each iteration, for each task, we search for an optimum that is non-sensitive to the update of other parameters within a fixed neighborhood. We term this kind of optima as *flat minima*.

To formally state this idea, assume that noise $\epsilon_i \sim \{\mathcal{U}(-b,b)\}^{d_{\epsilon_i}}$ with $b > 0$, $d_\epsilon = d_{\theta_i}$ and d_{θ_i} the dimension of θ_i. Then for *task 1*, at t-th iteration our target is to minimize the expected loss function with regard to the parameters (θ_1,ϕ_1) and noise ϵ_2, *i.e.,*

$$(U)\ \mathcal{R}_1^{[t]}(\theta_1,\phi_1) = \int_{\mathbb{R}^{d_{\epsilon_2}}} \mathcal{L}_1(\theta_1,\phi_1,\theta_2^{[t-1]}+\epsilon_2)\,dP(\epsilon_2) = \mathbb{E}[\mathcal{L}_1(\theta_1,\phi_1,\theta_2^{[t-1]}+\epsilon_2)], \quad (3)$$

$$s.t.\ |\theta_1 - \theta_1^{[t-1]}| < b,$$

where $P(\epsilon_2)$ is the noise distribution, and the solution is denoted as $(\theta_1^{[t]},\phi_1^{[t]})$. Similarly, for *task 2*, the loss function is as follows,

$$(L)\ \mathcal{R}_2^{[t]}(\theta_2,\phi_2) = \int_{\mathbb{R}^{d_{\epsilon_1}}} \mathcal{L}_2(\theta_2,\phi_2,\theta_1^{[t]}+\epsilon_1)\,dP(\epsilon_1) = \mathbb{E}[\mathcal{L}_2(\theta_2,\phi_2,\theta_1^{[t]}+\epsilon_1)], \quad (4)$$

$$s.t.\ |\theta_2 - \theta_2^{[t-1]}| < b.$$

Abaltion Study

Table: Performance of SOTA MTL methods on MNIST dataset. We set the number of parameters of **Joint** method as the base 1, and the values in the column 'Params' are the ratio of the parameter number of each method to the **Joint**.

Methods	Params	*Task 1*	*Task 2*
Independent	≈ 2	99.41 ± 0.03492	98.77 ± 0.06029
Ours (Vanilla)	1	99.61±0.06210	99.37±0.04494
Ours (*w/o* Reg)	1	99.66±0.03765	99.56±0.07203
MT-COOL (Ours)	1	**99.72±0.03978**	**99.62±0.01576**
Joint	1	99.60 ± 0.03765	99.51 ±0.06281
CAGrad [1]	1	99.67±0.05293	99.51±0.05229
GradDrop [2]	1	99.65± 0.03492	99.53±0.04245
MGDA [3]	1	99.63± 0.05883	99.47±0.05078
PCGrad [4]	1	99.66±0.04180	99.51±0.09108

Comparison Study

Table: Performance of SOTA MTL methods on REFUGE2018 dataset.

Methods	Params	Segmentation		Classification			
		Cup (Dice%)	Disc (Dice%)	Acc	AUROC	Sen	Spe
Independent	≈ 2	95.14±0.05110	86.87± 005644	0.900±0.00235	0.902±0.0106	0.658±0.0117	0.927±0.00392
Joint	1	91.19±0.7600	77.36±0.5236	0.907±0.0183	0.895±0.0221	0.658±0.0656	0.935±0.0264
CAGrad [1]	1	92.67±0.7702	81.71±0.2874	0.914±0.00513	0.904±0.00562	0.658±0.0235	0.942±0.00796
GradDrop [2]	1	91.70±0.6376	78.91±1.439	0.909±0.00424	0.922±0.0115	0.716±0.0471	0.930±0.00988
MGDA [3]	1	93.87±0.5017	83.87±0.9732	0.895±0.0154	0.914±0.00610	0.633±0.0824	0.924±0.0260
PCGrad [4]	1	91.74±0.5569	79.80±0.8748	0.911±0.00849	0.898±0.0136	0.675±0.0204	0.937±0.00796
MT-COOL (Ours)	1	**94.37±0.1706**	**86.18±0.3046**	**0.937±0.0113**	**0.942±0.0149**	**0.750±0.000**	**0.958±0.0126**

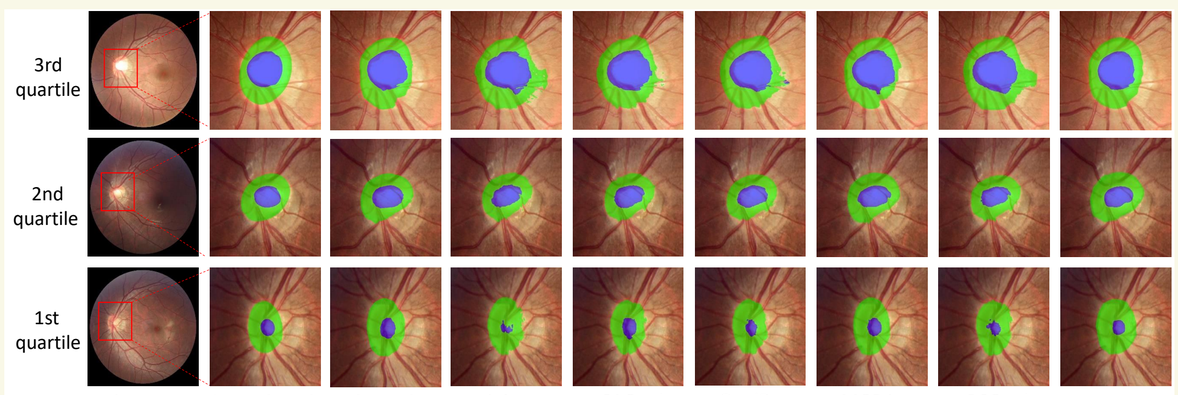

Figure: Visualization results from MTL methods on REFUGE2018 dataset. The selected samples rank the 1st quartile, median and 3rd quartile in terms of the segmentation performance of **Independent**.

Table: Performance of SOTA MTL methods on HRF-AV dataset.

Methods	Params	A/V Segmentation						Binary Segmentation	
		Acc (A)	F1 (A)	Acc (V)	F1 (V)	Acc (AV)	F1 (A/V)	Acc	F1
Independent	≈ 2	0.9814	0.6999	0.9821	0.7492	0.9692	0.7698	0.9691	0.7831
Joint	1	0.9622	0.3537	0.9661	0.5171	0.9664	0.7360	0.9691	0.7835
CAGrad [1]	1	0.9687	0.4754	0.9696	0.5520	0.9668	0.7364	0.9690	0.7790
GradDrop [2]	1	0.9708	0.5127	0.9716	0.5736	0.9666	0.7343	0.9686	0.7742
MGDA [3]	1	0.9636	0.2343	0.9632	0.5315	0.9660	0.7263	0.9691	0.7793
PCGrad [4]	1	0.9671	0.4262	0.9681	0.5387	0.9667	0.7357	0.9687	0.7763
MT-COOL (Ours)	1	**0.9801**	**0.6671**	**0.9811**	**0.7135**	**0.9674**	**0.7424**	**0.9701**	**0.7912**

Conclusion

In this work, we propose a novel MTL framework via bi-level optimization. Our method learns features for each task in a cooperative manner, instead of competing for resources with each other. We validate our model on three datasets, and the results prove its great potential in MTL.

Referencias

[1] Liu, B., Liu, X., Jin, X., Stone, P., Liu, Q.: Conflict-averse gradient descent for multi-task learning. Advances in Neural Information Processing Systems 34, 18878–18890 (2021)

[2] Chen, Z., Ngiam, J., Huang, Y., Luong, T., Kretzschmar, H., Chai, Y., Anguelov, D.: Just pick a sign: Optimizing deep multitask models with gradient sign dropout. Advances in Neural Information Processing Systems 33, 2039–2050 (2020)

[3] Sener, O., Koltun, V.: Multi-task learning as multi-objective optimization. Advances in neural information processing systems 31 (2018)

[4] Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K., Finn, C.: Gradient surgery for multi-task learning. Advances in Neural Information Processing Systems 33, 5824–5836 (2020)

Fuping Wu,Le Zhang,Yang Sun, Yuanhan Mo,Thomas Nichols,Bartlomiej W. Papiez

{Fuping.Wu,Le.Zhang}@ndph.ox.ac.uk,{yang.sun, thomas.nichols, bartlomiej.papiez}@bdi.ox.ac.uk,yuanhan.mo@ndm.ox.ac.uk

University of Oxford