Split-and-Bridge: Adaptive Class Incremental Learning within a Single Neural Network

Jong-Yeong Kim, Dong-Wan Choi

Department of Computer Science and Engineering, Inha University, South Korea

kjy93217@naver.com, dchoi@inha.ac.kr

https://github.com/bigdata-inha/Split-and-Bridge

Standard KD-based Class Incremental Learning

- **Class incremental learning**
  Learning tasks arrive in a sequence and deep neural network θ must continually learn to increment already acquired knowledge.
- **Rehearsal method**
  Store a subset of previous samples M_t, and train them together with samples of a new task D_t to prevent forgetting previous knowledge.
- **Knowledge Distillation based method**
  Try to mitigate forgetting by transferring the previous knowledge distilled from the pre-trained model.
- **Standard KD-based Method loss function**

$$\lambda\, \mathcal{L}_{kd}(\mathcal{D}_t \cup \mathcal{M}_t, \Theta_t) + (1-\lambda)\, \mathcal{L}_{ce}(\mathcal{D}_t \cup \mathcal{M}_t, \Theta_t)$$

$$\mathcal{L}_{kd}(\mathcal{D}, \Theta) = -\sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}} \hat{q}(\mathbf{x}) \log q(\mathbf{x}) \qquad \mathcal{L}_{ce}(\mathcal{D}, \Theta) = -\sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}} \mathbf{y} \log p(\mathbf{x})$$

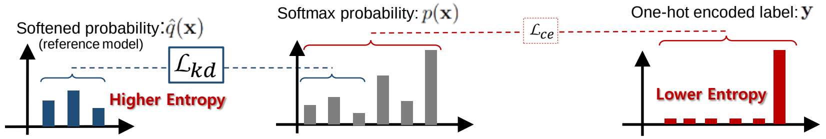
Softened probability(reference model): $\hat{q}(\mathbf{x})$; Softened probability: $q(\mathbf{x})$; Softmax probability: $p(\mathbf{x})$

Motivation

- **Our observation**: we can think of class incremental learning as the problem of learning 3 types of knowledge.

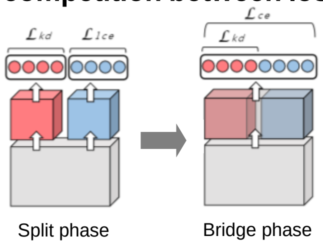
  Intra-old / Intra-new / Cross-task

- We can further identify **which part of the loss function is utilized** to acquire each type of knowledge in Standard KD-based method.
  - Intra-old knowledge: $\mathcal{L}_{kd}(\mathcal{D}_t \cup \mathcal{M}_t, \Theta_t) + \mathcal{L}_{ce}(\mathcal{M}_t, \Theta_t)$
  - Intra-new knowledge: $\mathcal{L}_{ce}(\mathcal{D}_t, \Theta_t)$
  - Cross-task knowledge: $\mathcal{L}_{ce}(\mathcal{D}_t \cup \mathcal{M}_t, \Theta_t)$
- KD-based method **suffer from learning intra-new and cross-task knowledge by CE loss**



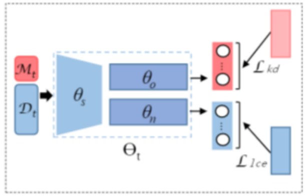
Overview

- We propose a two phase learning method within a single network to learn **without any competition between losses**



Proposed Adaptable Incremental Learning

- **Separated learning within a single network**
  To learn the intra-new knowledge as independently as possible from the task of preserving the intra-old knowledge.



Separated partition: $\Theta_t \rightarrow \langle \theta_s, [\theta_o, \theta_n] \rangle_t$ (Old partition: θ_o; New partition: θ_n)

Loss function:

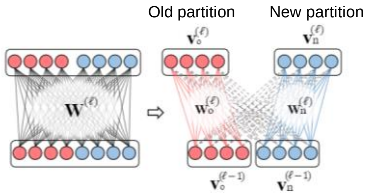
$$\mathcal{L}_{kd}(\mathcal{D}_t \cup \mathcal{M}_t, \langle\theta_s, \theta_o\rangle_t) + \mathcal{L}_{lce}(\mathcal{D}_t, \langle\theta_s, \theta_n\rangle_t)$$

Localized Cross Entropy:

$$\mathcal{L}_{lce}(\mathcal{D}_t, \Theta) = -\sum_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_t} \mathbf{y}_t \log p_t(\mathbf{x})$$

Local Softmax probability: $p_t(\mathbf{x})$

- **Weight sparsification across tasks**
  We gradually remove inter-connected weights $W_{o,n}$ and $W_{n,o}$ to get a separated network with less previous knowledge loss.

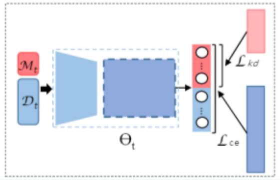


Loss function:

$$\mathcal{L}_{kd}(\mathcal{D}_t \cup \mathcal{M}_t, \Theta_t) + \mathcal{L}_{lce}(\mathcal{D}_t, \Theta_t)$$

$$+\ \gamma \sum_{\ell=S+1}^{L} \left(\|\mathbf{W}_{o,n}^{(\ell)}\|_2 + \|\mathbf{W}_{n,o}^{(\ell)}\|_2\right),$$

- **Bridge phase**
  We re-connect two partitions θ_o and θ_n in order to learn the cross-task knowledge between them.



Re-connect separated partitions:

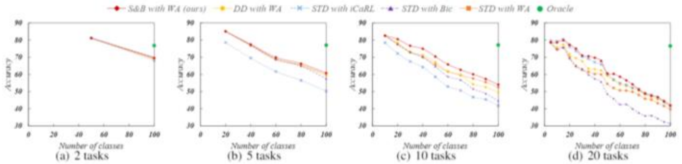
$\langle \theta_s, [\theta_o, \theta_n] \rangle_t \rightarrow \Theta_t$

connect through zero-initialized weight

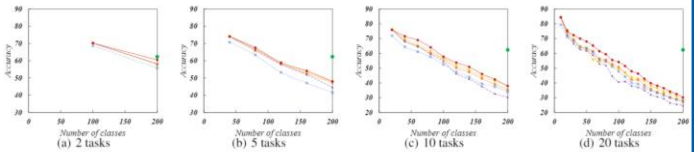
$$\lambda\, \mathcal{L}_{kd}(\mathcal{D}_t \cup \mathcal{M}_t, \Theta_t) + (1-\lambda)\, \mathcal{L}_{ce}(\mathcal{D}_t \cup \mathcal{M}_t, \Theta_t)$$

Experimental Results

- **CIFAR-100** in ResNet-18
  consistently achieves the highest accuracy



- **Tiny-ImageNet** in ResNet-18 (highest accuracy)



- Average **intra-new and intra-old accuracy**
  highest intra-new accuracy & preserving intra-old accuracy