UNIVERSITY OF CAMBRIDGE

No. 2806

Improving Feature Generalizability With Multitask Learning In Class Incremental Learning

Dong Ma[*1,2], *Chi Ian Tang*[*1], *Cecilia Mascolo*[1]

[1]University of Cambridge, [2]Singapore Management University, *Co-primary authors

Introduction & Motivation

Many deep learning applications, like keyword spotting, require the incorporation of new concepts (classes) over time, referred to as Class Incremental Learning (CIL). The major challenge in CIL is catastrophic forgetting, i.e., preserving as much of the old knowledge as possible while learning new tasks. Various techniques, such as regularization [1], knowledge distillation [2], and the use of exemplars [3], have been proposed to resolve this issue. However, as shown in Figure 1, prior works primarily focus on the incremental learning step, while ignoring the optimization during the base model training.

We hypothesis that a more transferable and generalizable feature representation from the base model would be beneficial to incremental learning. This assumption is tenable due to the fact that for a given neural network and accuracy, there exist multiple sets of weights, i.e., training a neural network multiple times with the same setting results in different sets of model weights. So, the question is how to obtain a set of weights that is more transferable to new classes.

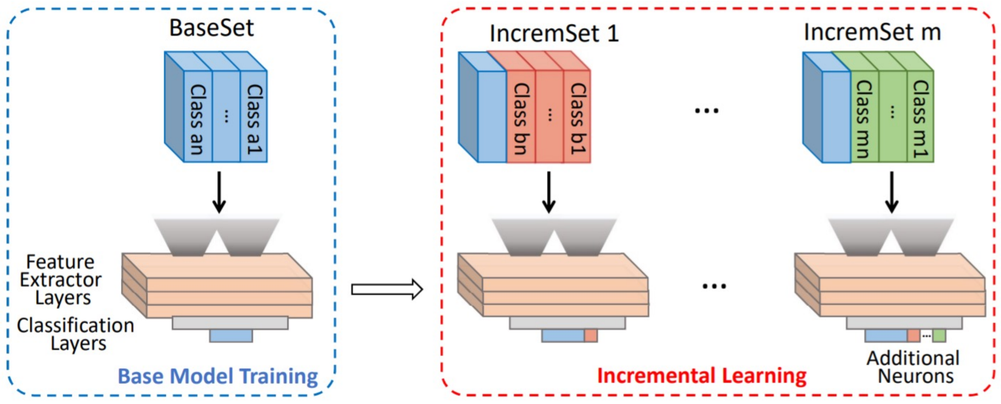

Fig. 1. Illustration of typical class incremental learning.

Our Approach

We propose the use of multitask learning during the base model training to improve the generalizability of embeddings. As shown in Figure 3, the full base dataset is decomposed into multiple subsets and they form a multitask setting, where the classification of each set is regarded as a task. These tasks are trained concurrently with a shared representation. After multitask training, the model backbone and the largest head (corresponding to full base dataset) is forwarded to the incremental learning step, where the state-of-the-art techniques can be directly applied.

Important design choices that arise from adopting multitask training are: (1) the number of tasks, and (2) the cluster of classes that each task corresponds to. In this work, we explore different choices of tasks, along two directions: (1) tasks with different number of classes (such as 5, 4, and 3 classes for each task), and (2) those with different subsets of classes but the same number of classes (such as 5, 4, 4 and 4 classes, but each task is a distinct subset).

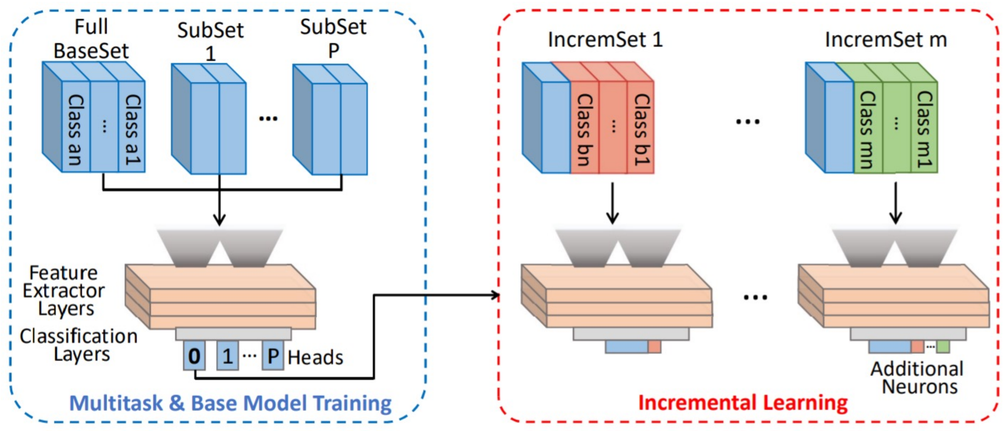

Fig. 3. Illustration of multitask learning based CIL.

Experiment Setting

- **Dataset**: Two audio datasets. UrbanSound8K dataset consists of 10 different environment sound events such as drilling, car horn, street music, etc. Google Speech Commands (GSC) is a widely used dataset in keyword spotting, from which we pick the 20 core keywords as the classes.
- **Model Architecture**: We designed a convolutional neural network to classifier the audio events. The network consists of four Conv2D layers with ReLU activation, each followed by a BatchNormalization layer. A Dropout (0.5) layer and an AveragePooling layer are connected before the final classification layer.
- **Baseline**: State-of-the-art results for CIL was achieved by Mittal et al. [4]. Specifically, [4] first utilizes the cross entropy (CE) loss and knowledge distillation (KD) loss on new classes to learn new knowledge. Then, it constructs a small but balanced exemplar set (including current incremental classes) to correct the bias and preserve old knowledge (with CE loss and KD loss)..

Results

- **Impact of Task Creation**: From Figure 4(a), we could infer that adding more tasks with the same number of classes but different subsets did not lead to consistent performance improvements. On the other hand, from Figure 4(b), we could see that increasing the number of tasks with different classes led to a consistent increase of performance, up to 5%, although a diminishing effect could be seen when we add more tasks.
- **Multitask Training Overhead**: Table 1 shows a consistent increase in training time per epoch when the number of tasks was increased, and when the number of classes in each task was increased.
- **Impact of Exemplar Quantity**: Figure 5 shows the accuracy of the model across different amounts of exemplars. We could see a general trend of improving performance as the number of exemplars was increased, with a diminishing effect. The performance of models starts to plateau with around 200 exemplars on the GSC dataset, and around 100 exemplars on US8K.
- **Comparison with Baseline**: From Figure 5, we could see that our method started with similar performance as the previous state-of-the-art method when very few exemplars were allowed and showed a consistent accuracy improvement of up to 5.5% as the amount of exemplars increases.
- **Impact of Losses**: Table 2 presents the results at different incremental steps and the average accuracy for incremental learning. We can observe that the performance gain is dominated by the use of exemplars and knowledge distillation provides limited accuracy enhancement.

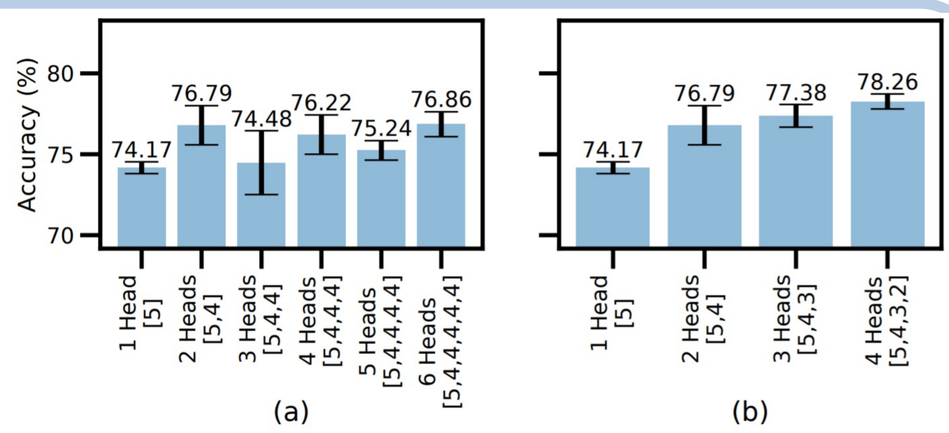

Fig. 4. Impact of task creation.

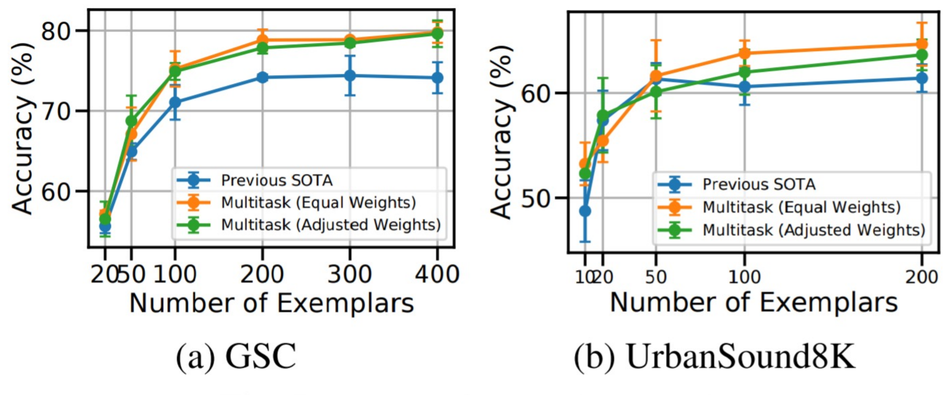

Fig. 5. Impact of exemplar quantity.

Table 1. Time overhead when training with different number and choice of tasks/heads.

Heads	Training Time per Epoch (s)	Heads	Training Time per Epoch (s)
[5]	2.16	[5,4,4]	4.79
[5,4]	3.41	[5,4,4,4]	5.35
[5,4,3]	4.65	[5,4,4,4,4]	6.44
[5,4,3,2]	5.44	[5,4,4,4,4,4]	7.42

Table 2. Effect of different losses on the incremental learning performance. CE refers to Cross Entropy, KD refers to Knowledge Distillation, N refers to New samples, and O refers to Old samples (exemplars).

# of class	5	8	11	14	17	20	Avg
CE_N	96.97	60.23	43.79	35.73	29.46	26.22	**39.09**
CE_N+KD_N	97.08	60.75	43.34	37.43	36.20	34.85	**42.52**
CE_N+CE_O	97.25	88.77	79.07	72.88	72.82	57.27	**74.16**
CE_N+CE_O +KD_N	96.78	84.65	78.27	77.91	73.60	72.55	**77.39**
CE_N+CE_O +KD_O	97.12	85.72	80.64	78.99	74.30	71.93	**78.32**
CE_N+CE_O+ KD_N+KD_O	97.35	87.92	81.47	77.66	73.80	73.27	**78.82**

Next Steps

- **Involving prior knowledge during multitask creation:** For a given set of base classes, a good construction of multiple tasks is critical for our approach. In this work, we focused on the number of classes and tasks in multitask learning. In the future, we will try to utilize prior knowledge about the dataset, such as semantic meaning, to create high-quality tasks.
- **Extending to other modalities**: Using audio datasets, this work demonstrated the effectiveness of the proposal approach for class incremental learning. Another future works is to extend the method to other modalities, such as image datasets and time series datasets, where class incremental learning is a more important application.

References

[1] Zhizhong Li and Derek Hoiem, “Learning without forgetting,” IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 12, pp. 2935–2947, 2017.

[2] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean, “Distilling the knowledge in a neural network,” arXiv preprint arXiv:1503.02531, 2015.

[3] Saurav Jha, Martin Schiemer, Franco Zambonelli, and Juan Ye, “Continual learning in sensor-based human activity recognition: an empirical benchmark analysis,” arXiv preprint arXiv:2104.09396, 2021.

[4] Sudhanshu Mittal, Silvio Galesso, and Thomas Brox, “Essentials for class incremental learning,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 3513–3522.