TCP: Triplet Contrastive-relationship Preserving for Class-Incremental Learning

Shiyao Li\textsuperscript{1}, Xuefei Ning\textsuperscript{1*}, Shanghang Zhang\textsuperscript{2}, Lidong Guo\textsuperscript{1}, Tianchen Zhao\textsuperscript{1}, Huazhong Yang\textsuperscript{1} and Yu Wang\textsuperscript{1*}  
\textsuperscript{1}Tsinghua University, China, \textsuperscript{2}Peking University, China  
lishiyao20@mails.tsinghua.edu.cn, foxdoraame@gmail.com, yu-wang@tsinghua.edu.cn

A. Experimental Setups

A.1. Datasets

We employ three datasets, which are extensively used in the literature of class-incremental learning, for our experiments: CIFAR-100, ImageNet-100 and ImageNet-1000. CIFAR-100 contains 60,000 images of the size 32×32 over 100 classes, including 50,000 training images and 10,000 test images, respectively. ImageNet-100 is a subset of ImageNet-1000 with only 100 classes, randomly sampled from the original 1000 classes and it contains about 130,000 training images and 5,000 testing images. For both datasets, we select 50 classes as the base classes, and the rest 50 classes are equally divided for incremental learning phases. ImageNet-1000 has 1000 classes, we use the first 500 classes to train the base model and the rest 5000 classes are used for incremental learning.

A.2. Training Details

All models are trained on RTX 3090 GPUs. We use ResNet-32 and ResNet-18 for CIFAR-100 and ImageNet, respectively. We add a nonlinear projection head after the ResNet\textsuperscript{2}, and remove the ReLU in the penultimate layer to allow the features to take both positive and negative values for the cosine classifier\textsuperscript{3}. We train the base 50\% classes model for 200 epochs on CIFAR-100 and ImageNet using SGD with a batch size of 256. The learning rate is initialized to 0.1 and follows a cosine annealing schedule. At each incremental learning phase, we finetune the model for 160 epochs with the memory bank $M$, batch size of 128, and new data batch size of 128. The learning rate is initially set to 0.005 for CIFAR-100 and ImageNet with the cosine annealing strategy. At the end of each incremental phase, we apply the herding sampling strategy proposed in iCaRL\textsuperscript{9} and use the data in memory bank to train a unified classifier with a learning rate 0.1 and batch size 256. Then, we evaluate the model on the union of all the encountered test datasets.

B. Hyperparameter Study

In this section, we report the ablation studies on two different hyperparameters, including the margin coefficient $\sigma$ in TCP loss, and the regularization coefficient $\alpha$ in the overall loss function. We perform all experiments on CIFAR100 under 5-phase setting.

B.1. The Effect of the Margin Coefficient

In our TCP loss, we need to choose a proper margin coefficient $\sigma$. In triplet loss\textsuperscript{11}, they regard $\sigma$ as a constant number for each triplet to learn new knowledge. However, TCP loss aims to distill old knowledge from a pretrained old model. Thus, the pretrained model can provide more in-

<table>
<thead>
<tr>
<th>margin of old model</th>
<th>constant $\sigma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ave. Acc.</td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td>66.54</td>
</tr>
</tbody>
</table>

Table 1. Average Accuracy of different margin applied in the triplet contrastive-relationship preserving loss

Figure 1. The accuracy of the first incremental step under the 5-phase setting on CIFAR100.
Table 2. Average few-shot classification accuracies (%) on CIFAR-FS and FC100 datasets.

<table>
<thead>
<tr>
<th>Model</th>
<th>backbone</th>
<th>CIFAR-FS 5-way</th>
<th>FC100 5-way</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1-shot</td>
<td>5-shot</td>
</tr>
<tr>
<td>TADAM [5]</td>
<td>ResNet-12</td>
<td>1-shot</td>
<td>69.2</td>
</tr>
<tr>
<td>Shot-Free [8]</td>
<td>ResNet-12</td>
<td>5-shot</td>
<td>70.4</td>
</tr>
<tr>
<td>TEWAM [7]</td>
<td>ResNet-12</td>
<td>1-shot</td>
<td>72.2</td>
</tr>
<tr>
<td>ProtoNet [12]</td>
<td>ResNet-12</td>
<td>5-shot</td>
<td>72.6</td>
</tr>
<tr>
<td>MetaOptNet [10]</td>
<td>ResNet-12</td>
<td>1-shot</td>
<td>73.9</td>
</tr>
<tr>
<td>RethinkDistill [13]</td>
<td>ResNet-12</td>
<td>5-shot</td>
<td>74.1</td>
</tr>
</tbody>
</table>

formative margins that indicate the sample similarities in each old triplet. For instance, if the negative sample is highly similar to the anchor sample, then the margin will be small; otherwise, the margin will be large. This additional similarity information in old model's margin will help TCP loss to preserve old knowledge, so we use the old model’s margins as \( \sigma \) instead of a constant \( \sigma \). Note that, when we compute the contrastive relationship \( D_{i,j,k} \) of new model, we simultaneously compute the contrastive relationship \( D_{i,j,k} \) of old model as the old margin \( \sigma \). As shown in Table 1, the results indicate that the average accuracy achieved by using the old model’s margins outperforms the best constant \( \sigma \) choice by 1.87%. This experiment demonstrates that incorporating the more informative margins of the old model is an effective approach to improve performance.

B.2. Regularization Coefficient Stability

The overall loss function of the distillation-based incremental learning methods can be described as Equation 1:

\[
L = L_{A2CL} + \alpha L_{\text{distill}},
\]

where \( L_{A2CL} \) and \( L_{\text{distill}} \) denote asymmetrical augmented contrastive loss and the distillation loss, \( \alpha \) denotes the regularization coefficient.

To investigate the impact of the regularization coefficient \( \alpha \), we employed the proposed A2CL to learn new data and evaluated the effectiveness of three different distillation losses in preserving old knowledge: point-wise FDL [3], pair-wise IRD [1], and our proposed TCP con.

Figure 1 displays the first phase accuracy of three distillation losses with varying values of the regularization coefficient \( \alpha \). FDL (blue line) and IRD (red line) exhibit sensitivity to the value of \( \alpha \), with appropriate values for \( \alpha \) only found within a narrow range. In contrast, TCP is less sensitive to the regularization coefficient \( \alpha \) than FDL and IRD. When using the TCP (green line) loss, the appropriate \( \alpha \) value can be chosen from a wide range (from 40 to 130), as shown in Figure 1. This outcome is expected since TCP offers greater flexibility to allow for changes in the feature space when learning new tasks. When \( \alpha \) is very large, the distillation loss carries greater weight, and the optimizer endeavors to minimize the distillation loss. In this case, FDL and IRD aim to maintain the exact value of old feature positions or similarities, leading the model to sacrifice the learning of new classes. However, using the TCP loss with a large \( \alpha \) enables the model to easily learn new classes since the TCP loss only preserves the contrastive relationship of features instead of any exact value.

C. TCP on Few-shot Learning

Few-shot scenarios are very common in real-world applications due to the long-tail distribution of data [6, 14]. In such scenarios, the model requires fine-tuning using a limited number of samples from new tasks [5, 12, 13]. We investigate whether the TCP loss can improve performance in the important few-shot scenarios by plugging it into existing few-shot learning algorithms. We report the few-shot learning ability of the proposed TCP loss on CIFAR-FS and FC100 datasets in Table 2.

The CIFAR-FS (CIFAR100 Few-shots) dataset is derived from the original CIFAR-100 by splitting 100 classes into 64, 16, and 20 classes for training, validation, and testing. The FC100 (Few-shot CIFAR100) dataset is also derived from CIFAR-100. Different from CIFAR-FS, it first groups the original 100 classes into 20 high-level classes, then it splits the 20 high-level classes into 12, 4, and 4 classes for training, validation, and testing. This results in 60 classes for training, 20 classes for validation, and 20 classes for testing.

TCP can easily be plugged into few-shot learning methods and boost their performance. As shown in Table 2, we plug the proposed TCP loss into RethinkDistill [13] and denote it as RethinkDistill+TCP. TCP can effectively boost the performance of the original RethinkDistill on both CIFAR-FS and FC100 datasets. RethinkDistill...
### D. The effect of class imbalance problem

As we mentioned in the abstract and introduction, the imbalance problem in CIL makes it difficult to preserve the feature relation of old classes and hard to learn the feature relation between old and new classes. To empirically substantiate our assertion, we devised a meticulous experiment. Specifically, we continually learn ten new classes based on a pretrained 50-class classifier on CIFAR-100 with the contrastive distillation loss [1] and evaluate the average accuracy on the 50 old classes. The findings, presented in Tab. 3, reveal the following: In case A, where all training data is utilized during continual learning (serving as our baseline), the mean accuracy of the old 50 classes stands at 77.01%. In case B, which limits the storage to merely 100 images for each old class, there’s a noticeable decline in accuracy from 77.01% to 73.45%. This case shows that the imbalance problem between old and new classes can truly make it difficult for the model to preserve the accuracy of the old classes. However, in case C, all classes have 300 training images, which means that the data in each class is balanced. Consequently, it becomes evident that the imbalance indeed impedes the model’s capacity to preserve the feature relation of old classes, making it hard for the model to learn the feature relation between old and new classes.

<table>
<thead>
<tr>
<th>ID</th>
<th>new image / class</th>
<th>old image / class</th>
<th>Avg. Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>500</td>
<td>500</td>
<td>77.01%</td>
</tr>
<tr>
<td>B</td>
<td>500</td>
<td>100</td>
<td>73.45%</td>
</tr>
<tr>
<td>C</td>
<td>300</td>
<td>300</td>
<td>75.96%</td>
</tr>
</tbody>
</table>

Table 3. Average Accuracy of the 50 old classes with different number of new images and old images per class in CIL.

In fact, the benefits of TCP loss can be leveraged in various scenarios where the model needs to effectively learn new knowledge while distilling relevant old knowledge from a teacher model. These scenarios include few-shot [12, 13], incremental [4, 9], and others. By simply integrating TCP loss into the original algorithms, we can enhance their performance.

### References


