Supplementary Material for the Paper Titled "TACLE: Task and Class-aware Exemplar-free Semi-supervised Class Incremental Learning"

Jayateja Kalla* Rohit Kumar* Soma Biswas
Department of Electrical Engineering
Indian Institute of Science, Bangalore, India.
{jayatejak,krohit,somabiswas}@iisc.ac.in

1. Effect of hyper-parameters α and β on task-wise threshold

This section analyzes the impact of hyper-parameters α and β on the task-wise adaptive threshold defined by the equation:

$$\gamma_a^{(t)} = \frac{\alpha}{1 + e^{\alpha t}} + \beta, \tag{1}$$

Figure 1 illustrates the behavior of the task-wise adaptive threshold as we vary α and β . Table 1 shows the average incremental accuracy achieved on the CIFAR-100 dataset with 0.8% labeled data per class across 10 incremental tasks.

As shown in Figure 1, the threshold value generally decreases with increasing task number (t). This aligns with the desired behavior of incorporating more unlabeled data as the number of labeled samples grows. The experiment results in Table 1 suggest that the choice of α and β impacts performance on incremental learning. For example, the configuration with $\alpha = 0.55$ and $\beta = 0.7$ leads to a lower average accuracy. This is likely due to a high threshold, which hinders the effective utilization of unlabeled data. We opted for this decaying threshold function inspired by the inverse sigmoid due to its simplicity and control over the initial and final threshold values. This allows for a smooth decrease in the threshold as tasks progress, enabling the model to leverage more unlabeled data effectively over time. The use of an inverse sigmoid function for adaptive thresholding is not a strict constraint. Instead, we can employ any mathematical decay function that adheres to the desired structure.

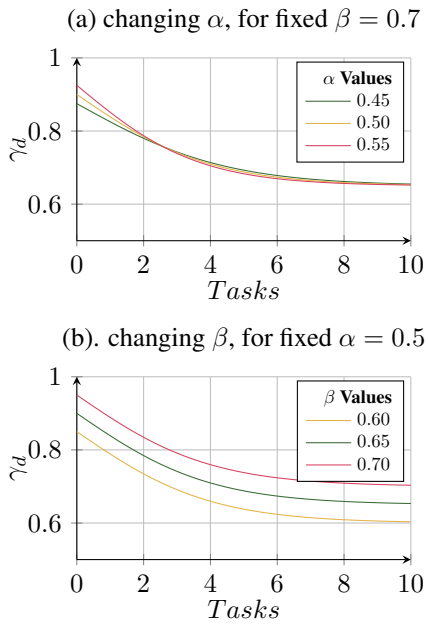


Figure 1. Task-wise adaptive threshold output values by changing hyper-paraneters (α, β).

Table 1. Impact of threshold hyper-parameters α and β on CIFAR100 dataset.

$\alpha \downarrow$ \ $\beta \rightarrow$	0.60	0.65	0.70
0.45	92.12	91.78	92.07
0.50	91.96	**92.35**	91.01
0.55	91.86	91.01	90.52

2. Algorithm

The complete summarization algorithm for TACLE for Exemplar-free Semi-supervised CIL is provided in Algorithm 1. It consists of two stages: (i) feature representation learning, and (ii) classifier alignment.

3. Task-wise cumulative accuracy results

In this section, we report the task-wise cumulative accuracy results for the proposed approach TACLE, SLCA, and SLCA+Fixed threshold. Figure 2 presents the results for CIFAR100 with 0.8% and 5% labeled data settings for the EFSS-CIL protocol. We also report the average incremen-

Algorithm 1: TACLE for semi-supervised class incremental learning

Input: $\{\Theta, \psi\} \leftarrow$ Model; $\{\mathcal{D}^{(1)}, \mathcal{D}^{(2)}, \ldots, \mathcal{D}^{(\mathcal{T})}\} \leftarrow$ Data stream;
$E_{s1} \leftarrow$ No. of epochs for stage 1; $E_{s2} \leftarrow$ No. of epochs for stage 2;
for $t \leftarrow 1$ *to* $\mathcal{T}$ **do**
  $\mathcal{D}_l^{(t)} = \{\mathbf{x}_i^l, y_i^l\}_{i=1}^{N_l^{(t)}}; \mathcal{D}_{ul}^{(t)} = \{\mathbf{x}_i^{ul}\}_{i=1}^{N_{ul}^{(t)}};$
  $\zeta \leftarrow$ Uniform distribution across all classes
  // #Stage 1: Feature Representation Learning //
  for $e_{s1} \leftarrow 1$ *to* E_{s1} **do**
    $\mathcal{B}_l = \text{SampleMiniBatch}(\mathcal{D}_l^{(t)}); \mathcal{B}_{ul} = \text{SampleMiniBatch}(\mathcal{D}_{ul}^{(t)});$
    $\hat{\mathcal{B}}_{ul} = \text{ImageAugmentations}(\mathcal{B}_{ul});$
    $\mathcal{O}_l, \mathcal{O}_{ul}, \hat{\mathcal{O}}_{ul} = \Theta(\psi^{(t)}(\mathcal{B}_l, \mathcal{B}_{ul}, \hat{\mathcal{B}}_{ul}));$
    $w^l \leftarrow$ Assigning class-aware weights for labeled data $\mathcal{B}_l$ using $\bar{\zeta}$;
    $w^{ul} \leftarrow$ Assigning class-aware weights for unlabeled data $\mathcal{B}_{ul}$ using $\bar{\zeta}$;
    $\mathcal{L}_{stage1} \leftarrow \mathcal{L}_s(\mathcal{B}_l) \cdot w^l + \mathcal{L}_{us}(\hat{\mathcal{B}}_{ul}) \cdot w^{ul};$ `// Total loss for stage1`
    $\zeta \leftarrow$ Update the histogram distribution using $\mathcal{D}_{ul}^{(t)}, \gamma_a^{(t)};$
    $\bar{\zeta} \leftarrow (2 - \zeta);$ `// Normalization`
    $\{\Theta, \psi^{(t)}\} \leftarrow$ Update model parameters using $\mathcal{L}_{stage1}$;
  //#Stage 2: Classifier Alignment//
  $\tilde{\mathcal{D}}^{(t)} \leftarrow$ Expanded labelled data set using $\mathcal{D}_l^{(t)}, \mathcal{D}_{ul}^{(t)}, \gamma_a^{(t)};$
  $\tilde{\mu}_k^{(t)}, \tilde{\Sigma}_k^{(t)} \leftarrow$ Estimate mean and variance using $\tilde{\mathcal{D}}^{(t)};$ `// where` $k \in 1, 2, ., |\mathcal{C}^{(t)}|$
  for $e_{s2} \leftarrow 1$ *to* E_{s2} **do**
    $\mathcal{L}_{stage2} \leftarrow \mathcal{L}_{ca}(\tilde{\mu}_k^{(1:t)}, \tilde{\Sigma}_k^{(1:t)});$ `// Alignment loss for classifiers`
    $\psi^{(1:t)} \leftarrow$ Update classifier parameters using $\mathcal{L}_{stage2}$;

tal accuracy at the end of the task for both cases where two different pre-trained models are used for model weight initialization. The proposed TACLE outperforms the baselines by a significant margin in the both the scenarios.

4. Challenging Scenarios

4.1. One-shot EFSS-CIL

Fig. 3 depicts the performance of different methods in the one-shot EF-SSCIL setting for the ImageNet-Subset100 dataset. In this setting, each class has only one labeled data point along with unlabeled data, hence it is referred to as the one-shot EF-SSCIL protocol. MoCo v3 pre-trained ViT is used for weight initialization in these experiments. The ImageNet-Subset 100 dataset is divided into 20 tasks, with each task containing 5 classes. Therefore, the number of labeled and unlabeled samples per task is 5 and 6500, respectively. Our method (TACLE) achieves a 8.75% higher accuracy compared to the SLCA method on this challenging setting.

4.2. Imbalance EFSS-CIL

Fig. 4a illustrates the data distribution in the standard SS-CIL setting, where the unlabeled data from every class is balanced, meaning the number of samples from all classes is equal in the unlabeled data (in the standard setting, they have access to exemplars also but we are not showing for simplicity). Conversely, Fig. 4b shows the data distribution for the imbalance EFSS-CIL proposed in the paper. In this scenario, we have a highly skewed distribution for the unlabeled data, with an imbalance ratio of 0.01, indicating that the ratio between the class with fewer samples and the class with more samples is 0.01. At every task, unlabeled data follows this imbalance (head-tail) distribution.

4.3. Training optimization details

During training, stage 1 for each task is trained for 10 epochs. A learning rate schedule is employed, reducing the learning rate by a factor of 10 after the 8^{th} epoch. To facilitate stable initial convergence, the network is first warmed up for a few iterations using only labeled data loss. Subsequently, unlabeled data losses are incorporated and added to the total loss function. The standard SGD optimizer with a batch size of 128 is employed for both CIFAR-10 and CIFAR-100 experiments. Due to GPU memory limitations, a reduced batch size of 64 is used for the ImageNet-subset100 experiments.

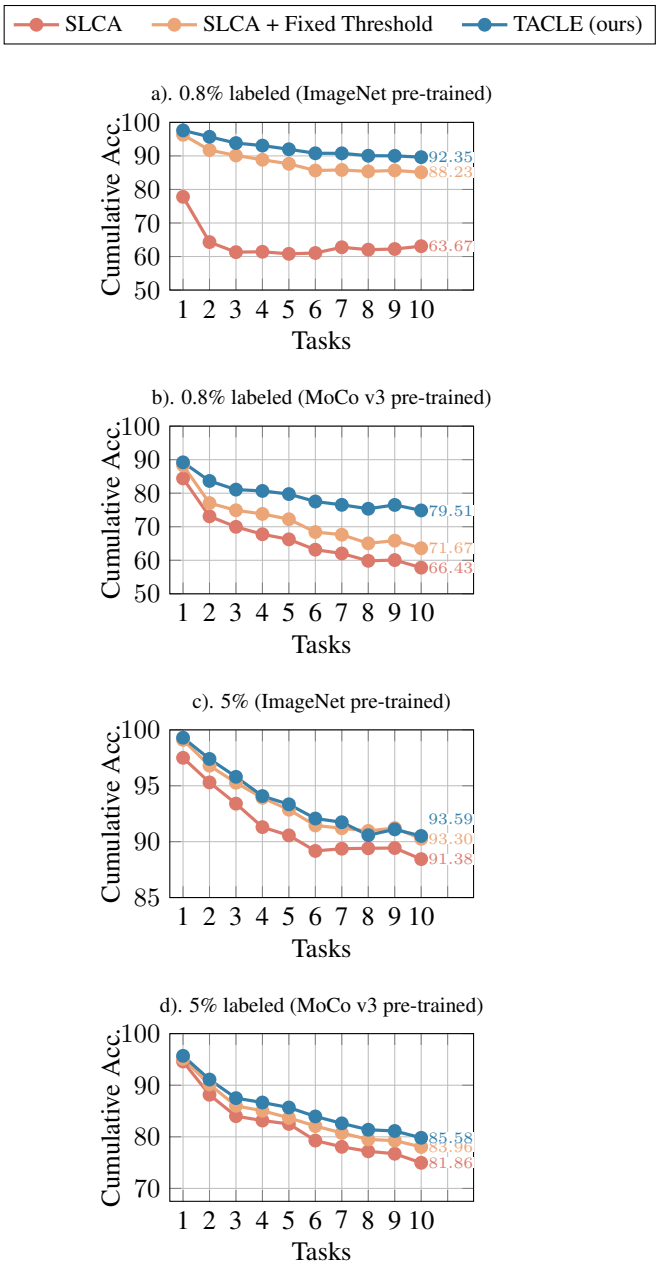


Figure 2. Analysis for CIFAR100 datasets for different methods. Experiments were conducted for 0.8% and 5% labeled data with 10 tasks, reporting top-1 cumulative accuracy at the end of each task and average cumulative accuracy at the end of each plot. Results are presented for both pre-trained models.

5. Additional Base-lines

This section examines TACLE's performance in comparison with additional baseline methods. We first evaluated FeCAM, an exemplar-based class incremental learning approach that relies on class statistics (mean, variance) for alignment. We found that vanilla FeCAM was not effective

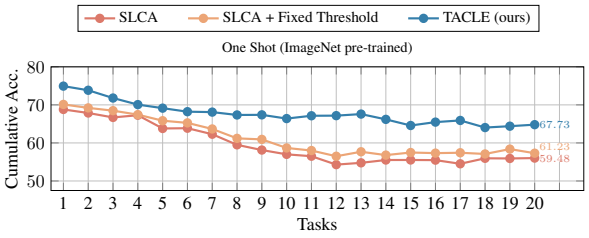
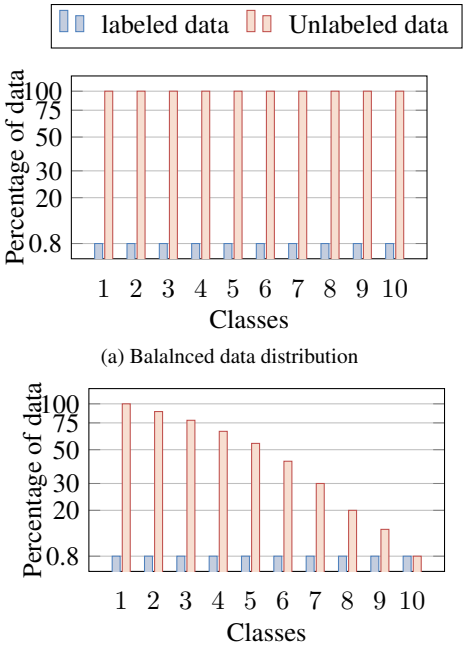


Figure 3. Evaluation of One-Shot Performance on ImageNet-100 with MoCo v3 Initialization. The experiment uses 1 labeled sample and 1300 unlabeled samples per class. The 100 classes divided into 20 tasks with 5 classes per task.



(a) Balalnced data distribution

(b) Imbalalnced data distribution

Figure 4. The bar graph illustrates the data distribution for the balanced and imbalanced unlabeled data per class-wise in the CIFAR100 dataset with 0.8% labeled data.

Table 2. Average incremental accuracy on CIFAR100 with additional base-lines (‡ indicates ImageNet-12k pretrained ViT).

#	**Method**	0.8%	5%	25%
1	FeCAM‡	23.08	40.23	54.78
2	FeCAM‡ + Utilizing Unlabel data	18.06	33.81	52.0
3	SLCA‡	63.67	91.38	93.69
4	SLCA‡+FlexMatch	90.71	92.32	93.98
5	TACLE‡ (ours)	92.35	93.59	94.10

for exemplar-free semi-supervised class incremental learning due to its inability to utilize unlabeled data effectively. Even attempts to integrate unlabeled data into class statistics did not yield improvements, as the model struggled with accurate pseudo-label generation. For SLCA, we incorpo-

rated FlexMatch, a pseudo-labeling approach that provides class-wise thresholds, offering a more refined method than FixMatch. Table 2 presents these experimental results on the CIFAR-100 dataset with varying labeled data percentages. TACLE consistently outperforms all baseline methods across different scenarios.

6. Visualization of features: SLCA vs TACLE (task 1,5,9)

To visualize the clustering of unlabeled and labeled data, we employ t-SNE dimensionality reduction on the image features extracted from the model feature extractor (Θ), which shares parameters across all tasks. We consider 4 labeled data points from each class, one class prototype for each, and all the task's unlabeled data (this is the data samples in CIFAR100 with 0.8% at every task). Figures 5 and 6 depict t-SNE plots for both the SLCA approach (which utilizes only labeled data) and our TACLE framework after tasks 1, 5, and 10. These plots consider two pre-trained models for initial model weight initialization: ImageNet and MoCo v3. We observe that, by leveraging unlabeled data, proposed TACLE achieves better clustering and learns superior feature representations, thereby enhancing the overall performance of EFSS-CIL.

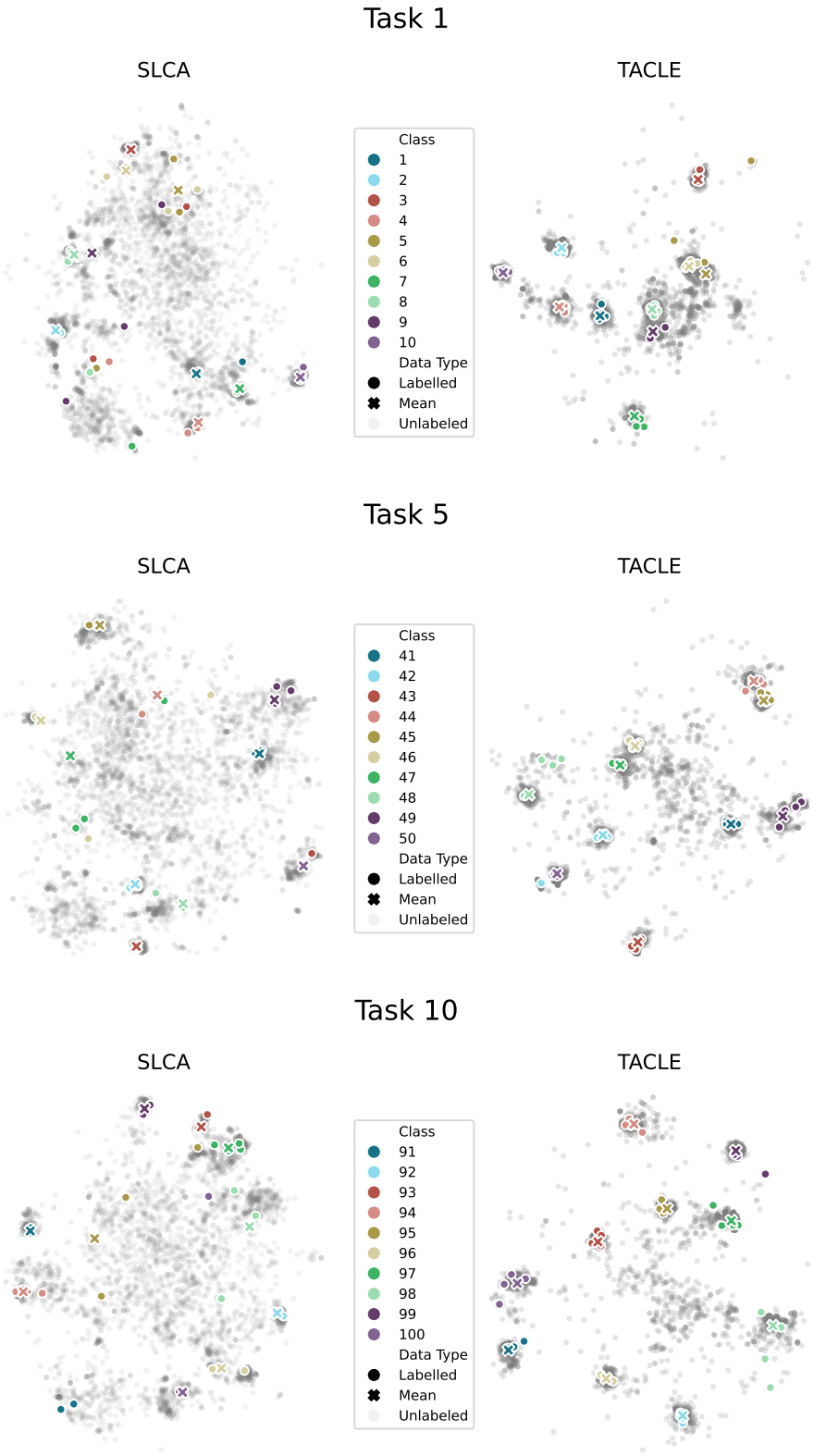


Figure 5. t-SNE visualization of SLCA vs TACLE for given task id 1, 5, and 10. Each point represents image feature vector of dimension 768 (using ImageNet as pre-trained model).

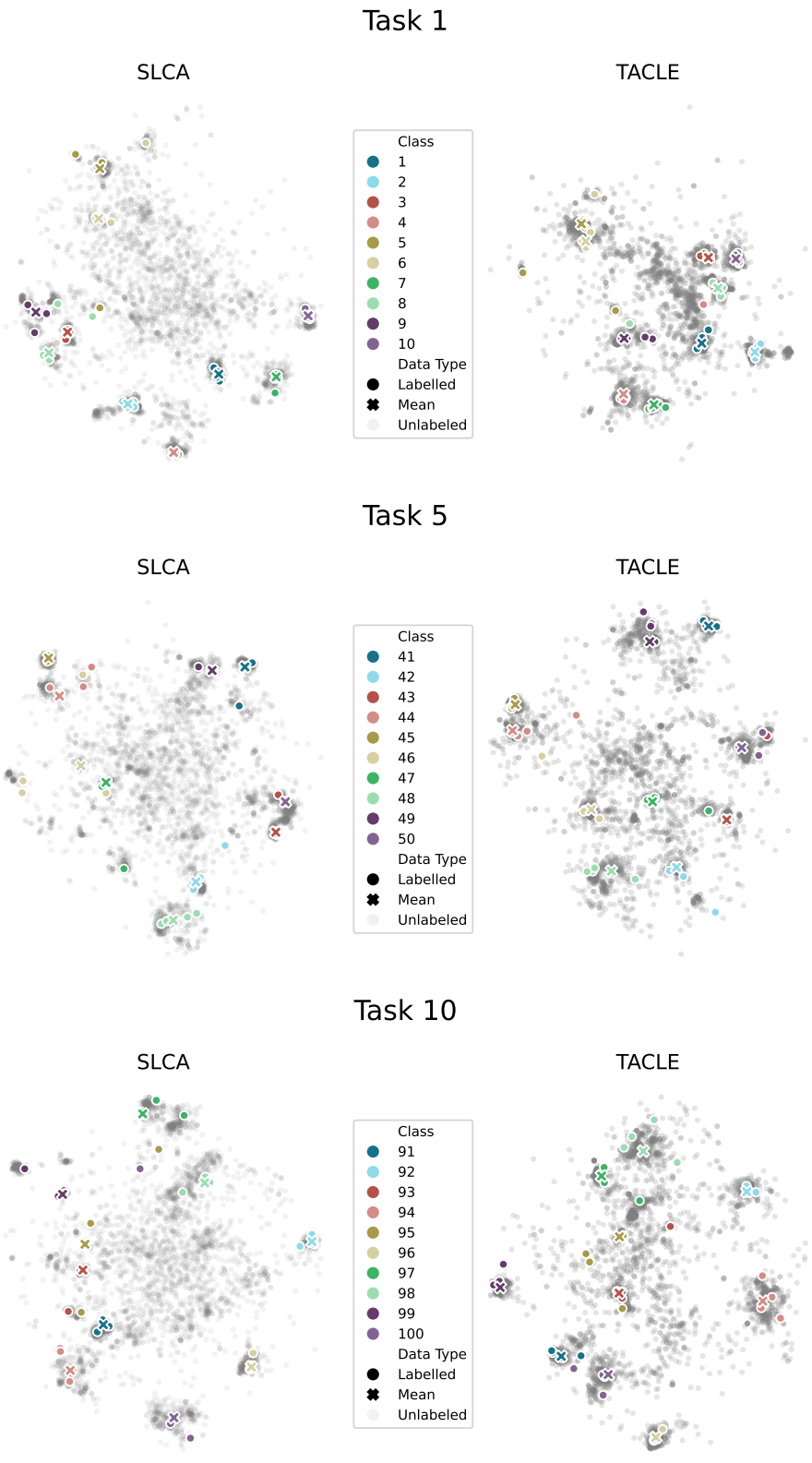


Figure 6. t-SNE visualization of SLCA vs TACLE for given task id 1, 5, and 10. Each point represents image feature vector of dimension 768 (using moco V3 as pre-trained model).