Provable Contrastive Continual Learning

Yichen Wen [*1,2] **Zhiquan Tan** [*3] **Kaipeng Zheng** [1] **Chuanlong Xie** [2] **Weiran Huang** [†1,4]

Abstract

Continual learning requires learning incremental tasks with dynamic data distributions. So far, it has been observed that employing a combination of contrastive loss and distillation loss for training in continual learning yields strong performance. To the best of our knowledge, however, this contrastive continual learning framework lacks convincing theoretical explanations. In this work, we fill this gap by establishing theoretical performance guarantees, which reveal how the performance of the model is bounded by training losses of previous tasks in the contrastive continual learning framework. Our theoretical explanations further support the idea that pre-training can benefit continual learning. Inspired by our theoretical analysis of these guarantees, we propose a novel contrastive continual learning algorithm called CILA, which uses adaptive distillation coefficients for different tasks. These distillation coefficients are easily computed by the ratio between average distillation losses and average contrastive losses from previous tasks. Our method shows great improvement on standard benchmarks and achieves new state-of-the-art performance.

1. Introduction

Incrementally learning a sequence of tasks with dynamic data distributions is a typical setting for continual learning. We call the learned neural networks "continual learners". The main challenge for continual learners is to obtain a suitable trade-off between learning plasticity and memory stability. Specifically, excessive focus on learning plasticity of new tasks often leads to greatly reduced performance on old tasks (McClelland et al., 1995), which is known as catastrophic forgetting.

To address the challenge, the literature on continual learning has proposed various approaches. Representation-based approaches take advantage of representations. As one of these approaches, self-supervised learning with contrastive loss has demonstrated notable efficacy in obtaining robust representations against catastrophic forgetting in continual learning (Gallardo et al., 2021; Fini et al., 2022). For these methods based on contrastive loss, the training of representations is often decoupled with the training of the classifier, unlike methods based on cross-entropy. Specifically, contrastively trained representations suffer less catastrophic forgetting than ones trained by cross-entropy loss (Cha et al., 2021). Replay-based approaches use buffers to restore a part of previous data, and train networks using data from a combination of the current task and the buffer (Lopez-Paz & Ranzato, 2017). Naturally, these methods are combined with knowledge distillation strategies to prevent the degradation of information in the network over time (Rebuffi et al., 2017). Regularization-based approaches introduce regularization terms to the target loss for continual learning to reach a balance between learning new tasks and preserving information from old tasks (Kirkpatrick et al., 2017). Two main sub-directions within regularization-based approaches include weight regularization (Ritter et al., 2018) and function regularization (Li & Hoiem, 2016).

To achieve effective continual learning, a natural idea is to combine the three approaches above, and this idea leads to a new framework called contrastive continual learning (Cha et al., 2021), as illustrated in Figure 1. This framework focuses on using contrastively learned representations to learn new tasks and utilizing knowledge distillation to preserve information from past tasks, with the help of memory buffer and function regularization. The target loss of this framework contains a contrastive loss and a distillation loss with a distillation coefficient λ. The training data will be selected from the combination of the current data and buffered data. Empirically, this framework has been observed to be efficient, showing promising performance in continual learning (Cha et al., 2021). Despite the growing attention directed towards this framework, limited theoretical works have been proposed to explain its superior performance.

[*]Equal contribution [1]MIFA Lab, Qing Yuan Research Institute, SEIEE, Shanghai Jiao Tong University [2]Beijing Normal University [3]Department of Mathematical Sciences, Tsinghua University [4]Shanghai AI Laboratory. [†]Correspondence to: Weiran Huang <weiran.huang@outlook.com>. This work was conducted during the period when the first two authors were visiting MIFA Lab.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

In this paper, we try to address the theoretical problem of why this framework is efficient. Therefore, we consider the losses provided in (Cha et al., 2021). We have found a clear relationship between the contrastive losses of two consecutive models in continual learning. Inspired by this, we propose theoretical performance guarantees that reveal how the population test loss, i.e., the total performance of the final model on all seen tasks, is bounded by the series of training losses for the contrastive continual learning framework. Based on our theory, we propose a new and efficient contrastive continual learning algorithm called CILA, which uses distillation coefficients adapted to different tasks. Moreover, CILA consistently outperforms all baselines in different scenarios, datasets, and buffer sizes, e.g., about 1.77% improvement compared with the previous state-of-the-art method Co^{2}L (Cha et al., 2021) on Seq-CIFAR-10 with a buffer of 500 samples for Class-IL scenario.

Overall, our contributions are listed as follows. (1) We provide theoretical performance guarantees for the contrastive continual learning scheme. We identify that the overall performance of the final learned model on all seen tasks can be bounded by a function of the series of training losses with the distillation coefficient; (2) We propose an efficient algorithm CILA, which uses adaptive distillation coefficient λ_t (replace λ with λ_t in Figure 1) for each task t; (3) We conduct extensive experiments to validate the efficacy of our algorithm, and the results strongly support our theory. Our method can inspire future works in contrastive continual learning.

2. Related Work

Continual learning. Continual learning is also referred to as incremental learning, which learns incremental tasks effectively (Wang et al., 2023). The literature in this field mainly focuses on several streams including weight and function regularization (Jung et al., 2020), memory replay (Prabhu et al., 2020), sparse representations (Javed & White, 2019), parameter isolation (Gurbuz & Dovrolis, 2022), and dynamic architecture (Ramesh & Chaudhari, 2021).

As one of these effective continual learning methods, replay-based methods have demonstrated superior performance in terms of both learning plasticity and memory stability (Riemer et al., 2019). Replay-based continual learning methods are developed from the idea of Experience Replay (Buzzega et al., 2020), which typically stores past training samples in a fixed-size buffer. Currently, these replay-based methods are divided into two main streams, including experience replay and generative replay. Experience replay-based methods focus on the construction of memory buffer (Riemer et al., 2019; Tiwari et al., 2022) and storage efficiency (Caccia et al., 2019; Bang et al., 2021). Generative replay-based methods concentrate on generative adversarial networks (GANs) to generate fine-grained data (Cong et al., 2020; Ayub & Wagner, 2021).

Representation-based methods for continual learning are also observed to be competitive. Recent works in continual learning take advantage of self-supervised learning to obtain robust representations, showing great performance on downstream tasks (Pham et al., 2021). Large-scale pre-training also contributes to improving transferable and robust representations for downstream continual learning (Gallardo et al., 2021; Ramasesh et al., 2022).

Regularization-based methods mainly focus on weight and function regularization. Weight regularization methods add penalties to the loss function, typically the penalty is a quadratic one (Kirkpatrick et al., 2017; Liu et al., 2018), and function regularization methods implement knowledge distillation on the intermediate or final output of the prediction function (Li & Hoiem, 2017; Lee et al., 2019). For function regularization methods, the teacher model is the frozen past model, and the student model is the current model. Besides, there are some theory works analyzing regularization-based methods. Evron et al. (2022) study the minimum norm estimator in CL under an over-parameterized and noise-free setup. Li et al. (2023) give a fixed design analysis of continual ridge regression for two-task linear regression. Zhao et al. (2024) consider a family of generalized ℓ_2-regularization estimators and give some optimality analysis.

Contrastive learning. Contrastive learning aims to learn representations that attract different views of the same image while repelling views from different images (Tian et al., 2020). Contrastive methods have been widely used in self-supervised learning and pre-training, showing superior performance on downstream tasks. The contrastive loss was first proposed in (Bromley et al., 1993) and then more formally defined in (Chopra et al., 2005) and (Hadsell et al., 2006). Later some theoretical analyses on the contrastive learning framework were provided in (Arora et al., 2019; Huang et al., 2023; Tan et al., 2023b;c;a; Zhang et al., 2023). There are various target losses in contrastive learning, for example, InfoNCE loss (van den Oord et al., 2019) is a widely adopted and efficient one. Notably, methods in this field have reached or even outperformed supervised learning methods (Khosla et al., 2020) for image classification. Representative approaches include SimCLR (Chen et al., 2020a), MoCo v1&v2 (He et al., 2020; Chen et al., 2020b). In this work, we employ the contrastive loss provided in (Khosla et al., 2020) for the contrastive continual learning framework to show some theoretical insights.

Knowledge distillation. In various scenarios of continual learning, knowledge distillation is used to preserve information from the old model to the current model, contributing to mitigate catastrophic forgetting. Typically, knowl-

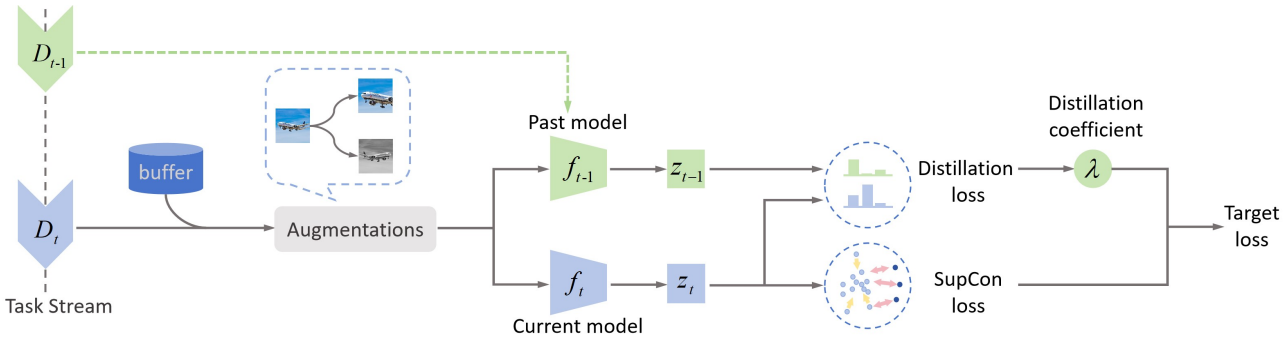

Figure 1. An illustration of contrastive continual learning framework. At the end of the previous task, we restore the previous model and values of losses. For the current task, augmentations are applied to both the buffered and the current data. Then the augmented data is passed through the current model and the previous frozen model to obtain representations. The target loss of contrastive continual learning is a weighted sum of contrastive loss and distillation loss with a distillation coefficient λ.

edge distillation learns a small student model from a large teacher model with limited resources (Gou et al., 2021). Various kinds of knowledge can be transferred by knowledge distillation, including response-based knowledge which is the neural response of the last output layer of the teacher model (Hinton et al., 2015), feature-based knowledge like feature maps (Zagoruyko & Komodakis, 2016) and relation-based knowledge referring to relationships between different layers or between different samples (Passalis et al., 2020). Learning schemes of knowledge distillation include three streams, they are online distillation, offline distillation, and self-distillation. Among them, self-distillation considers the same structure between the teacher model and the student model (Zhang & Sabuncu, 2020; Mobahi et al., 2020).

3. Problem Setup

We are given a sequence of T supervised tasks, with each task presented sequentially, one after the other. For each task t, the training samples are assumed to be drawn from an unknown data distribution $\mathcal{D}_t$. The model can be updated after seeing each task. The goal of continual learning is to train a model f that performs well over all seen tasks.

Supervised contrastive loss (Khosla et al., 2020) has shown its superiority over the cross-entropy loss in the standard supervised classification, and it is then introduced into the continual learning by Co^{2}L (Cha et al., 2021). Specifically, contrastive continual learning updates the model at each time step t according to two losses, the contrastive loss and the distillation loss, which measure the learning plasticity and memory stability, respectively.

Contrastive loss. For each task t, we use μ_t to denote the class distribution of task $\mathcal{D}_t$, and $\mathcal{D}_c$ to denote the data distribution associated with each class c. In this paper, we consider a contrastive loss involving two similar samples x, x^+ i.i.d. drawn from the same class distribution $\mathcal{D}_c$. Meanwhile, there are several negative samples randomly picked from the whole data distribution $\mathcal{D}_t$. For simplicity, we only consider the case of one negative sample here, the case of multiple negative samples can be found in Appendix D and E. Therefore, the contrastive loss can be formulated as

$$L_{\text{con}}(f;\mathcal{D}_t) = \mathop{\mathbb{E}}_{\substack{c^+\sim\mu_t \\ c^-\sim\mu_t}} \mathop{\mathbb{E}}_{\substack{x,x^+\sim\mathcal{D}_{c^+} \\ x^-\sim\mathcal{D}_{c^-}}} \ell\left[f(x)^\top(f(x^+)-f(x^-))\right],$$

where function $\ell(v)$ is defined as $\log(1+\exp(-v))$ and embeddings are conventionally normalized, i.e., $\|f\| = 1$.

Distillation loss. Continual learning focuses on retaining previously acquired information while simultaneously learning new knowledge. In the specific context of contrastive continual learning, the model achieves knowledge preservation by keeping the model's ability to differentiate between similar and dissimilar (negative) samples. To do so, we first compute the similarity probability distribution as $\boldsymbol{p}(f;x,x^+,x^-) = \text{softmax}(f(x)^\top f(x^+), f(x)^\top f(x^-))$, and then regulate the cross-entropy between the past similarity probability distribution and the current one (e.g., IRD loss in (Cha et al., 2021)). Specifically, for task t, we denote the distribution of all seen data by $\mathcal{D}_{1:t-1} := \sum_{j=1}^{t-1} k_{tj}\mathcal{D}_j$, where we allow different tasks have different weights $k_{tj} > 0$ with $\sum_{j=1}^{t-1} k_{tj} = 1$. Therefore, the distillation loss considered in this paper can be formulated as

$$L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}_{1:t-1}) = \mathop{\mathbb{E}}_{\substack{c^+\sim\mu_{1:t-1} \\ c^-\sim\mu_{1:t-1}}} \mathop{\mathbb{E}}_{\substack{x,x^+\sim\mathcal{D}_{c^+} \\ x^-\sim\mathcal{D}_{c^-}}} \left[-\boldsymbol{p}(f_{t-1};x,x^+,x^-)\cdot\log\boldsymbol{p}(f_t;x,x^+,x^-)\right],$$

where $\mu_{1:t-1}$ represents the class distribution of $\mathcal{D}_{1:t-1}$.

The total training loss of f_t on task $t \geq 2$ is

$$L_{\text{train}}(f_t; f_{t-1}, \mathcal{D}_t, \mathcal{D}_{1:t-1}) = L_{\text{con}}(f_t; \mathcal{D}_t) + \lambda \cdot L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}_{1:t-1}),$$

where λ is a hyper-parameter for balancing the two loss terms. For the first task $t = 1$, the training loss does not have the distillation term, i.e., $L_{\text{train}}(f_1; \mathcal{D}_1) = L_{\text{con}}(f_1; \mathcal{D}_1)$.

To evaluate the contrastive continual learning model, we use the total performance (test loss) of the final model f_T on all seen tasks, which can be formulated as

$$L_{\text{test}}(f_T; \mathcal{D}_1, \ldots, \mathcal{D}_T) := \sum_{t=1}^{T} L_{\text{con}}(f_T; \mathcal{D}_t).$$

4. Theoretical Analysis

Contrastive continual learning has demonstrated strong performance in practice. The focus of this paper is to examine its performance guarantees theoretically. In particular, our study aims to investigate the relationship between the test loss $L_{\text{test}}(f_T; \mathcal{D}_1, \ldots, \mathcal{D}_T)$ and the series of training losses $L_{\text{train}}(f_1; \mathcal{D}_1)$, $L_{\text{train}}(f_2; f_1, \mathcal{D}_2, \mathcal{D}_1)$, …, $L_{\text{train}}(f_T; f_{T-1}, \mathcal{D}_T, \mathcal{D}_{1:T-1})$.

According to the definition, despite the distillation loss terms, the training losses involve $\{L_{\text{con}}(f_t; \mathcal{D}_t)\}_{t=1}^T$, while the test loss consists of $\{L_{\text{con}}(f_T; \mathcal{D}_t)\}_{t=1}^T$. To bridge the test loss and the training losses, we first provide the relationship between the contrastive losses of two consecutive models f_t and f_{t-1} in the following lemma.

Lemma 1. *When $t \geq 2$, for any data distribution $\mathcal{D}$, the contrastive losses of current model f_t and previous model f_{t-1} can be connected via the distillation loss, i.e.,*

$$L_{\text{con}}(f_t; \mathcal{D}) \leq \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) + \beta,$$
$$L_{\text{con}}(f_t; \mathcal{D}) \geq \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) + \beta',$$

where $\alpha = \frac{2e^2}{1+e^2}$, $\beta = 2 - \alpha + \alpha \log \frac{\alpha}{2}$, and $\beta' = -\alpha \log(1 + e^2) - \alpha$.

The above lemma can be directly proved using the formulae for contrastive loss and distillation loss. A detailed proof is provided in the appendix due to the space limitation.

According to Lemma 1, when considering $\mathcal{D} = \mathcal{D}_t$ ($t \leq T$), a connection between $L_{\text{con}}(f_T; \mathcal{D}_t)$ and $L_{\text{con}}(f_{T-1}; \mathcal{D}_t)$ can be established. Similarly, a link between $L_{\text{con}}(f_{T-1}; \mathcal{D}_t)$ and $L_{\text{con}}(f_{T-2}; \mathcal{D}_t)$ can be drawn, and so on. This approach allows us to build a bridge between $L_{\text{con}}(f_T; \mathcal{D}_t)$ and $L_{\text{con}}(f_t; \mathcal{D}_t)$ for any given t, which are the components of test loss and training losses, respectively. Thus, with Lemma 1, we can now derive the relationship between the test loss $L_{\text{test}}(f_T; \mathcal{D}_1, \ldots, \mathcal{D}_T)$ and the series of training losses $L_{\text{train}}(f_1; \mathcal{D}_1)$, $L_{\text{train}}(f_2; f_1, \mathcal{D}_2, \mathcal{D}_1)$, …, $L_{\text{train}}(f_T; f_{T-1}, \mathcal{D}_T, \mathcal{D}_{1:T-1})$. Our results are presented in the following main theorem.

Theorem 1. *For the contrastive continual learning involving $T \geq 2$ tasks, the test loss of the final model f_T can be bounded via a linear combination of the training losses associated with each task. More specifically, the following two bounds are applicable.*

(1) Upper bound:

$$L_{\text{test}}(f_T; \mathcal{D}_1, \ldots, \mathcal{D}_T) \leq \alpha^{T-1} L_{\text{train}}(f_1; \mathcal{D}_1) + \sum_{t=2}^{T} \frac{\alpha^{T-t}}{\gamma_t(\lambda)} L_{\text{train}}(f_t; f_{t-1}, \mathcal{D}_t, \mathcal{D}_{1:t-1}) + \eta,$$

(2) Lower bound:

$$L_{\text{test}}(f_T; \mathcal{D}_1, \ldots, \mathcal{D}_T) \geq \alpha^{T-1} L_{\text{train}}(f_1; \mathcal{D}_1) + \sum_{t=2}^{T} \frac{\alpha^{T-t}}{\gamma'_t(\lambda)} L_{\text{train}}(f_t; f_{t-1}, \mathcal{D}_t, \mathcal{D}_{1:t-1}) + \eta',$$

where

$$\begin{cases} \alpha = \frac{2e^2}{1+e^2}, \\ \gamma_t(\lambda) = \min\left(\{\frac{1}{t}\} \cup \{\lambda k_{tj}\}_{j=1}^{t-1}\right), \\ \gamma'_t(\lambda) = \max\left(\{1\} \cup \{\lambda k_{tj}\}_{j=1}^{t-1}\right), \\ \eta = (2 - \alpha + \alpha \log \frac{\alpha}{2}) \frac{T-1-T\alpha+(\alpha)^T}{(1-\alpha)^2} \\ \quad + \sum_{t=2}^{T} \alpha^{T-t}(1 - \frac{1}{\gamma_t(\lambda)}) \min_f L_{\text{con}}(f; \mathcal{D}_t), \\ \eta' = -(\alpha \log(1 + e^2) + \alpha) \frac{T-1-T\alpha+(\alpha)^T}{(1-\alpha)^2}. \end{cases}$$

The proof for the theorem can be found in the appendix. It can be concluded from Theorem 1 that, the performance of the final model f_T on all T tasks, namely $L_{\text{test}}(f_T; \mathcal{D}_1, \ldots, \mathcal{D}_T)$, can be well bounded by training losses on all seen tasks, suggesting that minimizing $L_{\text{train}}(f_t; f_{t-1}, \mathcal{D}_t, \mathcal{D}_{1:t-1})$ during each task t can help to improve the performance of the final model on all seen tasks. Note that there is also a lower bound of $L_{\text{test}}(f_T; \mathcal{D}_1, \ldots, \mathcal{D}_T)$, which means that minimizing the training loss during each task t is necessary. In particular, given that the training loss is a weighted sum of contrastive loss and distillation loss, these bounds also emphasize the necessity of both contrastive loss and distillation loss in effectively learning a contrastive continual learning model.

Taking inspiration from Theorem 1, we can infer that pre-training can benefit continual learning. The coefficients of training losses associated with each task become fixed if λ exceeds a certain value. For example, the denominators of the coefficients of training losses, i.e., $\{\gamma_t(\lambda)\}_{t=2}^T$ for the upper bound become constant values if λ is large. Note that the component $\alpha > 1$, then the weight $\alpha^{T-t}/\gamma_t(\lambda)$ for

the training loss of task t decreases greatly as t increases, reducing the importance of task t in the bounds. Therefore, we have the following corollary, which shows that improved training performance of initial tasks contributes more to improving the later models' performance guarantees than that of later tasks. This aligns with the idea that pre-training can benefit continual learning, as observed in previous literature (Wang et al., 2022; Hu et al., 2022).

After choosing a suitable distillation coefficient λ, training performances of initial tasks in contrastive continual learning contribute more to improving the overall performance of the final model on all tasks compared with that of the latter ones, explaining that a well pre-trained network can benefit continual learning.

We conclude from the statement above that small changes in the training performance on the first task may lead to great changes in the overall performance of the final model. For example, the weight of $L_{\text{train}}(f_1; \mathcal{D}_1)$ in the upper bound increases greatly when adding more tasks, and a large value of $L_{\text{train}}(f_1; \mathcal{D}_1)$ implies a potential great increase of the upper bound. Therefore, well-trained initial models with small training losses in continual learning can be beneficial.

5. Further Discussion on the Distillation Coefficient λ

5.1. Analysis on the distillation coefficient

Inspired by additional analysis on Theorem 1, we find that the suitable distillation coefficient λ is correlated with the weights $\{\{k_{tj}\}_{j=1}^{t-1}\}_{t=1}^{T}$ that depends on the data distributions. Specifically, we would like to choose the suitable value of λ as the turning point of the upper bound to get better theoretical guarantees. We define the turning point as the minimum value of λ at which the upper bound no longer decreases. Once the distillation coefficient λ exceeds the value of this turning point, the upper bound becomes a fixed value that is no longer influenced by λ, as illustrated in Figure 2. In the following part of this section, we will present several examples and calculate the corresponding turning point values. This may help us better understand the choice of distillation coefficient ($\lambda = 1$) employed in the experiments of Co^{2}L (Cha et al., 2021).

We begin by clarifying that in the contrastive continual learning framework, choosing a fixed distillation coefficient as one for all tasks is favorable for achieving a balance between learning new tasks and preserving old knowledge. Specifically, with well-constructed weights $\{k_{tj}\}_{j=1}^{t-1}$ for task t, the suggested λ value for learning tends to stay close to one, thereby contributing to a tighter upper bound. To illustrate this point, we provide an example below.

Example 1. Assume that there are five tasks, each task with data distribution $\mathcal{D}_t$, $t \in \{1, \ldots, 5\}$, and we have corresponding models $\{f_t\}_{t=1}^{5}$. We make the assumption that these models obtain the same value of training loss, i.e.,

$$L_{\text{train}}(f_5; f_4, \mathcal{D}_5, \mathcal{D}_{1:4}) = \cdots = L_{\text{train}}(f_1; \mathcal{D}_1).$$

Weights of tasks are given as follows, i.e., for $t \geq 2$,

$$k_{tj} = \begin{cases} \frac{2}{t}, & j = 1, \\ \frac{1}{t}, & \text{else.} \end{cases}$$

These weights can be considered well-constructed, as they are uniform across different tasks, except for the weight of the first task which has a larger value than the others. Indeed, this strategy emphasizes the importance of the first task which is typically regarded as a base task.

Then according to our Theorem 1, the value of appropriate λ is suggested to be close to one to get a tighter upper bound on the overall performance for the final model.

Note that in Example 1, we have assumed that the values of total training losses of different tasks remain the same, which may not align with realistic settings. To provide a more realistic illustration, we construct another example with adaptive ratios between the training losses of different tasks. Specifically, we allow the value of the training loss of each task to either increase or decrease with the same ratio ρ close to one. The weights construction strategy is correlated to ρ to maintain an alignment with the changing training loss. Constructed in this way, the following example illustrates how the appropriate value of λ remains close to one even when ρ fluctuates around one, and further achieves a tighter upper bound. This observation inspires us to consider adjusting the value of λ around one.

Example 2. Assume that there are five tasks, each task with data distribution $\mathcal{D}_t$, $t \in \{1, \ldots, 5\}$, and we have corresponding models $\{f_t\}_{t=1}^{5}$. We assume that the value of the training loss of each task has a fixed ratio $\rho \approx 1$, i.e.,

$$L_{\text{train}}(f_5; f_4, \mathcal{D}_5, \mathcal{D}_{1:4}) = \cdots = \rho^4 L_{\text{train}}(f_1; \mathcal{D}_1).$$

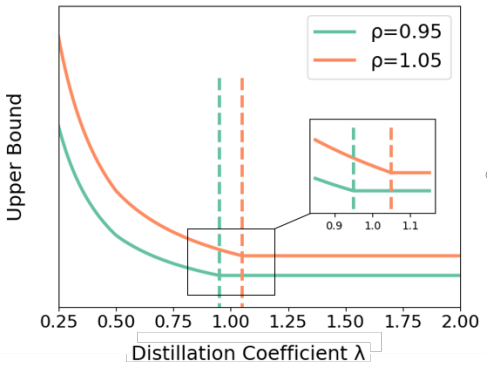

Figure 2. An illustration of Example 2. The suggesting λ for $\rho = 0.95$ or $\rho = 1.05$ stays close to one.

Weights of different tasks are given by a biased strategy related to ρ, i.e, for $t \geq 2$,

$$k_{tj} = \begin{cases} 1 - \frac{t-2}{\rho t}, & j = 1, \\ \frac{1}{\rho t}, & \text{else.} \end{cases}$$

Then according to Theorem 1, the value of appropriate λ is suggested to get close to one as ρ changes slightly around one, ensuring a tight upper bound on the overall performance of the final model. As illustrated in Figure 2, setting $\rho = 0.95$ or $\rho = 1.05$ implies that suggesting λ value remains close to one. The settings in Example 2 are more realistic and can closely resemble the Co^{2}L configuration (Cha et al., 2021). Thus this example can further support the rationale for choosing $\lambda = 1$ in Co^{2}L.

However, in an extreme case of the weights construction, an undesirable result may occur with a value of λ greater than one. This insight suggests that data distribution may play a crucial role in selecting a suitable λ since the weights $\{k_{tj}\}_{j=1}^{t-1}$ are strongly correlated with the data distribution of task t. As illustrated in the following example, when there exists a weight significantly smaller than the other weights, the value of appropriate λ would deviate from one. In such a case, persistently using $\lambda = 1$ may not achieve the best theoretical guarantees.

Example 3. Take the same assumption from Example 1, excluding the weights construction strategy. Weights of tasks are given as follows, i.e., for $t \geq 3$,

$$k_{tj} = \begin{cases} \frac{2.9}{t}, & j = 1, \\ \frac{0.1}{t}, & j = 2, \\ \frac{1}{t}, & \text{else.} \end{cases}$$

For the second task, we set $k_{21} = 1$.

Then according to our Theorem 1, the suggesting value of appropriate λ is close to ten in Example 3. If we choose $\lambda = 1$, the upper bound may get large and fail to provide the best guarantees. The failure of Example 3 is attributed to a change in the weights construction strategy, highlighting a substantial relationship between the suitable value of distillation coefficient λ and data distributions.

5.2. Adaptive selection of distillation coefficients

Inspired by our theoretical analysis, we are curious whether it is possible to provide better theoretical guarantees by dynamically adjusting distillation coefficients. Interestingly, the analysis of Theorem 1 can be adapted to the case of adaptive distillation coefficient λ_t, simply by replacing λ with λ_t for the coefficient of training loss of task t in the bounds. Then we can conclude from Theorem 1 for the new case that, *increasing* λ_t for each task t with a threshold strategy can provide better guarantees.

Note that our target is to get a set of distillation coefficients $\{\lambda_t\}_{t=2}^T$ that tighten the upper bound in Theorem 1. Therefore, we want to adaptively select λ_t for task t to achieve this goal. We now propose our theoretical explanations for the adaptive selection of distillation coefficients. First, we give some related definitions. Suppose there are T tasks for continual learning setting. For each task $t \geq 2$, we denote λ_t as the task-specific distillation coefficient, and define

$$U_t = \sum_{j=2}^{t} \frac{\alpha^{t-j}}{\gamma_j(\lambda_t)} L_{\text{train}}(f_j; f_{j-1}, \mathcal{D}_j, \mathcal{D}_{1:j-1}).$$

Motivated by Theorem 1 and explanations above, at the end of task t, if the calculated U_t has a relatively large value, then a slight increase in λ_{t+1} around one can be beneficial for improving the upper bound. Inspired by this, we will maintain an extra set of task-specific threshold values $\{u_t\}_{t=2}^T$ where $u_t > 0$, and a set of update momentums $\{\Delta_t\}_{t=1}^T$ where $\Delta_t \geq 0$. After training task t, if $U_t > u_t$, we let $\lambda_{t+1} = \lambda_t + \Delta_t$, else, $\lambda_{t+1} = \lambda_t$. Then the total training loss of each task t for model f_t can be rewritten as

$$\begin{aligned} &L_{\text{train}}(f_t; f_{t-1}, \mathcal{D}_t, \mathcal{D}_{1:t-1}) \\ &= L_{\text{con}}(f_t; \mathcal{D}_t) + \lambda_t \cdot L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}_{1:t-1}). \end{aligned}$$

The following theorem provides a theoretical explanation for the benefits of the adaptive λ_t selection protocol above.

Theorem 2. *Assume that the training loss of each task is larger than zero. For each task $t \geq 2$, there exists a task-specific constant $u_t > 0$. If we have*

$$U_t = \sum_{j=2}^{t} \frac{\alpha^{t-j}}{\gamma_j(\lambda_t)} L_{train}(f_j; f_{j-1}, \mathcal{D}_j, \mathcal{D}_{1:j-1}) > u_t,$$

where α, $\{\gamma_j(\lambda_t)\}_{j=2}^t$ are defined in Theorem 1, then we increase λ_t by Δ_t , i.e., $\lambda_{t+1} = \lambda_t + \Delta_t$, else, $\lambda_{t+1} = \lambda_t$. By choosing λ_t in this way, we get tighter upper bounds.

It can be concluded from Theorem 2 that, increasing λ_t by a threshold strategy about the performance of the current model can help make the upper bound tighter. Moreover, Theorem 2 can also provide theoretical support for the choosing strategy of λ in previous examples.

5.3. Contrastive Incremental Learning with Adaptive distillation (CILA)

Note that we lack access to the construction of weights during the real training phase. Consequently, computing U_t is not available for task t, and estimating $\{k_{tj}\}_{j=1}^{t-1}$ may be also inaccessible due to the high computing load and low estimating accuracy. However, according to our explanations above, it is suggested that the adaptive λ_t for each task t

stays around one with a relatively larger value. Hence, we can find an easy-to-get and adaptive metric to replace λ_t. Before we introduce the chosen metric, we first give some related definitions. We first define the empirical contrastive loss (Khosla et al., 2020). For each task t, we denote D_t as the given batch of N training samples $\{(x_{t,i}, c_{t,i})\}_{i=1}^N$, and the augmented batch is $\{(\tilde{x}_{t,i}, \tilde{c}_{t,i})\}_{i=1}^{2N}$ which is generated by making two randomly augmented versions of $x_{t,i}$ as $\tilde{x}_{t,2i-1}$ and $\tilde{x}_{t,2i}$ with $\tilde{c}_{t,2i-1} = \tilde{c}_{t,2i} = c_{t,i}$. The augmented samples are mapped to a d-dimensional Euclidean sphere by a model f_t. Denote $\boldsymbol{z}_{t,i} = f_t(\tilde{x}_{t,i})$, then the empirical contrastive loss can be formulated as

$$\hat{L}_{\text{con}}(f_t; D_t) = \sum_{i=1}^{2N} \frac{-1}{|p_{t,i}|} \sum_{j \in p_{t,i}} \log \left(\frac{\exp(\boldsymbol{z}_{t,i}^\top \boldsymbol{z}_{t,j}/\tau)}{\sum_{k \neq i} \exp(\boldsymbol{z}_{t,i}^\top \boldsymbol{z}_{t,k}/\tau)} \right),$$

where $p_{t,i} = \{j \in \{1, \ldots, 2N\} | j \neq i, c_{t,j} = c_{t,i}\}$ and $\tau > 0$ is the temperature hyperparameter. Then we define the empirical distillation loss (Cha et al., 2021). We first define a similarity vector

$$\boldsymbol{p}(f_t, \tau; \tilde{x}_i) = \text{softmax}(\boldsymbol{z}_{t,i}^\top \boldsymbol{z}_{t,1}/\tau, \ldots, \boldsymbol{z}_{t,i}^\top \boldsymbol{z}_{t,i-1}/\tau, \boldsymbol{z}_{t,i}^\top \boldsymbol{z}_{t,i+1}/\tau, \ldots \boldsymbol{z}_{t,i}^\top \boldsymbol{z}_{t,2N}/\tau).$$

Then the empirical distillation loss is formulated as

$$\hat{L}_{\text{dis}}(f_t; f_{t-1}, D_t) = \sum_{i=1}^{2N} -\boldsymbol{p}(f_{t-1}, \tau^*; \tilde{x}_{t,i}) \cdot \log \boldsymbol{p}(f_t, \tau; \tilde{x}_{t,i}).$$

Here, both τ^* and τ will remain fixed for all tasks. Actually, when constructing the experiment, we found that the ratio

$$\sum_{j=2}^{t-1} \hat{L}_{\text{dis}}(f_j; f_{j-1}, D_j) / \sum_{j=2}^{t-1} \hat{L}_{\text{con}}(f_j; D_j),$$

is stable and stays close to one as task index $t \geq 3$ varies. This ratio is easy to get during the training procedure and is aligned with the idea that the distillation coefficient is suggested to stay close to one and varies according to the data distribution which depends on the task index. Therefore, inspired by Example 2 and 3, an applicable method is to use

$$\lambda_t = \max(1, \kappa \sum_{j=2}^{t-1} \hat{L}_{\text{dis}}(f_j; f_{j-1}, D_j) / \sum_{j=2}^{t-1} \hat{L}_{\text{con}}(f_j; D_j)),$$

for task $t \geq 3$, where κ is a balancing distillation coefficient, and for the second task, we use $\lambda_2 = 1$. More details can be found in Algorithm 1. In the following section, we will conduct several experiments.

Algorithm 1 CILA: Contrastive Incremental Learning with Adaptive distillation

Input: Buffer size B, a sequence of training sets $\{D_t\}_{t=1}^T$, base distillation coefficient λ_0, balancing distillation coefficient κ.

Initialize model f_0 and set buffer $\mathcal{M} \leftarrow \phi$;
for task $t = 1, \cdots, T$ **do**
  Construct dataset $D \leftarrow D_t \cup \mathcal{M}$;
  Initialize model $f_t \leftarrow f_{t-1}$;
  Compute L by $L \leftarrow \hat{L}_{\text{con}}(f_t; D)$;
  if $t > 1$ **then**
    Adaptively update λ_t by $\lambda_t \leftarrow \max(\lambda_0, \kappa \cdot \frac{\sum_{j=2}^{t-1} \hat{L}_{\text{dis}}(f_j; f_{j-1}, D_j)}{\sum_{j=2}^{t-1} \hat{L}_{\text{con}}(f_j; D_j)})$;
    Update L by $L \leftarrow L + \lambda_t \cdot \hat{L}_{\text{dis}}(f_t; f_{t-1}, D)$;
  end if
  Update f_t by SGD;
  Collect buffer samples until $|\mathcal{M}| = B$;
end for

6. Experiment

Learning settings and datasets. We conducted experiments on three basic continual learning scenarios, Class-IL, Task-IL, and Domain-IL (van de Ven & Tolias, 2019). Each scenario was evaluated using different datasets. Specifically, for Class-IL and Task-IL, we utilized Seq-CIFAR-10 and Seq-Tiny-ImageNet datasets. Seq-CIFAR-10 is a modified version of the CIFAR-10 (Krizhevsky, 2009) dataset, where it is divided into 5 distinct subsets, each comprising two classes. Similarly, Seq-Tiny-ImageNet is an adapted version of the Tiny-ImageNet (Le & Yang, 2015) dataset, where the 200 classes are split into 10 separate sets, each containing 20 classes. The order of splits in Seq-CIFAR-10 and Seq-Tiny-ImageNet remains consistent across multiple runs.

For Domain-IL, we employed R-MINST, which is a variant of the MNIST (Lecun et al., 1998) dataset. In R-MINST, the original images are randomly rotated by an angle between 0 and π. R-MINST consists of 20 tasks, with each task corresponding to a randomly selected rotation angle. During the training process, samples from different tasks with the same digital class are treated as distinct classes.

In summary, our experiments covered Class-IL, Task-IL, and Domain-IL scenarios, utilizing Seq-CIFAR-10, Seq-Tiny-ImageNet, and R-MINST datasets, respectively.

Baselines. We compare our contrastive continual learning algorithm with replay-based continual learning baselines, including ER (Riemer et al., 2019), GEM (Lopez-Paz & Ranzato, 2017), A-GEM (Chaudhry et al., 2018), iCaRL (Rebuffi et al., 2017), FDR (Benjamin et al., 2019), GSS (Aljundi et al., 2019), HAL (Chaudhry et al., 2019), DER

Table 1. Classification accuracies for Seq-CIFAR-10, Seq-Tiny-ImageNet, and R-MNIST on replay-based baselines and our algorithm. All results are averaged over ten independent trials. The best performance is marked as bold.

Dataset	Seq-CIFAR-10				Seq-Tiny-ImageNet				R-MNIST	
Scenario	Class-IL		Task-IL		Class-IL		Task-IL		Domain-IL	
Buffer	200	500	200	500	200	500	200	500	200	500
ER	44.79±1.86	57.74±0.27	91.19±0.94	93.61±0.27	8.49±0.16	9.99±0.29	38.17±2.00	48.64±0.46	93.53±1.15	94.89±0.95
GEM	25.54±0.76	26.20±1.26	90.44±0.94	92.16±0.64	–	–	–	–	89.86±1.23	92.55±0.85
A-GEM	20.04±0.34	22.67±0.57	83.88±1.49	89.48±1.45	8.07±0.08	8.06±0.04	22.77±0.03	25.33±0.49	89.03±2.76	89.04±7.01
iCaRL	49.02±3.20	47.55±3.95	88.99±2.13	88.22±2.62	7.53±0.79	9.38±1.53	28.19±1.47	31.55±3.27	–	–
FDR	30.91±2.74	28.71±3.23	91.01±0.68	93.29±0.59	8.70±0.19	10.54±0.21	40.36±0.68	49.88±0.71	93.71±1.51	95.48±0.68
GSS	39.07±5.59	49.73±4.78	88.80±2.89	91.02±1.57	–	–	–	–	87.10±7.23	89.38±3.12
HAL	32.36±2.70	41.79±4.46	82.51±3.20	84.54±2.36	–	–	–	–	89.40±2.50	92.35±0.81
DER	61.93±1.79	70.51±1.67	91.40±0.92	93.40±0.39	11.87±0.78	17.75±1.14	40.22±0.67	51.78±0.88	96.43±0.59	97.57±1.47
DER++	64.88±1.17	72.70±1.36	91.92±0.60	93.88±0.50	10.96±1.17	19.38±1.41	40.87±1.16	51.91±0.68	95.98±1.06	97.54±0.43
Co^{2}L	65.57±1.37	74.26±0.77	93.43±0.78	95.90±0.26	13.88±0.40	20.12±0.42	42.37±0.74	53.04±0.69	97.90±1.92	98.65±0.31
CILA (Ours)	**67.06**±1.59	**76.03**±0.79	**94.29**±0.24	**96.40**±0.21	**14.55**±0.39	**20.64**±0.59	**44.15**±0.70	**54.13**±0.72	**98.36**±0.45	**98.76**±0.22

(Buzzega et al., 2020), DER++ (Buzzega et al., 2020), and Co^{2}L (Cha et al., 2021).

Details of training. Following the configuration of previous studies, we trained ResNet-18 on the Seq-CIFAR-10 and Tiny-ImageNet datasets. We implemented a simple network with convolution layers for the R-MNIST dataset. In our training process, we employed buffers of sizes 200 and 500. The base distillation coefficient λ_0 is set as one following the default configuration of Co^{2}L (Cha et al., 2021).

Evaluation. Like Co^{2}L, CILA follows the idea of "first pre-training, then linear probing". Thus, unlike the joint representation-classifier training approaches, an additional classifier needs to be trained on top of the frozen representations. To ensure a fair comparison, the classifier is trained using only the samples from the last task and buffered samples, leveraging the representations learned by CILA. To mitigate the challenges posed by class imbalance, we employ a class-balanced sampling strategy during the training of a linear classifier. The strategy involves the following steps. We first uniformly select a class from the available set of classes. This ensures that each class has an equal chance of being chosen. Once a class is selected, we further uniformly sample an instance from that specific class. This guarantees that all instances within the chosen class are equal to be selected.

For all experiments, a linear classifier is trained for a fixed number of epochs and we adopt 100 epochs to align with prior work. After training, the classification test accuracy is reported based on the predictions made by this classifier.

Main results. In Table 1, our method outperforms all baselines in different scenarios, datasets, and buffer sizes, especially compared with Co^{2}L. This result verifies the superiority of our adaptive method and supports our theories strongly. Our algorithm successfully reaches a balance between learning plasticity and memory stability in continual learning. Under appropriate adaptation of the distillation coefficients, we also mitigate the catastrophic forgetting problem. Besides, the power of adaptation also impacts the performance of the learned continual learner, we will talk about it in the following ablation study.

Table 2. Accuracies on Seq-CIFAR-10 with 200 buffer samples.

Adaptive Method	Class-IL	Task-IL
Co^{2}L	65.57	93.43
Pure-adapted	66.52	94.27
Min-adapted	66.36	94.21
Max-adapted	**67.06**	**94.29**

7. Ablation Studies

We conduct ablation experiments to verify the effectiveness of adaptive distillation coefficients. We consider two setups, Class-IL and Task-IL, and perform experiments on Seq-CIFAR-10 with three variants of adapted λ_t for each task t. Variants include

(1) pure-adapted

$$\lambda_{t,\text{pure}} = \kappa \sum_{j=2}^{t-1} \hat{L}_{\text{dis}}(f_j, f_{j-1}; D_j) / \sum_{j=2}^{t-1} \hat{L}_{\text{con}}(f_j; D_j),$$

(2) min-adapted

$$\lambda_{t,\text{min}} = \min(1, \lambda_{t,pure}),$$

and (3) max-adapted

$$\lambda_{t,\max} = \max(1, \lambda_{t,pure}),$$

where κ is a balancing distillation coefficient. For our ablation experiments, we train the linear classifier on top of the representations with 200 buffer samples.

As shown in Table 2, methods with adaptive distillation coefficients show superior performance compared with the method with a fixed distillation coefficient with about 1% improvement on both settings. As the adaptive λ_t increases with moderate limitations, the performance of the model boosts with an obvious improvement on Class-IL. This verifies our assumption based on the theoretical results that continual learners with larger adaptive distillation coefficients show greater performance.

8. Conclusion

Contrastive learning has demonstrated remarkable performance in the field of continual learning, although there remains a lack of theoretical explanations. In this study, we aim to fill this gap by introducing theoretical performance guarantees for the final model in contrastive continual learning. Drawing inspirations from a detailed theoretical analysis, we propose the utilization of adaptive distillation coefficients for the distillation training loss in contrastive continual learning. Through comprehensive experiments conducted in diverse settings for continual learning, our approach surpasses baseline methods in terms of performance. We anticipate that our work can establish a robust foundation for continual learning from a representation perspective, and potentially spark further theoretical insights into the realm of contrastive continual learning.

Acknowledgment

Weiran Huang is supported by 2023 CCF-Baidu Open Fund and Microsoft Research Asia. Chuanlong Xie is supported by the National Nature Science Foundation of China (No.12201048).

We would also like to express our sincere gratitude to the reviewers of ICML 2024 for their insightful and constructive feedback. Their valuable comments have greatly contributed to improving the quality of our work.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Aljundi, R., Lin, M., Goujaud, B., and Bengio, Y. Gradient based sample selection for online continual learning. In *NeurIPS*, 2019.

Arora, S., Khandeparkar, H., Khodak, M., Plevrakis, O., and Saunshi, N. A theoretical analysis of contrastive unsupervised representation learning. *arXiv:1902.09229*, 2019.

Ayub, A. and Wagner, A. R. Eec: Learning to encode and regenerate images for continual learning. In *ICLR*, 2021.

Bang, J., Kim, H., Yoo, Y., Ha, J.-W., and Choi, J. Rainbow memory: Continual learning with a memory of diverse samples. In *CVPR*, 2021.

Benjamin, A. S., Rolnick, D., and Kording, K. Measuring and regularizing networks in function space, 2019.

Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., and Shah, R. Signature verification using a" siamese" time delay neural network. In *NeurIPS*, 1993.

Buzzega, P., Boschini, M., Porrello, A., Abati, D., and Calderara, S. Dark experience for general continual learning: a strong, simple baseline. In *NeurIPS*, 2020.

Caccia, L., Belilovsky, E., Caccia, M., and Pineau, J. Online learned continual compression with adaptive quantization modules. In *ICML*, 2019.

Cha, H., Lee, J., and Shin, J. Co2l: Contrastive continual learning. In *ICCV*, 2021.

Chaudhry, A., Marc'Aurelio, R., Rohrbach, M., and Elhoseiny, M. Efficient lifelong learning with a-gem. In *ICLR*, 2018.

Chaudhry, A., Gordo, A., Dokania, P. K., Torr, P. H. S., and Lopez-Paz, D. Using hindsight to anchor past knowledge in continual learning. In *AAAI*, 2019.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *ICML*, 2020a.

Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning. *arXiv:2003.04297*, 2020b.

Chopra, S., Hadsell, R., and LeCun, Y. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005.

Cong, Y., Zhao, M., Li, J., Wang, S., and Carin, L. Gan memory with no forgetting. In *NeurIPS*, 2020.

Evron, I., Moroshko, E., Ward, R., Srebro, N., and Soudry, D. How catastrophic can catastrophic forgetting be in linear regression? In *Conference on Learning Theory*, pp. 4028–4079. PMLR, 2022.

Fini, E., da Costa, V. G. T., Alameda-Pineda, X., Ricci, E., Alahari, K., and Mairal, J. Self-supervised models are continual learners. In *CVPR*, 2022.

Gallardo, J., Hayes, T. L., and Kanan, C. Self-supervised training enhances online continual learning. In *BMVC*, 2021.

Gou, J., Yu, B., Maybank, S. J., and Tao, D. Knowledge distillation: A survey. *IJCV*, 2021.

Gurbuz, M. B. and Dovrolis, C. Nispa: Neuro-inspired stability-plasticity adaptation for continual learning in sparse networks. In *ICML*, 2022.

Hadsell, R., Chopra, S., and LeCun, Y. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv:1503.02531*, 2015.

Hu, D., Yan, S., Lu, Q., HONG, L., Hu, H., Zhang, Y., Li, Z., Wang, X., and Feng, J. How well does self-supervised pre-training perform with streaming data? In *ICLR*, 2022.

Huang, W., Yi, M., Zhao, X., and Jiang, Z. Towards the generalization of contrastive self-supervised learning. In *The Eleventh International Conference on Learning Representations*, 2023.

Javed, K. and White, M. Meta-learning representations for continual learning. In *NeurIPS*, 2019.

Jung, S., Ahn, H., Cha, S., and Moon, T. Continual learning with node-importance based adaptive group sparse regularization. In *NeurIPS*, 2020.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. Supervised contrastive learning. In *NeurIPS*, 2020.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., and Hadsell, R. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 2017.

Krizhevsky, A. Learning multiple layers of features from tiny images. https://www.cs.toronto.edu/~kriz/index.html, 2009.

Le, Y. and Yang, X. S. Tiny imagenet visual recognition challenge. https://tiny-imagenet.herokuapp.com, 2015.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

Lee, K., Lee, K., Shin, J., and Lee, H. Overcoming catastrophic forgetting with unlabeled data in the wild. In *ICCV*, 2019.

Li, H., Wu, J., and Braverman, V. Fixed design analysis of regularization-based continual learning. In *Conference on Lifelong Learning Agents*, pp. 513–533. PMLR, 2023.

Li, Z. and Hoiem, D. Learning without forgetting. *TPAMI*, 2016.

Li, Z. and Hoiem, D. Learning without forgetting. In *TPAMI*, 2017.

Liu, X., Masana, M., Herranz, L., de Weijer, J. V., Lopez, A. M., and Bagdanov, A. D. Rotate your networks: Better weight consolidation and less catastrophic forgetting. *arXiv:1802.02950*, 2018.

Lopez-Paz, D. and Ranzato, M. Gradient episodic memory for continual learning. In *NeurIPS*, 2017.

McClelland, J. L., McNaughton, B. L., and O'Reilly, R. C. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological review*, 1995.

Mobahi, H., Farajtabar, M., and Bartlett, P. L. Self-distillation amplifies regularization in hilbert space. In *NeurIPS*, 2020.

Passalis, N., Tzelepi, M., and Tefas, A. Heterogeneous knowledge distillation using information flow modeling. In *CVPR*, 2020.

Pham, Q., Liu, C., and Hoi, S. Dualnet: Continual learning, fast and slow. In *NeurIPS*, 2021.

Prabhu, A., Torr, P. H. S., and Dokania, P. K. Gdumb: A simple approach that questions our progress in continual learning. In *ECCV*, 2020.

Ramasesh, V. V., Lewkowycz, A., and Dyer, E. Effect of scale on catastrophic forgetting in neural networks. In *ICLR*, 2022.

Ramesh, R. and Chaudhari, P. Model zoo: A growing brain that learns continually. In *ICLR*, 2021.

Rebuffi, S.-A., Kolesnikov, A., Sperl, G., and Lampert, C. H. icarl: Incremental classifier and representation learning. In *CVPR*, 2017.

Riemer, M., Cases, I., Ajemian, R., Liu, M., Rish, I., Tu, Y., and Tesauro, G. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *ICLR*, 2019.

Ritter, H., Botev, A., and Barber, D. Online structured laplace approximations for overcoming catastrophic forgetting. In *NeurIPS*, 2018.

Tan, Z., Yang, J., Huang, W., Yuan, Y., and Zhang, Y. Information flow in self-supervised learning. *arXiv preprint arXiv:2309.17281*, 2023a.

Tan, Z., Zhang, Y., Yang, J., and Yuan, Y. Contrastive learning is spectral clustering on similarity graph. *arXiv preprint arXiv:2303.15103*, 2023b.

Tan, Z., Zheng, K., and Huang, W. Otmatch: Improving semi-supervised learning with optimal transport. *arXiv preprint arXiv:2310.17455*, 2023c.

Tian, Y., Krishnan, D., and Isola, P. Contrastive multiview coding. In *ECCV*, 2020.

Tiwari, R., Killamsetty, K., Iyer, R., and Shenoy, P. Gcr: Gradient coreset based replay buffer selection for continual learning. In *CVPR*, 2022.

van de Ven, G. M. and Tolias, A. S. Three scenarios for continual learning. *arXiv:1904.07734*, 2019.

van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2019.

Wang, L., Zhang, X., Su, H., and Zhu, J. A comprehensive survey of continual learning: Theory, method and application. *arXiv:2302.00487*, 2023.

Wang, Z., Shen, L., Fang, L., Suo, Q., Zhan, D., Duan, T., and Gao, M. Meta-learning with less forgetting on large-scale non-stationary task distributions. In *ECCV*, 2022.

Zagoruyko, S. and Komodakis, N. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *ICLR*, 2016.

Zhang, Y., Tan, Z., Yang, J., Huang, W., and Yuan, Y. Matrix information theory for self-supervised learning. *arXiv preprint arXiv:2305.17326*, 2023.

Zhang, Z. and Sabuncu, M. R. Self-distillation as instance-specific label smoothing. In *NeurIPS*, 2020.

Zhao, X., Wang, H., Huang, W., and Lin, W. A statistical theory of regularization-based continual learning. In *International conference on machine learning*. PMLR, 2024.

Appendix

A. Proof of Lemma 1

We recall Lemma 1.

Lemma 1. *When $t \geq 2$, for any data distribution $\mathcal{D}$, the contrastive losses of current model f_t and previous model f_{t-1} can be connected via the distillation loss, i.e.,*

$$L_{\text{con}}(f_t; \mathcal{D}) \leq \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) + \beta,$$
$$L_{\text{con}}(f_t; \mathcal{D}) \geq \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) + \beta',$$

where $\alpha = \frac{2e^2}{1+e^2}$, $\beta = 2 - \alpha + \alpha \log \frac{\alpha}{2}$, and $\beta' = -\alpha \log(1 + e^2) - \alpha$.

Proof. For model f, denote the data pair $\boldsymbol{x} = (x, x^+, x^-)$ to simplify the proof, we first define $v(f; \boldsymbol{x}) = f(x)^\top (f(x^+) - f(x^-))$, and

$$q(f; \boldsymbol{x}) = \frac{\exp(v(f; \boldsymbol{x}))}{1 + \exp(v(f; \boldsymbol{x}))}.$$

For models f_t and f_{t-1}, the function $\ell(v) = \log(1 + \exp(-v))$, and $\boldsymbol{p}(f; \boldsymbol{x}) = \text{softmax}(f(x)^\top f(x^+), f(x)^\top f(x^-))$, we have the following equation

$$\begin{aligned}
&-\boldsymbol{p}(f_{t-1}; \boldsymbol{x}) \cdot \log \boldsymbol{p}(f_t; \boldsymbol{x}) \\
&= -q(f_{t-1}; \boldsymbol{x}) \log(q(f_t; \boldsymbol{x})) - (1 - q(f_{t-1}; \boldsymbol{x})) \log(1 - q(f_t; \boldsymbol{x})) \\
&= q(f_{t-1}; \boldsymbol{x}) \log(1 + \exp(-v(f_t; \boldsymbol{x}))) + (1 - q(f_{t-1}; \boldsymbol{x})) \log(1 + \exp(v(f_t; \boldsymbol{x}))) \\
&= q(f_{t-1}; \boldsymbol{x}) \log(1 + \exp(-v(f_t; \boldsymbol{x}))) + (1 - q(f_{t-1}; \boldsymbol{x})) \log(1 + \exp(-v(f_t; \boldsymbol{x}))) + (1 - q(f_{t-1}; \boldsymbol{x})) \log(\exp(v(f_t; \boldsymbol{x}))) \\
&= \ell(v(f_t; \boldsymbol{x})) + (1 - q(f_{t-1}; \boldsymbol{x})) v(f_t; \boldsymbol{x}).
\end{aligned}$$

Then for any data distribution $\mathcal{D}$, we have

$$\begin{aligned}
&L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) \\
&= \mathop{\mathbb{E}}_{\substack{c^+ \sim \mu \\ c^- \sim \mu}} \mathop{\mathbb{E}}_{\substack{x, x^+ \sim \mathcal{D}_{c^+} \\ x^- \sim \mathcal{D}_{c^-}}} -\boldsymbol{p}(f_{t-1}; \boldsymbol{x}) \cdot \log \boldsymbol{p}(f_t; \boldsymbol{x}) \\
&= \mathop{\mathbb{E}}_{\substack{c^+ \sim \mu \\ c^- \sim \mu}} \mathop{\mathbb{E}}_{\substack{x, x^+ \sim \mathcal{D}_{c^+} \\ x^- \sim \mathcal{D}_{c^-}}} \ell(v(f_t; \boldsymbol{x})) + \mathop{\mathbb{E}}_{\substack{c^+ \sim \mu \\ c^- \sim \mu}} \mathop{\mathbb{E}}_{\substack{x, x^+ \sim \mathcal{D}_{c^+} \\ x^- \sim \mathcal{D}_{c^-}}} (1 - q(f_{t-1}; \boldsymbol{x})) v(f_t; \boldsymbol{x}) \\
&= L_{\text{con}}(f_t; \mathcal{D}) + \mathop{\mathbb{E}}_{\substack{c^+ \sim \mu \\ c^- \sim \mu}} \mathop{\mathbb{E}}_{\substack{x, x^+ \sim \mathcal{D}_{c^+} \\ x^- \sim \mathcal{D}_{c^-}}} (1 - q(f_{t-1}; \boldsymbol{x})) v(f_t; \boldsymbol{x}).
\end{aligned}$$

Note that f_t, f_{t-1} are normalized, i.e., $-2 \leq v(f_{t-1}; \boldsymbol{x}) \leq 2$, and $-2 \leq v(f_t; \boldsymbol{x}) \leq 2$. We first prove the upper bound. By using the following inequality

$$\alpha \log(1 + e^{-h}) + \beta - \frac{2}{1 + e^h} \geq 0,$$

where $\alpha = \frac{2e^2}{1+e^2}$, $\beta = 2 - \alpha + \alpha \log \frac{\alpha}{2}$, $-2 \leq h \leq 2$, and using $v(f_{t-1}; \boldsymbol{x})$ to replace h, we have

$$\begin{aligned}
(1 - q(f_{t-1}; \boldsymbol{x})) v(f_t; \boldsymbol{x}) &\geq -2(1 - q(f_{t-1}; \boldsymbol{x})) \\
&= -\frac{2}{1 + \exp(v(f_{t-1}; \boldsymbol{x}))} \\
&\geq -\alpha \log(1 + \exp(-v(f_{t-1}; \boldsymbol{x}))) - \beta \\
&= -\alpha \ell(v(f_{t-1}; \boldsymbol{x})) - \beta.
\end{aligned}$$

Using the result above, we have

$$\begin{aligned}L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) &= L_{\text{con}}(f_t; \mathcal{D}) + \mathop{\mathbb{E}}_{\substack{c^+\sim\mu \\ c^-\sim\mu}} \mathop{\mathbb{E}}_{x, x^+\sim\mathcal{D}_{c^+} \\ x^-\sim\mathcal{D}_{c^-}} (1 - q(f_{t-1}; \boldsymbol{x}))v(f_t; \boldsymbol{x}) \\ &\geq L_{\text{con}}(f_t; \mathcal{D}) + \mathop{\mathbb{E}}_{\substack{c^+\sim\mu \\ c^-\sim\mu}} \mathop{\mathbb{E}}_{x, x^+\sim\mathcal{D}_{c^+} \\ x^-\sim\mathcal{D}_{c^-}} [-\alpha\ell(v(f_{t-1}; \boldsymbol{x})) - \beta] \\ &= L_{\text{con}}(f_t; \mathcal{D}) - \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) - \beta.\end{aligned}$$

which means

$$L_{\text{con}}(f_t; \mathcal{D}) \leq \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) + \beta.$$

This finishes the upper bound part. Then let us consider the lower bound. We use the inequality

$$\alpha \log(1 + e^{-h}) + \beta' + \frac{2}{1 + e^h} \leq 0,$$

where $\alpha = \frac{2e^2}{1+e^2}$, $\beta' = -\alpha \log(1 + e^2) - \alpha$, $-2 \leq h \leq 2$, and using $v(f_{t-1}; \boldsymbol{x})$ to replace h, then we have

$$\begin{aligned}(1 - q(f_{t-1}; \boldsymbol{x}))v(f_t; \boldsymbol{x}) &\leq 2(1 - q(f_{t-1}; \boldsymbol{x})) \\ &= \frac{2}{1 + \exp(v(f_{t-1}; \boldsymbol{x}))} \\ &\leq -\alpha \log(1 + \exp(-v(f_{t-1}; \boldsymbol{x})) - \beta' \\ &= -\alpha\ell(v(f_{t-1}; \boldsymbol{x})) - \beta'.\end{aligned}$$

Using the result above, we have

$$\begin{aligned}L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) &= L_{\text{con}}(f_t; \mathcal{D}) + \mathop{\mathbb{E}}_{\substack{c^+\sim\mu \\ c^-\sim\mu}} \mathop{\mathbb{E}}_{x, x^+\sim\mathcal{D}_{c^+} \\ x^-\sim\mathcal{D}_{c^-}} (1 - q(f_{t-1}; \boldsymbol{x}))v(f_t; \boldsymbol{x}) \\ &\leq L_{\text{con}}(f_t; \mathcal{D}) + \mathop{\mathbb{E}}_{\substack{c^+\sim\mu \\ c^-\sim\mu}} \mathop{\mathbb{E}}_{x, x^+\sim\mathcal{D}_{c^+} \\ x^-\sim\mathcal{D}_{c^-}} [-\alpha\ell(v(f_{t-1}; \boldsymbol{x})) - \beta'] \\ &= L_{\text{con}}(f_t; \mathcal{D}) - \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) - \beta'.\end{aligned}$$

It can be translated into

$$L_{\text{con}}(f_t; \mathcal{D}) \geq \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) + \beta'.$$

The proof has finished. □

B. Proof of Theorem 1

We recall Theorem 1.

Theorem 1. *For the contrastive continual learning involving $T \geq 2$ tasks, the test loss of the final model f_T can be bounded via a linear combination of the training losses associated with each task. More specifically, the following two bounds are applicable.*

(1) Upper bound:

$$\begin{aligned}L_{\text{test}}(f_T; \mathcal{D}_1, \ldots, \mathcal{D}_T) &\leq \alpha^{T-1} L_{\text{train}}(f_1; \mathcal{D}_1) \\ &+ \sum_{t=2}^{T} \frac{\alpha^{T-t}}{\gamma_t(\lambda)} L_{\text{train}}(f_t; f_{t-1}, \mathcal{D}_t, \mathcal{D}_{1:t-1}) + \eta,\end{aligned}$$

(2) Lower bound:

$$L_{\text{test}}(f_T;\mathcal{D}_1,\ldots,\mathcal{D}_T) \geq \alpha^{T-1}L_{\text{train}}(f_1;\mathcal{D}_1) + \sum_{t=2}^{T}\frac{\alpha^{T-t}}{\gamma_t'(\lambda)}L_{\text{train}}(f_t;f_{t-1},\mathcal{D}_t,\mathcal{D}_{1:t-1}) + \eta',$$

where

$$\begin{cases} \alpha = \frac{2e^2}{1+e^2}, \\ \gamma_t(\lambda) = \min\left(\{\frac{1}{t}\}\cup\{\lambda k_{tj}\}_{j=1}^{t-1}\right), \\ \gamma_t'(\lambda) = \max\left(\{1\}\cup\{\lambda k_{tj}\}_{j=1}^{t-1}\right), \\ \eta = (2-\alpha+\alpha\log\frac{\alpha}{2})\frac{T-1-T\alpha+(\alpha)^T}{(1-\alpha)^2} \\ \qquad + \sum_{t=2}^{T}\alpha^{T-t}(1-\frac{1}{\gamma_t(\lambda)})\min_f L_{\text{con}}(f;\mathcal{D}_t), \\ \eta' = -(\alpha\log(1+e^2)+\alpha)\frac{T-1-T\alpha+(\alpha)^T}{(1-\alpha)^2}. \end{cases}$$

Proof. We first proof the upper bound. For models f_t and f_{t-1}, we have

$$\begin{aligned} L_{\text{dis}}(f_t;f_{t-1},\mathcal{D}_{1:t-1}) &= \mathop{\mathbb{E}}_{\substack{c^+\sim\mu_{1:t-1}\\ c^-\sim\mu_{1:t-1}}}\ \mathop{\mathbb{E}}_{x,x^+\sim\mathcal{D}_{c^+}\ x^-\sim\mathcal{D}_{c^-}}[-\boldsymbol{p}(f_{t-1};\boldsymbol{x})\cdot\log\boldsymbol{p}(f_t;\boldsymbol{x})] \\ &= \sum_{j=1}^{t-1}k_{tj}\mathop{\mathbb{E}}_{\substack{c^+\sim\mu_j\\ c^-\sim\mu_j}}\ \mathop{\mathbb{E}}_{x,x^+\sim\mathcal{D}_{c^+}\ x^-\sim\mathcal{D}_{c^-}}[-\boldsymbol{p}(f_{t-1};\boldsymbol{x})\cdot\log\boldsymbol{p}(f_t;\boldsymbol{x})] \\ &= \sum_{j=1}^{t-1}k_{tj}L_{\text{dis}}(f_t;f_{t-1},\mathcal{D}_j). \end{aligned}$$

Then we can write the training loss as

$$\begin{aligned} L_{\text{train}}(f_t;f_{t-1},\mathcal{D}_t,\mathcal{D}_{1:t-1}) &= L_{\text{con}}(f_t;\mathcal{D}_t)+\lambda L_{\text{dis}}(f_t;f_{t-1},\mathcal{D}_{1:t-1}) \\ &= L_{\text{con}}(f_t;\mathcal{D}_t)+\lambda\sum_{j=1}^{t-1}k_{tj}L_{\text{dis}}(f_t;f_{t-1},\mathcal{D}_j). \end{aligned}$$

for task $t\geq 2$, and $L_{\text{train}}(f_1;\mathcal{D}_1) = L_{\text{con}}(f_1;\mathcal{D}_1)$. According to the proof of Lemma 1, for models f_t and f_{t-1}, data distribution $\mathcal{D}_j$ $(j\leq t)$, we have

$$L_{\text{con}}(f_t,\mathcal{D}_j)\leq\alpha L_{\text{con}}(f_{t-1},\mathcal{D}_j)+L_{\text{dis}}(f_t;f_{t-1},\mathcal{D}_j)+\beta.$$

Denote $\gamma_t(\lambda) = \min\left(\{\frac{1}{t}\} \cup \{\lambda k_{tj}\}_{j=1}^{t-1}\right)$ for task $t \geq 2$. Then we have

$$
\begin{aligned}
&L_{\text{test}}(f_T; \mathcal{D}_{1:T}) \\
&= L_{\text{con}}(f_T; \mathcal{D}_T) + \sum_{t=1}^{T-1} L_{\text{con}}(f_T, \mathcal{D}_t) \\
&\leq L_{\text{con}}(f_T; \mathcal{D}_T) + \sum_{t=1}^{T-1} [L_{\text{dis}}(f_T; f_{T-1}, \mathcal{D}_t) + \alpha L_{\text{con}}(f_{T-1}; \mathcal{D}_t) + \beta] \\
&\leq (\frac{1}{\gamma_T(\lambda)} - \frac{1}{\gamma_T(\lambda)} + 1) L_{\text{con}}(f_T; \mathcal{D}_T) + \frac{\lambda}{\gamma_T(\lambda)} \sum_{t=1}^{T-1} k_{Tt} L_{\text{dis}}(f_T; f_{T-1}, \mathcal{D}_t) + \sum_{t=1}^{T-1} [\alpha L_{\text{con}}(f_{T-1}; \mathcal{D}_t) + \beta] \\
&\leq \frac{1}{\gamma_T(\lambda)} L_{\text{train}}(f_T; f_{T-1}, \mathcal{D}_T, \mathcal{D}_{1:T-1}) + (1 - \frac{1}{\gamma_T(\lambda)}) \min_f L_{\text{con}}(f; \mathcal{D}_T) \\
&\quad + (T-1)\beta + \alpha L_{\text{test}}(f_{T-1}; \mathcal{D}_{1:T-1}) \\
&\quad \vdots \\
&\leq \alpha^{T-1} L_{\text{train}}(f_1; \mathcal{D}_1) + \sum_{t=2}^{T} \frac{\alpha^{T-t}}{\gamma_t(\lambda)} L_{\text{train}}(f_t; f_{t-1}, \mathcal{D}_t, \mathcal{D}_{1:t-1}) + \eta.
\end{aligned}
$$

where $\alpha = \frac{2e^2}{1+e^2}$, $\eta = (2 - \alpha + \alpha \log \frac{\alpha}{2}) \frac{T-1-T\alpha+(\alpha)^T}{(1-\alpha)^2} + \sum_{t=2}^{T} \alpha^{T-t}(1 - \frac{1}{\gamma_t(\lambda)}) \min_f L_{\text{con}}(f; \mathcal{D}_t)$.

Let us prove the lower bound. According to the proof of Lemma 1, for models f_t and f_{t-1}, and data distribution $\mathcal{D}_j$ $(j \leq t)$, we have

$$
L_{\text{con}}(f_t, \mathcal{D}_j) \geq \alpha L_{\text{con}}(f_{t-1}, \mathcal{D}_j) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}_j) + \beta'.
$$

Denote $\gamma'_t(\lambda) = \max\left(\{1\} \cup \{\lambda k_{tj}\}_{j=1}^{t-1}\right)$ for task $t \geq 2$. The proof is similar to that of the upper bound.

$$
\begin{aligned}
&L_{\text{test}}(f_T; \mathcal{D}_{1:T}) \\
&= L_{\text{con}}(f_T; \mathcal{D}_T) + \sum_{t=1}^{T-1} L_{\text{con}}(f_T, \mathcal{D}_t) \\
&\geq L_{\text{con}}(f_T; \mathcal{D}_T) + \sum_{t=1}^{T-1} [L_{\text{dis}}(f_T; f_{T-1}, \mathcal{D}_t) + \alpha L_{\text{con}}(f_{T-1}; \mathcal{D}_t) + \beta'] \\
&\geq \frac{1}{\gamma'_t(\lambda)} [L_{\text{con}}(f_T; \mathcal{D}_T) + \lambda \sum_{t=1}^{T-1} k_{Tt} L_{\text{dis}}(f_T; f_{T-1}, \mathcal{D}_t)] + \sum_{t=1}^{T-1} [\alpha L_{\text{con}}(f_{T-1}; \mathcal{D}_t) + \beta'] \\
&= \frac{1}{\gamma'_t(\lambda)} L_{\text{train}}(f_T; f_{T-1}, \mathcal{D}_T, \mathcal{D}_{1:T-1}) + \alpha \sum_{t=1}^{T-1} L_{\text{con}}(f_{T-1}; \mathcal{D}_t) + (T-1)\beta' \\
&= \frac{1}{\gamma'_t(\lambda)} L_{\text{train}}(f_T; f_{T-1}, \mathcal{D}_T, \mathcal{D}_{1:T-1}) + \alpha L_{\text{test}}(f_{T-1}; \mathcal{D}_{1:T-1}) + (T-1)\beta' \\
&\geq \alpha^{T-1} L_{\text{train}}(f_1; \mathcal{D}_1) + \sum_{t=2}^{T} \frac{\alpha^{T-t}}{\gamma'_t(\lambda)} L_{\text{train}}(f_t; f_{t-1}, \mathcal{D}_t, \mathcal{D}_{1:t-1}) + \eta'.
\end{aligned}
$$

where $\alpha = \frac{2e^2}{1+e^2}$, $\eta' = -(\alpha \log(1+e^2) + \alpha) \frac{T-1-T\alpha+(\alpha)^T}{(1-\alpha)^2}$. □

C. Proof of Theorem 2

We recall Theorem 2.

Theorem 2. *Assume that the training loss of each task is larger than zero. For each task $t \geq 2$, there exists a task-specific constant $u_t > 0$. If we have*

$$U_t = \sum_{j=2}^{t} \frac{\alpha^{t-j}}{\gamma_j(\lambda_t)} L_{train}(f_j; f_{j-1}, \mathcal{D}_j, \mathcal{D}_{1:j-1}) > u_t,$$

where α, $\{\gamma_j(\lambda_t)\}_{j=2}^t$ are defined in Theorem 1, then we increase λ_t by Δ_t , i.e., $\lambda_{t+1} = \lambda_t + \Delta_t$, else, $\lambda_{t+1} = \lambda_t$. By choosing λ_t in this way, we get tighter upper bounds.

Proof. Note that $U_t > 0$, thus u_t exists. If λ_t increases by $\Delta \geq 0$, then $\lambda_{t+1} \geq \lambda_t$ and $\gamma_j(\lambda_{t+1}) \geq \gamma_j(\lambda_t)$. We have

$$U'_{t+1} = \sum_{j=2}^{t+1} \frac{\alpha^{t+1-j}}{\gamma_j(\lambda_{t+1})} L_{\text{train}}(f_j; f_{j-1}, \mathcal{D}_j) \leq U_{t+1} = \sum_{j=2}^{t+1} \frac{\alpha^{t+1-j}}{\gamma_j(\lambda_t)} L_{\text{train}}(f_j; f_{j-1}, \mathcal{D}_j).$$

Thus the upper bound becomes tighter. □

D. The Case of Multiple Negative Examples for Lemma 1

Following our definitions in the paper, the contrastive loss for the case of $k (k \geq 1)$ negative samples can be formulated as

$$L_{\text{con}}(f; \mathcal{D}_t) = \mathop{\mathbb{E}}_{\substack{c^+ \sim \mu_t \\ c_i^- \sim \mu_t}} \mathop{\mathbb{E}}_{\substack{x, x^+ \sim \mathcal{D}_{c^+} \\ x_i^- \sim \mathcal{D}_{c_i^-}}} \ell\left[\{f(x)^\top (f(x^+) - f(x_i^-))\}\right],$$

where function $\ell(\boldsymbol{v})$ is defined as $\ell(\boldsymbol{v}) = \log(1 + \sum_{i=1}^k \exp(-v_i))$ for $\boldsymbol{v} \in \mathbb{R}^k$ and embeddings are conventionally normalized, i.e., $\|f\| = 1$. The distillation loss for the case of k negative samples can be formulated as

$$L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}_{1:t-1}) = \mathop{\mathbb{E}}_{\substack{c^+ \sim \mu_{1:t-1} \\ c_i^- \sim \mu_{1:t-1}}} \mathop{\mathbb{E}}_{\substack{x, x^+ \sim \mathcal{D}_{c^+} \\ x_i^- \sim \mathcal{D}_{c_i^-}}} \left[-\boldsymbol{p}(f_{t-1}; x, x^+, x_1^-, \ldots, x_k^-) \cdot \log \boldsymbol{p}(f_t; x, x^+, x_1^-, \ldots, x_k^-)\right],$$

where $\mu_{1:t-1}$ represents the class distribution of $\mathcal{D}_{1:t-1}$ and

$$\boldsymbol{p}(f; x, x^+, x_1^-, \ldots, x_k^-) = \text{softmax}(f(x)^\top f(x^+), f(x)^\top f(x_1^-), \ldots, f(x)^\top f(x_k^-)).$$

Then we provide the extended version of Lemma 1 and its proof.

Lemma 2. *When $t \geq 2$ and the number of negative samples $k \geq 1$, for any data distribution $\mathcal{D}$, the contrastive losses of current model f_t and previous model f_{t-1} can be connected via the distillation loss, i.e.,*

$$L_{\text{con}}(f_t; \mathcal{D}) \leq \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) + \beta,$$
$$L_{\text{con}}(f_t; \mathcal{D}) \geq \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) + \beta',$$

where $\alpha = \frac{2e^2}{k+e^2}$, $\beta = 2 - \alpha + \alpha \log \frac{\alpha}{2}$, and $\beta' = -\alpha \log(1 + ke^2) - \frac{2ke^2}{1+ke^2}$.

Proof. For model f, denote the data pair $\boldsymbol{x} = (x, x^+, x_1^-, \ldots, x_k^-)$ to simplify the proof, we first define $v_i(f; \boldsymbol{x}) = f(x)^\top (f(x^+) - f(x_i^-))$, and

$$q_i(f; \boldsymbol{x}) = \frac{\exp(-v_i(f; \boldsymbol{x}))}{1 + \sum_{i=1}^k \exp(-v_i(f; \boldsymbol{x}))},$$
$$q(f; \boldsymbol{x}) = \frac{1}{1 + \sum_{i=1}^k \exp(-v_i(f; \boldsymbol{x}))},$$

where $q(f;\boldsymbol{x})+\sum_{i=1}^{k}q_i(f;\boldsymbol{x})=1$. For models f_t and f_{t-1}, the function $\ell(\boldsymbol{v})=\log(1+\sum_{i=1}^{k}\exp(-v_i))$ for $\boldsymbol{v}\in\mathbb{R}^k$, and

$$\boldsymbol{p}(f;\boldsymbol{x})=\mathrm{softmax}(f(x)^\top f(x^+),f(x)^\top f(x_1^-),\ldots,f(x)^\top f(x_k^-)),$$

we have the following equation

$$\begin{aligned}
&-\boldsymbol{p}(f_{t-1};\boldsymbol{x})\cdot\log\boldsymbol{p}(f_t;\boldsymbol{x})\\
&=-q(f_{t-1};\boldsymbol{x})\log(q(f_t;\boldsymbol{x}))-\sum_{i=1}^{k}q_i(f_{t-1};\boldsymbol{x})\log(q_i(f_t;\boldsymbol{x}))\\
&=q(f_{t-1};\boldsymbol{x})\log(1+\sum_{i=1}^{k}\exp(-v_i(f_t;\boldsymbol{x})))-\sum_{i=1}^{k}q_i(f_{t-1};\boldsymbol{x})[-v_i(f_t;\boldsymbol{x})-\log(1+\sum_{i=1}^{k}\exp(-v_i(f_t;\boldsymbol{x})))]\\
&=q(f_{t-1};\boldsymbol{x})\log(1+\sum_{i=1}^{k}\exp(-v_i(f_t;\boldsymbol{x})))+\sum_{i=1}^{k}q_i(f_{t-1};\boldsymbol{x})[v_i(f_t;\boldsymbol{x})+\log(1+\sum_{i=1}^{k}\exp(-v_i(f_t;\boldsymbol{x})))]\\
&=\log(1+\sum_{i=1}^{k}\exp(-v_i(f_t;\boldsymbol{x})))+\sum_{i=1}^{k}q_i(f_{t-1};\boldsymbol{x})v_i(f_t;\boldsymbol{x})\\
&=\ell(\boldsymbol{v}(f_t;\boldsymbol{x}))+\sum_{i=1}^{k}q_i(f_{t-1};\boldsymbol{x})v_i(f_t;\boldsymbol{x}).
\end{aligned}$$

Then for any data distribution $\mathcal{D}$, we have

$$\begin{aligned}
&L_{\text{dis}}(f_t;f_{t-1},\mathcal{D})\\
&=\mathop{\mathbb{E}}_{\substack{c^+\sim\mu\\ c_i^-\sim\mu}}\mathop{\mathbb{E}}_{\substack{x,x^+\sim\mathcal{D}_{c^+}\\ x_i^-\sim\mathcal{D}_{c_i^-}}}-\boldsymbol{p}(f_{t-1};\boldsymbol{x})\cdot\log\boldsymbol{p}(f_t;\boldsymbol{x})\\
&=\mathop{\mathbb{E}}_{\substack{c^+\sim\mu\\ c_i^-\sim\mu}}\mathop{\mathbb{E}}_{\substack{x,x^+\sim\mathcal{D}_{c^+}\\ x_i^-\sim\mathcal{D}_{c_i^-}}}\ell(\boldsymbol{v}(f_t;\boldsymbol{x}))+\mathop{\mathbb{E}}_{\substack{c^+\sim\mu\\ c_i^-\sim\mu}}\mathop{\mathbb{E}}_{\substack{x,x^+\sim\mathcal{D}_{c^+}\\ x_i^-\sim\mathcal{D}_{c_i^-}}}\sum_{i=1}^{k}q_i(f_{t-1};\boldsymbol{x})v_i(f_t;\boldsymbol{x})\\
&=L_{\text{con}}(f_t;\mathcal{D})+\mathop{\mathbb{E}}_{\substack{c^+\sim\mu\\ c_i^-\sim\mu}}\mathop{\mathbb{E}}_{\substack{x,x^+\sim\mathcal{D}_{c^+}\\ x_i^-\sim\mathcal{D}_{c_i^-}}}\sum_{i=1}^{k}q_i(f_{t-1};\boldsymbol{x})v_i(f_t;\boldsymbol{x}).
\end{aligned}$$

Note that f_t, f_{t-1} are normalized, i.e., $-2\le v_i(f_{t-1};\boldsymbol{x})\le 2$, and $-2\le v_i(f_t;\boldsymbol{x})\le 2$. We first prove the upper bound. By using the following inequality

$$\alpha\log h+\beta\ge 2(1-\frac{1}{h}),$$

where $\alpha=\frac{2e^2}{k+e^2}$, $\beta=2-\alpha+\alpha\log\frac{\alpha}{2}$, $1+ke^{-2}\le h\le 1+ke^2$, and using $1+\sum_{i=1}^{k}\exp(-v_i(f_{t-1};\boldsymbol{x}))$ to replace h in the inequality above, we have

$$\begin{aligned}
\sum_{i=1}^{k}q_i(f_{t-1};\boldsymbol{x})v_i(f_t;\boldsymbol{x})&\ge -2(1-q(f_{t-1};\boldsymbol{x}))\\
&=-2(1-\frac{1}{1+\sum_{i=1}^{k}\exp(-v_i(f_{t-1};\boldsymbol{x}))})\\
&\ge-\alpha\log(1+\sum_{i=1}^{k}\exp(-v_i(f_{t-1};\boldsymbol{x})))-\beta\\
&=-\alpha\ell(\boldsymbol{v}(f_{t-1};\boldsymbol{x}))-\beta.
\end{aligned}$$

Using the results above, we have

$$\begin{aligned}
L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) &= L_{\text{con}}(f_t; \mathcal{D}) + \mathop{\mathbb{E}}_{\substack{c^+\sim\mu \\ c_i^-\sim\mu}} \mathop{\mathbb{E}}_{\substack{x, x^+\sim\mathcal{D}_{c^+} \\ x_i^-\sim\mathcal{D}_{c_i^-}}} \sum_{i=1}^{k} q_i(f_{t-1}; \boldsymbol{x}) v_i(f_t; \boldsymbol{x}) \\
&\geq L_{\text{con}}(f_t; \mathcal{D}) + \mathop{\mathbb{E}}_{\substack{c^+\sim\mu \\ c_i^-\sim\mu}} \mathop{\mathbb{E}}_{\substack{x, x^+\sim\mathcal{D}_{c^+} \\ x_i^-\sim\mathcal{D}_{c_i^-}}} [-\alpha \ell(\boldsymbol{v}(f_{t-1}; \boldsymbol{x})) - \beta] \\
&= L_{\text{con}}(f_t; \mathcal{D}) - \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) - \beta.
\end{aligned}$$

which means

$$L_{\text{con}}(f_t; \mathcal{D}) \leq \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) + \beta.$$

This finishes the upper bound part. Then let us consider the lower bound. We use the inequality

$$-\alpha \log h - \beta' \geq 2(1 - \frac{1}{h}),$$

where $\alpha = \frac{2e^2}{k+e^2}$, $\beta' = -\alpha \log(1 + ke^2) - \frac{2ke^2}{1+ke^2}$, $1 + ke^{-2} \leq h \leq 1 + ke^2$, and using $1 + \sum_{i=1}^{k} \exp(-v_i(f_{t-1}; \boldsymbol{x}))$ to replace h, then we have

$$\begin{aligned}
\sum_{i=1}^{k} q_i(f_{t-1}; \boldsymbol{x}) v_i(f_t; \boldsymbol{x}) &\leq 2(1 - q(f_{t-1}; \boldsymbol{x})) \\
&= 2(1 - \frac{1}{1 + \sum_{i=1}^{k} \exp(-v_i(f_{t-1}; \boldsymbol{x}))}) \\
&\leq -\alpha \log(1 + \sum_{i=1}^{k} \exp(-v_i(f_{t-1}; \boldsymbol{x})) - \beta' \\
&= -\alpha \ell(\boldsymbol{v}(f_{t-1}; \boldsymbol{x})) - \beta'.
\end{aligned}$$

Using the results above, we have

$$\begin{aligned}
L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) &= L_{\text{con}}(f_t; \mathcal{D}) + \mathop{\mathbb{E}}_{\substack{c^+\sim\mu \\ c_i^-\sim\mu}} \mathop{\mathbb{E}}_{\substack{x, x^+\sim\mathcal{D}_{c^+} \\ x_i^-\sim\mathcal{D}_{c_i^-}}} \sum_{i=1}^{k} q_i(f_{t-1}; \boldsymbol{x}) v_i(f_t; \boldsymbol{x}) \\
&\leq L_{\text{con}}(f_t; \mathcal{D}) + \mathop{\mathbb{E}}_{\substack{c^+\sim\mu \\ c_i^-\sim\mu}} \mathop{\mathbb{E}}_{\substack{x, x^+\sim\mathcal{D}_{c^+} \\ x_i^-\sim\mathcal{D}_{c_i^-}}} [-\alpha \ell(\boldsymbol{v}(f_{t-1}; \boldsymbol{x})) - \beta'] \\
&= L_{\text{con}}(f_t; \mathcal{D}) - \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) - \beta'.
\end{aligned}$$

It can be translated into

$$L_{\text{con}}(f_t; \mathcal{D}) \geq \alpha L_{\text{con}}(f_{t-1}; \mathcal{D}) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}) + \beta'.$$

The proof has finished. □

E. The Case of Multiple Negative Examples for Theorem 1

We provide the extended version of Theorem 1 and its proof.

Theorem 3. *For the contrastive continual learning involving $T \geq 2$ tasks where each task involves k negative samples, the test loss of the final model f_T can be bounded via a linear combination of the training losses associated with each task. More specifically, the following two bounds are applicable.*

(1) Upper bound:

$$L_{\text{test}}(f_T;\mathcal{D}_1,\ldots,\mathcal{D}_T) \le \alpha^{T-1}L_{\text{train}}(f_1;\mathcal{D}_1) + \sum_{t=2}^{T}\frac{\alpha^{T-t}}{\gamma_t(\lambda)}L_{\text{train}}(f_t;f_{t-1},\mathcal{D}_t,\mathcal{D}_{1:t-1}) + \eta,$$

(2) Lower bound:

$$L_{\text{test}}(f_T;\mathcal{D}_1,\ldots,\mathcal{D}_T) \ge \alpha^{T-1}L_{\text{train}}(f_1;\mathcal{D}_1) + \sum_{t=2}^{T}\frac{\alpha^{T-t}}{\gamma'_t(\lambda)}L_{\text{train}}(f_t;f_{t-1},\mathcal{D}_t,\mathcal{D}_{1:t-1}) + \eta',$$

where

$$\begin{cases} \alpha = \frac{2e^2}{k+e^2}, \\ \gamma_t(\lambda) = \min\left(\{\frac{1}{t}\}\cup\{\lambda k_{tj}\}_{j=1}^{t-1}\right), \\ \gamma'_t(\lambda) = \max\left(\{1\}\cup\{\lambda k_{tj}\}_{j=1}^{t-1}\right), \\ \eta = (2-\alpha+\alpha\log\frac{\alpha}{2})\frac{T-1-T\alpha+(\alpha)^T}{(1-\alpha)^2} \\ \qquad + \sum_{t=2}^{T}\alpha^{T-t}(1-\frac{1}{\gamma_t(\lambda)})\min_f L_{\text{con}}(f;\mathcal{D}_t), \\ \eta' = -(\alpha\log(1+ke^2)+\frac{2ke^2}{1+ke^2})\frac{T-1-T\alpha+(\alpha)^T}{(1-\alpha)^2}. \end{cases}$$

Proof. We first proof the upper bound. For models f_t and f_{t-1}, we have

$$\begin{aligned} L_{\text{dis}}(f_t;f_{t-1},\mathcal{D}_{1:t-1}) &= \mathop{\mathbb{E}}_{\substack{c^+\sim\mu_{1:t-1}\\ c^-\sim\mu_{1:t-1}}}\mathop{\mathbb{E}}_{\substack{x,x^+\sim\mathcal{D}_{c^+}\\ x^-\sim\mathcal{D}_{c^-}}}[-\boldsymbol{p}(f_{t-1};\boldsymbol{x})\cdot\log\boldsymbol{p}(f_t;\boldsymbol{x})] \\ &= \sum_{j=1}^{t-1}k_{tj}\mathop{\mathbb{E}}_{\substack{c^+\sim\mu_j\\ c^-\sim\mu_j}}\mathop{\mathbb{E}}_{\substack{x,x^+\sim\mathcal{D}_{c^+}\\ x^-\sim\mathcal{D}_{c^-}}}[-\boldsymbol{p}(f_{t-1};\boldsymbol{x})\cdot\log\boldsymbol{p}(f_t;\boldsymbol{x})] \\ &= \sum_{j=1}^{t-1}k_{tj}L_{\text{dis}}(f_t;f_{t-1},\mathcal{D}_j). \end{aligned}$$

Then we can write the training loss as

$$\begin{aligned} L_{\text{train}}(f_t;f_{t-1},\mathcal{D}_t,\mathcal{D}_{1:t-1}) &= L_{\text{con}}(f_t;\mathcal{D}_t)+\lambda L_{\text{dis}}(f_t;f_{t-1},\mathcal{D}_{1:t-1}) \\ &= L_{\text{con}}(f_t;\mathcal{D}_t)+\lambda\sum_{j=1}^{t-1}k_{tj}L_{\text{dis}}(f_t;f_{t-1},\mathcal{D}_j). \end{aligned}$$

for task $t\ge 2$, and $L_{\text{train}}(f_1;\mathcal{D}_1) = L_{\text{con}}(f_1;\mathcal{D}_1)$. According to the proof of Lemma 1, for models f_t and f_{t-1}, data distribution $\mathcal{D}_j$ $(j\le t)$, we have

$$L_{\text{con}}(f_t,\mathcal{D}_j) \le \alpha L_{\text{con}}(f_{t-1},\mathcal{D}_j) + L_{\text{dis}}(f_t;f_{t-1},\mathcal{D}_j) + \beta.$$

Denote $\gamma_t(\lambda) = \min\left(\{\frac{1}{t}\} \cup \{\lambda k_{tj}\}_{j=1}^{t-1}\right)$ for task $t \geq 2$. According to the equations above, we have

$$
\begin{aligned}
&L_{\text{test}}(f_T; \mathcal{D}_{1:T}) \\
&= L_{\text{con}}(f_T; \mathcal{D}_T) + \sum_{t=1}^{T-1} L_{\text{con}}(f_T, \mathcal{D}_t) \\
&\leq L_{\text{con}}(f_T; \mathcal{D}_T) + \sum_{t=1}^{T-1} [L_{\text{dis}}(f_T; f_{T-1}, \mathcal{D}_t) + \alpha L_{\text{con}}(f_{T-1}; \mathcal{D}_t) + \beta] \\
&\leq (\frac{1}{\gamma_T(\lambda)} - \frac{1}{\gamma_T(\lambda)} + 1) L_{\text{con}}(f_T; \mathcal{D}_T) + \frac{\lambda}{\gamma_T(\lambda)} \sum_{t=1}^{T-1} k_{Tt} L_{\text{dis}}(f_T; f_{T-1}, \mathcal{D}_t) + \sum_{t=1}^{T-1} [\alpha L_{\text{con}}(f_{T-1}; \mathcal{D}_t) + \beta] \\
&\leq \frac{1}{\gamma_T(\lambda)} L_{\text{train}}(f_T; f_{T-1}, \mathcal{D}_T, \mathcal{D}_{1:T-1}) + (1 - \frac{1}{\gamma_T(\lambda)}) \min_f L_{\text{con}}(f; \mathcal{D}_T) \\
&\quad + (T-1)\beta + \alpha L_{\text{test}}(f_{T-1}; \mathcal{D}_{1:T-1}) \\
&\quad \vdots \\
&\leq \alpha^{T-1} L_{\text{train}}(f_1; \mathcal{D}_1) + \sum_{t=2}^{T} \frac{\alpha^{T-t}}{\gamma_t(\lambda)} L_{\text{train}}(f_t; f_{t-1}, \mathcal{D}_t, \mathcal{D}_{1:t-1}) + \eta.
\end{aligned}
$$

where $\alpha = \frac{2e^2}{k+e^2}$, $\eta = (2 - \alpha + \alpha \log \frac{\alpha}{2}) \frac{T-1-T\alpha+(\alpha)^T}{(1-\alpha)^2} + \sum_{t=2}^{T} \alpha^{T-t}(1 - \frac{1}{\gamma_t(\lambda)}) \min_f L_{\text{con}}(f; \mathcal{D}_t)$.

Let us prove the lower bound. According to the proof of Lemma 1, for models f_t and f_{t-1}, and data distribution $\mathcal{D}_j$ ($j \leq t$), we have

$$
L_{\text{con}}(f_t, \mathcal{D}_j) \geq \alpha L_{\text{con}}(f_{t-1}, \mathcal{D}_j) + L_{\text{dis}}(f_t; f_{t-1}, \mathcal{D}_j) + \beta'.
$$

Denote $\gamma'_t(\lambda) = \max\left(\{1\} \cup \{\lambda k_{tj}\}_{j=1}^{t-1}\right)$ for task $t \geq 2$. The proof is similar to that of the upper bound. Similarly, we have

$$
\begin{aligned}
&L_{\text{test}}(f_T; \mathcal{D}_{1:T}) \\
&= L_{\text{con}}(f_T; \mathcal{D}_T) + \sum_{t=1}^{T-1} L_{\text{con}}(f_T, \mathcal{D}_t) \\
&\geq L_{\text{con}}(f_T; \mathcal{D}_T) + \sum_{t=1}^{T-1} [L_{\text{dis}}(f_T; f_{T-1}, \mathcal{D}_t) + \alpha L_{\text{con}}(f_{T-1}; \mathcal{D}_t) + \beta'] \\
&\geq \frac{1}{\gamma'_t(\lambda)} [L_{\text{con}}(f_T; \mathcal{D}_T) + \lambda \sum_{t=1}^{T-1} k_{Tt} L_{\text{dis}}(f_T; f_{T-1}, \mathcal{D}_t)] + \sum_{t=1}^{T-1} [\alpha L_{\text{con}}(f_{T-1}; \mathcal{D}_t) + \beta'] \\
&= \frac{1}{\gamma'_t(\lambda)} L_{\text{train}}(f_T; f_{T-1}, \mathcal{D}_T, \mathcal{D}_{1:T-1}) + \alpha \sum_{t=1}^{T-1} L_{\text{con}}(f_{T-1}; \mathcal{D}_t) + (T-1)\beta' \\
&= \frac{1}{\gamma'_t(\lambda)} L_{\text{train}}(f_T; f_{T-1}, \mathcal{D}_T, \mathcal{D}_{1:T-1}) + \alpha L_{\text{test}}(f_{T-1}; \mathcal{D}_{1:T-1}) + (T-1)\beta' \\
&\geq \alpha^{T-1} L_{\text{train}}(f_1; \mathcal{D}_1) + \sum_{t=2}^{T} \frac{\alpha^{T-t}}{\gamma'_t(\lambda)} L_{\text{train}}(f_t; f_{t-1}, \mathcal{D}_t, \mathcal{D}_{1:t-1}) + \eta'.
\end{aligned}
$$

where $\alpha = \frac{2e^2}{k+e^2}$, $\eta' = -(\alpha \log(1 + ke^2) + \frac{2ke^2}{1+ke^2}) \frac{T-1-T\alpha+(\alpha)^T}{(1-\alpha)^2}$. □