Self-Evolved Dynamic Expansion Model for Task-Free Continual Learning

Fei Ye and Adrian G. Bors
Department of Computer Science, University of York, York YO10 5GH, UK
fy689@york.ac.uk, adrian.bors@york.ac.uk

Abstract

Task-Free Continual Learning (TFCL) aims to learn new concepts from a stream of data without any task information. The Dynamic Expansion Model (DEM) has shown promising results in TFCL by dynamically expanding the model's capacity to deal with shifts in the data distribution. However, existing approaches only consider the recognition of the input shift as the expansion signal and ignore the correlation between the newly incoming data and previously learned knowledge, resulting in adding and training unnecessary parameters. In this paper, we propose a novel and effective framework for TFCL, which dynamically expands the architecture of a DEM model through a self-assessment mechanism evaluating the diversity of knowledge among existing experts as expansion signals. This mechanism ensures learning additional underlying data distributions with a compact model structure. A novelty-aware sample selection approach is proposed to manage the memory buffer that forces the newly added expert to learn novel information from a data stream, which further promotes the diversity among experts. Moreover, we also propose to reuse previously learned representation information for learning new incoming data by using knowledge transfer in TFCL, which has not been explored before. The DEM expansion and training are regularized through a gradient updating mechanism to gradually explore the positive forward transfer, further improving the performance. Empirical results on TFCL benchmarks show that the proposed framework outperforms the state-of-the-art while using a reasonable number of parameters. The code is available at https://github.com/dtuzi123/SEDEM/.

1. Introduction

An ideal artificial intelligence system should be able to constantly learn and acquire new concepts from a changing environment all the time. Such a capability, which is increasingly emerging as a hot topic in AI, is referred to as continual/lifelong learning. However, most modern Deep Learning models fail to achieve the goal of continual learning (CL) since they rewrite previously learnt parameters to fit new tasks and then suffer from a significant drop in performance on the past tasks. Such a phenomenon is called catastrophic forgetting [37, 39].

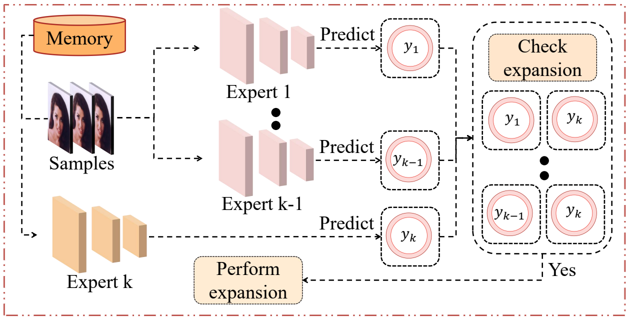


Figure 1. The diversity evaluation between experts in the Self-Evolved Dynamic Expansion Model (SEDEM). We draw all samples from the memory buffer as inputs for each expert. We compare the outputs $\{y_1, \cdots, y_k\}$ between all previously learnt experts and the currently k-th updated expert, and use Eq. (1) to check the model expansion.

Current work on reducing forgetting in continual learning (CL) falls into three categories: Memory/experience replay [6], regularisation-based approaches [29], and dynamic network architectures [46]. A simple and efficient approach among these methods is to maintain a fixed-capacity memory buffer with training examples, which replays past examples to the model along with learning new tasks. Regularisation approaches can be used on memory buffers to further improve the performance in continual learning [53]. In addition to the memory-based methods, the dynamic expansion architecture approach increases the capacity of the model as it learns new tasks, providing better generalisation performance [16].

Although previous work in CL has shown promising results, most approaches assume that the task's identity is known during training. Nevertheless, such a learning scenario is rarely encountered in the real world. In this work, we study a more challenging CL scenario called Task-Free Continual Learning (TFCL) [4], where a model is trained on a data stream without accessing the task information at any point in time. Current memory-based approaches can be extended for TFCL by developing an efficient sample selection strategy [3] that selectively stores samples and replays them at each training time. However, these approaches would suffer from the interference between the probabilistic representations of old and the newly seen samples [32]. This issue can be solved by using the dynamic expansion model (DEM) [33, 43, 67] which increases the model's capacity to handle incoming samples while freezing previously learnt experts to preserve prior knowledge. The main challenge for DEMs is that of learning a compact model structure without sacrificing much performance. Learning a lightweight DEM in TFCL can have two main advantages, such as scalability, learning infinite data streams, and fast inference at the testing phase. However, existing DEM methods fail to achieve this goal since they do not consider the knowledge diversity among experts when expanding their architecture, resulting in experts learning redundant information.

In this paper, we address two core issues in TFCL, yet untouched before.

First, instead of previous methods directly detecting the outlier samples as expansion signals, we solve the trade-off between model size and performance by formulating the expansion process for a DEM as the knowledge diversity evaluation in the mixture system. Specifically, we evaluate the diversity among the mixture's experts through a self-assessment mechanism. This allows us to control easily the growth of the model's complexity. In addition, maintaining the diversity among experts can allow us to model more underlying data distributions with a compact structure. We call our mixture system the Self-Evolved Dynamic Expansion Model (SEDEM) since it evaluates the diversity of the system, as shown in Fig. 1, where we assume to have already learnt k experts (classifiers) ('Expert 1',..., 'Expert k'). We draw all samples from a memory buffer as inputs for each expert, and then we compare the outputs $\{y_1, \ldots, y_k\}$ between all previously learnt experts ('Expert 1' ,..., 'Expert k-1') and the currently updated expert ('Expert k') as the diversity score. Expansion signals are provided if and only if this diversity score is above a certain threshold controlling the model's complexity.

Second, as most current works do not explore the benefit from the knowledge transfer in TFCL, we propose incorporating feature representations extracted from all previously learned experts into a currently updated expert. We propose the Dynamic Expansible Knowledge Mask Mechanism (DEKMM), which generates soft masks to regulate these representations when learning incoming samples. DEKMM continuously updates mask values to progressively explore potential knowledge transfer through a gradient optimisation mechanism. The DEKMM has several advantages : (1) It does not require the task information for knowledge transfer; (2) It can find the optimal mask values maximising the benefits from the positive knowledge transfer; (3) It can dynamically create new mask parameters to adapt to the expansion of SEDEM without forgetting.

Moreover, a novelty-aware sample selection approach is proposed to selectively store those training samples that are sufficiently different from the knowledge preserved by all previously learnt experts. Such a selective approach encourages the current expert to learn novel information, further promoting the diversity among experts and improving the performance of SEDEM.

We perform a series of experiments demonstrating that the proposed methodology outperforms the state-of-the-art under all settings while employing fewer experts than other DEM methods, which is consistent with our theoretical results. We summarise our contributions as follows :

- We propose a new model for TFCL, namely the Self-Evolved Dynamic Expansion Model (SEDEM) which evaluates the diversity among experts as the expansion signals, inducing a diverse and compact mixture system.
- We propose a novelty-aware sample selection approach which allows the current expert to learn novel samples, further promoting the diversity among experts.
- We propose the Dynamic Expansible Knowledge Mask Mechanism (DEKMM) to regulate previously learnt representation information when learning incoming data in TFCL, maximising the positive knowledge transfer.
- We provide theoretical guarantees for the proposed SEDEM, which are consistent with the empirical results.
- The proposed model achieves state-of-the-art performance in standard TFCL benchmarks.

2. Related Work

Memory based methods : One efficient approach to relieve catastrophic forgetting in TFCL is by storing a subset of training samples for each task into a memory buffer [6, 7, 9, 19, 44, 49, 54, 55, 58]. During subsequent tasks learning, the memory buffer replays samples that are combined with newly given data for training the model. The memory-based approaches have also been enabled with regularization, resulting into a unified optimization framework [29, 26, 36, 5, 11, 10, 35, 15, 47, 53, 38, 2, 20, 72, 21, 23, 17, 14, 52], where the replayed samples are used to penalize the change of some network parameters that are important to past tasks during the optimization. In addition, training a generator to produce past samples by using a Variational Autoencoder (VAE) [28] or a Generative Adversarial Nets (GANs) [18] was considered in several continual learning approaches [1, 42, 43, 48, 73, 59, 62, 71, 69, 68, 63, 66, 61, 70, 64, 60]. These methods are characterized by the ability to generate infinite numbers of samples [42] with a fixed size model which can be used for a growing number of tasks.

Dynamic Expansion Approaches : The memory-based approaches, including the generative replay mechanism, are not scalable for learning an unlimited number of tasks due to their fixed memory capacity and because of requiring repeated training processes [73]. Recently, the mixture/ensemble model, enabled with an expansion mechanism, was proposed to deal with continual learning challenges [12, 22, 34, 40, 43, 46, 56, 57, 74, 41, 25, 16]. These models usually achieve optimal performance on past tasks while outperforming static models on the multi-domain setting due to their scalability [56].

Task-Free Continual Learning (TFCL) : Recent works have drawn attention to the TFCL and one promising approach is to employ a memory buffer for storing selected past learnt samples. This approach was first investigated in [4] for training a classifier under TFCL and then it was extended in the Maximal Interfered Retrieval (MIR) [3] to train both VAEs and classifiers through a retrieval mechanism that selectively stores the most perturbed samples. Aljundi *et al.* [5], considered the Gradient Sample Selection (GSS) as a constrained optimization problem for the memory buffer. More recently, the sample selection was implemented through a *learner-evaluator* framework, called the Continual Prototype Evolution (CoPE) [13], which aims to maintain a balanced memory buffer, providing improved performances on imbalanced data streams. Meanwhile, the Gradient-based Memory EDiting (GMED) modifies the memorized samples such that to increase the loss in the upcoming model updates. However, all these approaches rely on a single memory system, which is not scalable for learning infinite data streams. The Dynamic Expansion Model (DEM) approach to TFCL, such as in the Continual Unsupervised Representation Learning (CURL) [43], aims to address the learning of infinite data streams. CURL dynamically builds new inference models to capture new experiences from a data stream. A Dirichlet process-based expansion mechanism aiming to increase the model's capacity was used in [33]. However, these approaches ignore the knowledge diversity among the experts when performing the expansion leading to non-optimal architectures.

3. Methodology

In this section, we describe a new model for TFCL, namely the Self-Evolved Dynamic Expansion Model (SEDEM). We start with defining the problem setting and basic network architecture.

3.1. Preliminary

Let $\mathcal{D}_r^S = \{\mathbf{x}_j^S, y_j^S\}_{j=1}^{N_r^S}$ and $\mathcal{D}_r^T = \{\mathbf{x}_j^T, y_j^T\}_{j=1}^{N_r^T}$ be the training and testing set of the r-th domain/dataset, where $\mathbf{x}_j^S$ and y_j^S are the data sample and its associated class label. Let $\mathcal{V}$ be a data stream consisting of samples from $\mathcal{D}_r^S$, expressed as $\mathcal{V} = \bigcup_{j=1}^{n} \mathcal{B}_j^r$, where $\mathcal{B}_j^r \in \mathcal{D}_r^S$ denotes a batch of samples (in the experiments the batch size is 10) and n represents the total number of training steps. During a certain training step ($\mathcal{S}_j$), the model only accesses $\mathcal{B}_j^r$ while all previously seen data batches $\{\mathcal{B}_1^r, \cdots, \mathcal{B}_{j-1}^r\}$ are not available. After all training steps are completed, we evaluate the model's performance on the testing set $\mathcal{D}_r^T$. In addition to the existing TFCL setting, we also consider a data stream $\mathcal{V}$ consisting of several different data domains, expressed as $\mathcal{V} = \bigcup_{r=1}^{w} \bigcup_{j=1}^{n} \mathcal{B}_j^r$ where w is the number of datasets. This setting is more challenging than those currently considered for CL since the data stream $\mathcal{V}$ consists of several different underlying data distributions.

Expert in SEDEM : Let $\mathbf{Q} = \{\mathcal{Q}_1, \cdots, \mathcal{Q}_k\}$ be a SEDEM model with k experts, where each $\mathcal{Q}_j$ consists of a feature extractor $f_{\omega_j} \colon \mathcal{X} \to \mathcal{Z}$ and a linear classifier $C_{\gamma_j} \colon \mathcal{Z} \to \mathcal{Y}$ where $\mathcal{X}$, $\mathcal{Z}$ and $\mathcal{Y}$ represent the space of the sample, features and class labels, respectively. We employ $f_{\omega_j} \circ C_{\gamma_j} \colon \mathcal{X} \to \mathcal{Y}$ to denote the prediction process where $\{\omega_j, \gamma_j\}$ are the parameters of expert $\mathcal{Q}_j$.

3.2. Expansion mechanism based on self-evaluation

Existing expansion criteria usually detect the outlier samples as expansion signals [33, 43] and ignore considering that the experts should learn an information diversity when adding new expert components, leading to non-optimal network architectures. In this section, we propose a novel dynamic expansion mechanism that evaluates the distance between the currently updated expert and the other experts, as an expansion signal. Let $\mathcal{C}_i$ denote a memory buffer of fixed capacity (the maximum number of memorized samples is λ) where the subscript i denotes that $\mathcal{C}_i$ is updated at $\mathcal{S}_i$. Suppose that we have trained SEDEM with k experts on $\mathcal{C}_i$ at $\mathcal{S}_i$, where $\mathcal{Q}_k$ is the currently updated expert while all previously learnt experts $\{\mathcal{Q}_1, \cdots, \mathcal{Q}_{k-1}\}$ are frozen to preserve past knowledge. The similarity measure between $\mathcal{Q}_k$ and $\{\mathcal{Q}_j \mid j = 1, \cdots, k-1\}$ at $\mathcal{S}_i$, is used as an expansion signal :

$$\max\left\{\mathcal{L}_b(\mathcal{Q}_1, \mathcal{Q}_k), \cdots, \mathcal{L}_b(\mathcal{Q}_{k-1}, \mathcal{Q}_k)\right\} \leq \beta\,, \quad (1)$$

where $\mathcal{L}_b(\mathcal{Q}_j, \mathcal{Q}_k)$ is the similarity measure function :

$$\mathcal{L}_b(\mathcal{Q}_j, \mathcal{Q}_k) = \frac{1}{m}\sum_{t=1}^{m}\left\{\mathcal{L}_e(f_{\omega_j} \circ C_{\gamma_j}(\mathbf{x}_t), f_{\omega_k} \circ C_{\gamma_k}(\mathbf{x}_t))\right\} \quad (2)$$

where $\mathbf{x}_t \sim \mathcal{C}_i$. $\mathcal{L}_e(y, y')$ returns 1 if $y = y'$, otherwise, returns 0. A large value for the left-hand-side expression of Eq. (1) indicates that adding $\mathcal{Q}_k$ can maintain the diversity of knowledge among the experts. We dynamically add a new expert ($\mathcal{Q}_k$) to $\mathbf{Q}$ if Eq. (1) is fulfilled during the training. The expansion threshold β controls the trade-off between the model size and generalization performance

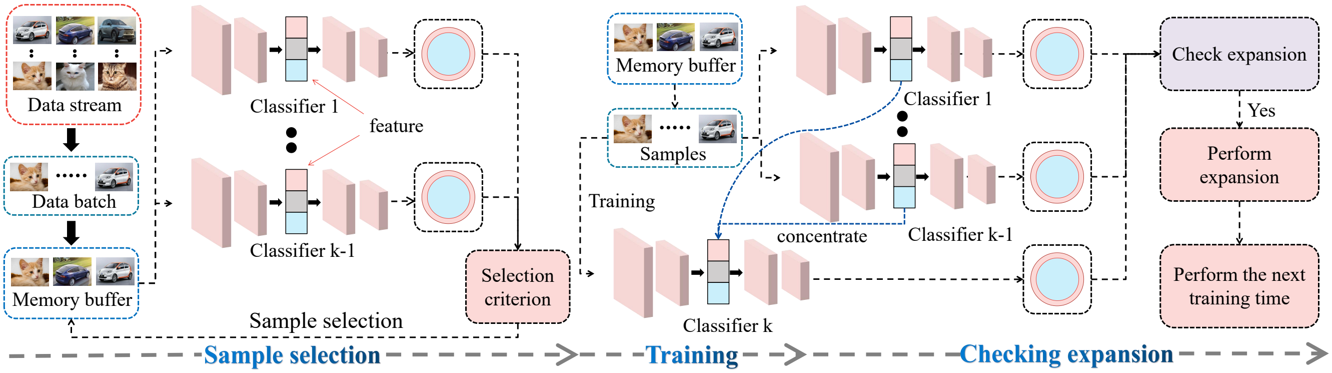


Figure 2. The learning procedure of the proposed SEDEM having k experts, where we omit the expert selector, DEKMM and the testing phase for simplicity. First, at the training step ($\mathcal{S}_i$), we perform the sample selection (Eq. (4)) if the memory buffer size increases more than λ. Second, we only update the current $\mathcal{Q}_k$ expert's classifier and VAE, using Eq. (8) and Eq. (9). Third, we check the model expansion using Eq. (1) if the memory buffer is full. If Eq. (1) is satisfied, we add a new expert $\mathcal{Q}_{k+1}$ into $\mathbf{Q}$.

and its range is considered as $\beta \in [0, 1]$ in Eq. (1). A large β encourages the model to add more experts, resulting in better performance. In contrast, a small β makes the model to have fewer experts, which would lead to a degenerated performance. The detailed theoretical analysis for choosing β is provided in **Appendix-A3** from the Supplemental Material (SM).

3.3. Novelty-Aware Sample Selection

Memory buffers in other CL studies [4, 5] are mainly used for storing past samples which are then replayed to relieve forgetting during the training. In this paper, instead of trying to preserve all past information, we propose a new sample selection approach aiming to store novel samples that are different from the knowledge preserved by previously trained experts. Such a sample selection approach can encourage the current expert to learn novel information, further promoting the knowledge diversity among experts during expansion. Let us suppose that we have trained k experts in the SEDEM model at $\mathcal{S}_i$, and then we calculate the selection score for each memorized sample as :

$$\mathcal{L}_s(\mathbf{x}_j^m) = -\frac{1}{k-1}\sum_{d=1}^{k-1}\sum_{t=1}^{C}\left\{y_j^m(t)\log(p_j^d(t))\right\}, \quad (3)$$

where C is the total number of classes and $\{\mathbf{x}_j^m, y_j^m\}$ is the j-th labelled sample drawn from $\mathcal{C}_i$. $p_j^d(t)$ is the SoftMax probability for the t-th class, predicted by $f_{\omega_d} \circ C_{\gamma_d}(\mathbf{x}_j^m)$ and $y_j^m(t)$ is the t-th dimension of the one-hot form of y_j^m. Eq. (3) estimates the average cross-entropy for each sample using all previously learnt experts. A large $\mathcal{L}_s(\mathbf{x}_j^m)$ indicates that $\mathbf{x}_j^m$ is novel with respect to the already learnt knowledge and should be added to the memory buffer. Then we perform the sample selection :

$$\mathcal{C}_i = \{\mathbf{x}_j^m \mid \mathcal{L}_s(\mathbf{x}_j^m) > \mathcal{L}_s(\mathbf{x}_{j+1}^m), j = 1, \cdots, \lambda\}, \quad (4)$$

where λ is the memory buffer size. Eq. (4) favours to preserve data with large average cross-entropy in the memory.

3.4. Dynamically Expansible Knowledge Mask

Most existing continual learning works focus mainly on addressing forgetting while ignoring the knowledge transfer in TFCL. In this section, we view all previously learnt experts as a knowledge base which would provide a positive forward transfer for future learning. To implement this goal, we propose the Dynamically Expansible Knowledge Mask Method (DEKMM), a new approach that utilizes the previously learnt representation information for learning new samples. Suppose that we have k experts $\mathbf{Q} = \{\mathcal{Q}_1, \cdots, \mathcal{Q}_k\}$ at $\mathcal{S}_i$. When training the current expert $\mathcal{Q}_k$ on $\mathcal{C}_{i+1}$, at the next training step ($\mathcal{S}_{i+1}$), we combine the previously learnt information with the currently learnt feature vectors into an augmented vector, $\mathbf{z} = \sum_{j=1}^{k-1}\{f_{\omega_j}(\mathbf{x})\} \oplus f_{\omega_k}(\mathbf{x})$ which is used as input for the classifier $C_{\gamma_k}(\mathbf{z})$, where $\oplus$ denotes the concentrated operator. However, this augmented feature vector ignores the correlation between each previously learnt feature and the incoming sample, which does not fully explore the benefit of knowledge transfer. To address this issue, DEKMM builds a trainable mask vector for $\mathcal{G}_k$, denoted as $\alpha^k \in \mathbb{R}^{k-1}$ and then normalizes it using the SoftMax function :

$$\pi^k[i] = \frac{\exp\{\alpha^k[i]\}}{\sum_{j=1}^{k-1}\{\exp\{\alpha^k[j]\}\}}, i = 1, \cdots, k-1, \quad (5)$$

where $\pi^k[i]$ denotes the i-th entry from π^k and is used to regulate the representation information from $\mathcal{Q}_i$, $i = 1, \ldots, k-1$ when optimizing $\mathcal{Q}_k$. Therefore, the augmented feature vector with the soft masks is expressed as $\mathbf{z} = \sum_{j=1}^{k-1}\{\pi^k[j]f_{\omega_j}(\mathbf{x})\} \oplus f_{\omega_k}(\mathbf{x})$. While learning the incoming samples, we update the mask vector α^k by minimizing the model's objective function (classification loss) to gradually explore the potential forward transfer. Moreover, once the proposed SEDEM dynamically adds a new expert ($\mathcal{Q}_{k+1}$), we freeze all previously learnt mask vectors $\{\alpha^1, \cdots, \alpha^k\}$ to preserve past information while building a new mask vector $\alpha^{k+1} \in \mathbb{R}^k$ to regulate the optimization of $\mathcal{Q}_{k+1}$ in subsequent learning.

3.5. The Expert Selector

The task information is unavailable in the TFCL framework in both the training and testing phases. It is necessary to develop a suitable mechanism for expert selection in the testing phase. To achieve this goal, we propose to train a simple VAE model $G_{(\phi_i,\varphi_i)}$ as an expert selector for each $\mathcal{Q}_i$, which consists of an encoding distribution $q_{\varphi_i}(\mathbf{z}_s \,|\, \mathbf{z})$ and a decoding distribution $p_{\phi_i}(\mathbf{z} \,|\, \mathbf{z}_s)$, where $\mathbf{z} \in \mathcal{Z}$ and $\mathbf{z}_s \in \mathcal{Z}_s$ are the variables over the feature space $\mathcal{Z}$ and the latent space $\mathcal{Z}_s$. Unlike in other VAE models [28] which take images as inputs, we aim to model the feature information extracted by the feature extractor f_{ω_i} of each expert $\mathcal{Q}_i$. This approach reduces the number of parameters further and provides an efficient inference mechanism at the testing stage. The main objective for training the i-th expert selector $G_{(\phi_i,\varphi_i)}$ is defined as :

$$\begin{aligned}\mathcal{L}_{VAE}(\mathbf{z}; G_{(\phi_i,\varphi_i)}) = \mathbb{E}_{q_{\varphi_i}(\mathbf{z}_s \,|\, \mathbf{z})}\left[\log p_{\phi_i}\left(\mathbf{z} \,|\, \mathbf{z}_s\right)\right] \\ - KL\left[q_{\varphi_i}\left(\mathbf{z}_s \,|\, \mathbf{z}\right) \,||\, p\left(\mathbf{z}_s\right)\right] ,\end{aligned} \tag{6}$$

where $KL(\cdot \,||\, \cdot)$ represents the Kullback-Leibler (KL) divergence and $p(\mathbf{z}_s) = \mathcal{N}(0,1)$ is a prior distribution (Gaussian). Suppose that we already have k experts after the training. At the testing phase, we perform the expert selection by comparing the sample log-likelihood estimated by Eq. (6) :

$$s^\star = \arg \max_{s=1,\cdots,k} \{\mathcal{L}_{VAE}(f_{\omega_s}(\mathbf{x}); G_{(\phi_s,\varphi_s)})\} , \tag{7}$$

where $\mathbf{x}$ is the input and $s^\star$ is the selected expert index. Eq. (7) chooses the expert with the highest sample log-likelihood. In the following section, we provide the implementation of the proposed SEDEM.

3.6. Optimization and Implementation

Each expert $\mathcal{Q}_i$ in the proposed SEDEM consists of a classifier module $f_{\omega_i} \circ C_{\gamma_i}$ and a VAE model $G_{(\phi_i,\varphi_i)}$ used as the expert selector. We provide the pseudocode used for training the SEDEM, in **Algorithm 1**, described as follows. We continually add incoming data batches $\mathcal{B}_i^r$ to $\mathcal{C}_i$, and perform the sample selection using Eq. (4) if the memory buffer size is larger than λ. Then we only optimize the current expert $\mathcal{Q}_k$ by using the two loss functions at $\mathcal{S}_i$:

$$\mathcal{L}_{cl} = -\frac{1}{\lambda} \sum\nolimits_{j=1}^{\lambda} \left\{ \sum\nolimits_{t=1}^{C} \left\{ y_j^m(t) \log(p_j^k(t)) \right\} \right\} , \tag{8}$$

$$\mathcal{L}_{Vl} = -\frac{1}{\lambda} \sum\nolimits_{j=1}^{\lambda} \left\{ \mathcal{L}_{VAE}(\mathbf{z}_j; G_{(\phi_k,\varphi_k)}) \right\} , \tag{9}$$

where $p_j^k(t)$ is the SoftMax probability for the t-th class, predicted by using $f_{\omega_k} \circ C_{\gamma_k}(\mathbf{x}_j^m), \mathbf{x}_j^m \sim \mathcal{C}_i$. $\mathbf{z}_j$ is the j-th feature vector extracted by using the feature extractor f_{ω_k} of $\mathcal{Q}_k$. Eq. (8) and Eq. (9) are employed to train the classifier $f_{\omega_k} \circ C_{\gamma_k}$ with the mask parameters and the expert selector $G_{(\phi_k,\varphi_k)}$ on $\mathcal{C}_i$ at $\mathcal{S}_i$, as shown in Fig. 2. We also check the model expansion using Eq. (1) if the memory is full ($|\mathcal{C}_i| = \lambda$), where $|\mathcal{C}_i|$ is the number of memorized samples. In the testing phase, we employ Eq. (7) to select an expert for the evaluation. The detailed implementation can be found in **Appendix-B** from SM.

Algorithm 1 Training algorithm for SEDEM

```
for $i < n$ do
    $\mathcal{C}_i = \mathcal{C}_{i-1} \bigcup \mathcal{B}_i^r, \mathcal{B}_i^r \sim \mathcal{S}$ Add a new data batch.
    Sample selection :
    if $|\mathcal{C}_i| > \lambda$ then
        $\mathcal{C}_i = \{\mathbf{x}_j^m \mid \mathcal{L}_s(\mathbf{x}_j^m) < \mathcal{L}_s(\mathbf{x}_{j+1}^m), j = 1, \cdots, \lambda\}$
    end if
    Training the SEDEM :
    if $|\mathbf{Q}| = 1$ and $i > \lambda$ then
        $\mathbf{Q} = \mathcal{Q}_2 \bigcup \mathbf{Q}$ Add the second expert.
    end if
    $k = |\mathbf{Q}|$ The number of experts.
    Train the classifier of $\mathcal{Q}_k$ on $\mathcal{C}_i$ using $\mathcal{L}_{cl}$
    Train the expert selector of $\mathcal{Q}_k$ on $\mathcal{C}_i$ using $\mathcal{L}_{Vl}$
    Dynamic expansion :
    if $|\mathcal{C}_i| > \lambda$ then
        if $\min\{\mathcal{L}_b(\mathcal{Q}_1, \mathcal{Q}_k), \cdots, \mathcal{L}_b(\mathcal{Q}_{k-1}, \mathcal{Q}_k)\} \geq \beta$ then
            $\mathbf{Q} = \mathcal{Q}_{k+1} \bigcup \mathbf{Q}$ Add the second expert.
        end if
    end if
end for
```

4. Theoretical Analysis

Inspired by the domain adaption theory [8], we develop a new theoretical analysis for the forgetting behaviour of the models under TFCL and provide theoretical guarantees for the proposed SEDEM. We first give several key definitions and notations as follows :

Definition 1 ***(The distribution of the data stream.)*** *For a given data stream $\mathcal{V} = \bigcup_{j=1}^{n} \mathcal{B}_j^r$, let $\mathbb{P}_{\mathbf{x}^r}$ representing the probabilistic representation of $\mathcal{D}_r^S$. Let $\mathbb{P}_i$ represent the distribution of all previously learnt data batches $\{\mathcal{B}_1^r, \cdots, \mathcal{B}_i^r\}$ drawn from $\mathcal{V}$ at $\mathcal{S}_i$.*

Definition 2 ***(The model risk and $d_{\mathcal{H}\triangle\mathcal{H}}$ distance.)*** *Let $\mathcal{H}$ be a hypothesis space with d Vapnik–Chervonenkis (VC) dimensions. For a given distribution $\mathbb{P}_{\mathbf{x}^r}$, the risk of a model $h \in \mathcal{H}$ is defined as $\mathcal{E}\big(h, \mathbb{P}_{\mathbf{x}^r}\big) \triangleq \mathbb{E}_{\{\mathbf{x},y\}\sim\mathbb{P}_{\mathbf{x}^r}}\big[\tau\big(y, h(\mathbf{x})\big)\big]$ where $\tau \colon \mathcal{Y} \times \mathcal{Y} \to [0,1]$ is the loss function. For two given distributions $\mathbb{P}_{\mathbf{x}^r}$ and $\mathbb{P}_i$, the $d_{\mathcal{H}\triangle\mathcal{H}}$ distance between them is defined as :*

$$\begin{aligned}d_{\mathcal{H}\triangle\mathcal{H}}\big(\mathbb{P}_{\mathbf{x}^r}(\mathbf{x}), \mathbb{P}_i(\mathbf{x})\big) \triangleq \sup_{(h,h')\in\mathcal{H}^2} \Big| \mathcal{E}\big(h, h', \mathbb{P}_{\mathbf{x}^r}(\mathbf{x})\big) \\ - \mathcal{E}\big(h, h', \mathbb{P}_i(\mathbf{x})\big) \Big| ,\end{aligned} \tag{10}$$

Methods	Split MNIST	Split CIFAR10	Split CIFAR100
finetune*	19.75 ± 0.05	18.55 ± 0.34	3.53 ± 0.04
MIR*	93.20 ± 0.36	42.80 ± 2.22	20.00 ± 0.57
GEM*	93.25 ± 0.36	24.13 ± 2.46	11.12 ± 2.48
iCARL*	83.95 ± 0.21	37.32 ± 2.66	10.80 ± 0.37
ER + GMED†	82.67 ± 1.90	34.84 ± 2.20	20.93 ± 1.60
ER$_a$ + GMED†	82.21 ± 2.90	47.47 ± 3.20	19.60 ± 1.50
reservoir*	92.16 ± 0.75	42.48 ± 3.04	19.57 ± 1.79
GSS*	92.47 ± 0.92	38.45 ± 1.41	13.10 ± 0.94
CoPE-CE*	91.77 ± 0.87	39.73 ± 2.26	18.33 ± 1.52
CoPE*	93.94 ± 0.20	48.92 ± 1.32	21.62 ± 0.69
CURL*	92.59 ± 0.66	-	-
CNDPM	95.36 ± 0.18	48.76 ± 0.28	22.52 ± 1.26
WGF-SVGD	-	47.90 ± 2.50	19.90 ± 2.30
Dynamic-OCM	94.02 ± 0.23	49.16 ± 1.52	21.79 ± 0.68
SEDEM	**98.35** ± 0.15	**55.27** ± 1.32	**24.85** ± 1.16

Table 1. Classification accuracy, representing the average of five independent runs, for the continuous learning of three datasets. * and † denote the results cited from [13] and [24], respectively.

where $\{h, h'\} \in \mathcal{H}^2$, $\mathbb{P}_{\mathbf{x}^r}(\mathbf{x})$ *is the marginal of* $\mathbb{P}_{\mathbf{x}^r}$, *and*

$$\mathcal{E}\big(h, h', \mathbb{P}_{\mathbf{x}^r}\big) \triangleq \mathbb{E}_{\{\mathbf{x},y\}\sim\mathbb{P}_{\mathbf{x}^r}}\big[\tau\big(h'(\mathbf{x}), h(\mathbf{x})\big)\big]. \qquad (11)$$

Assumption 1 *We assume that* $\mathbf{Q} = \{\mathcal{Q}_1, \cdots, \mathcal{Q}_c\}$ *has trained* c *experts at* $\mathcal{S}_i$. *Let* $\mathcal{C}_{a_j}$ *denote a memory buffer used for training the* j*-th expert* $\mathcal{Q}_j$. *The evaluation of SEDEM can be implemented by a single model* h *trained on all memory buffers* $\{\mathcal{C}_{a_1}, \cdots, \mathcal{C}_{a_{c-1}}, \mathcal{C}_i\}$.

Based on the above definitions, we provide the theoretical guarantee for SEDEM.

Theorem 1 ***(Theoretical guarantee.)*** *Based on Assumption 1 we derive a Generalization Bound (GB) with probability (at least* $1-\delta$*) at* $\mathcal{S}_i$ *:*

$$\begin{aligned}\mathcal{E}\big(h, \mathbb{P}_i\big) \leq\, & \mathcal{E}\big(h, h_{\mathcal{C}_{a_1,\cdots,a_{c-1}}\otimes\mathcal{C}_i}, \mathbb{P}_{\mathcal{C}_{a_1,\cdots,a_{c-1}}\otimes\mathcal{C}_i}\big) \\ & + \frac{1}{2} d_{\mathcal{H}\triangle\mathcal{H}}(\mathcal{R}_{\mathbb{P}_i}, \mathcal{R}_{\mathcal{C}_{a_1,\cdots,a_{c-1}}\otimes\mathcal{C}_i}) \\ & + 4\sqrt{\frac{2d\log(2m') + \log(\frac{2}{\delta})}{m'}} \\ & + \mathcal{L}_{\text{Error}}\big(\mathbb{P}_i, \mathbb{P}_{\mathcal{C}_{a_1,\cdots,a_{c-1}}\otimes\mathcal{C}_i}\big)\,, \end{aligned} \qquad (12)$$

where $\mathcal{R}_{\mathbb{P}_i}$ *and* $\mathcal{R}_{\mathcal{C}_{a_1,\cdots,a_{c-1}}\otimes\mathcal{C}_i}$ *are the set of* m' *unlabelled samples drawn from* $\mathbb{P}_i$ *and* $\mathbb{P}_{\mathcal{C}_{a_1,\cdots,a_{c-1}}\otimes\mathcal{C}_i}$, *respectively.* $\mathbb{P}_{\mathcal{C}_{a_1,\cdots,a_{c-1}}\otimes\mathcal{C}_i}$ *represents the distribution of all memory buffers and* $h_{\mathcal{C}_{a_1,\cdots,a_{c-1}}\otimes\mathcal{C}_i}$ *is the true labelling function that always returns the true labels for samples from* $\mathbb{P}_{\mathcal{C}_{k_1,\cdots,k_{c-1}}\otimes\mathcal{C}_i}$. $\mathcal{L}_{\text{Error}}\big(\mathbb{P}_i, \mathbb{P}_{\mathcal{C}_{a_1,\cdots,k_{c-1}}\otimes\mathcal{C}_i}\big)$ *is the optimal error for* $\mathbb{P}_i$ *and* $\mathbb{P}_{\mathcal{C}_{a_1,\cdots,a_{c-1}}\otimes\mathcal{C}_i}$. d *is the Vapnik–Chervonenkis dimension. The detailed proof is provided in* ***Appendix-A2*** *from SM.*

Methods	Split MiniImageNet
MIR+GMED	26.50 ± 1.3
MIR	25.21 ± 2.2
ER$_a$	25.92 ± 1.2
ER + GMED	27.27 ± 1.8
CNDPM	27.97 ± 2.3
Dynamic-OCM	26.55 ± 2.1
SEDEM	**29.57** ± 1.9

Table 2. Classification accuracy for 20 runs when testing various models on Split MiniImageNet.

Methods	Split MNIST	Split CIFAR10	Split MImageNet
finetune	21.53 ± 0.1	20.69 ± 2.4	3.05 ± 0.6
ER	79.74 ± 4.0	37.15 ± 1.6	26.47 ± 2.3
MIR	84.80 ± 1.9	38.70 ± 1.7	25.83 ± 1.5
ER + GMED	82.73 ± 2.6	40.57 ± 1.7	28.20 ± 0.6
MIR+GMED	86.17 ± 1.7	41.22 ± 1.1	26.86 ± 0.7
CNDPM	88.23 ± 1.6	42.62±1.3	26.89 ± 1.2
SEDEM	**91.24** ± 1.2	**44.68** ± 1.5	**29.16** ± 1.1

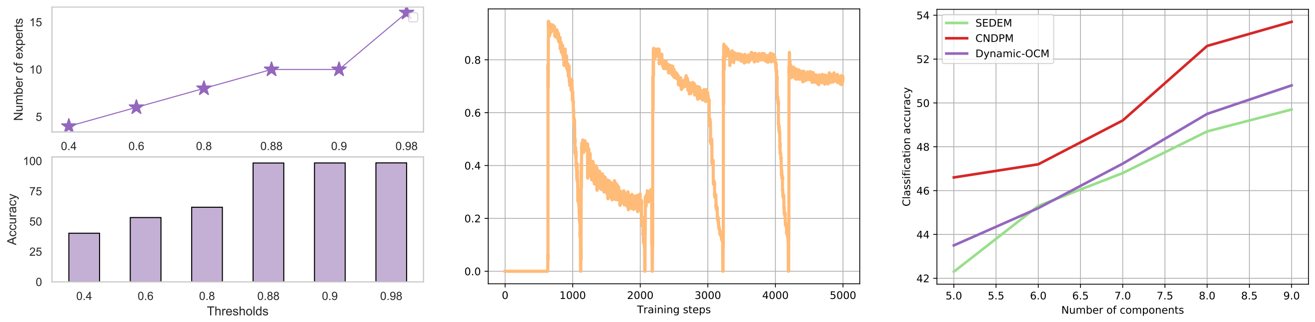
Table 3. The classification accuracy of five runs for various models over data streams with fuzzy task boundaries.

Remark. We have several observations from **Theorem 1** : (1) The $d_{\mathcal{H}\triangle\mathcal{H}}$ distance in Eq. (10) is crucial for the performance of SEDEM. Based on Assumption 1, the knowledge diversity among experts can allow $\mathbb{P}_{\mathcal{C}_{a_1,\cdots,a_{c-1}}\otimes\mathcal{C}_i}$ to represent more underlying data distributions of $\mathcal{V}$, which would decrease $d_{\mathcal{H}\triangle\mathcal{H}}$ and thus improve the performance; (2) Existing models fail to achieve a low GB in Eq. (12) with a minimum number of experts since they would train statistically overlapping experts. In contrast, the proposed expansion criterion (Eq. (1)) in SEDEM can promote the necessary statistical diversity among the probabilistic representations of the experts, which leads to a better trade-off between the model's complexity and performance.

5. Experiments

5.1. Setting and Dataset

Datasets : Split MNIST divides MNIST [31] containing 60k training samples, into five tasks according to images of pairs of digits in their increasing order [13]. **Split CIFAR10** splits CIFAR10 [30] into five tasks where each task consists of images from two different classes [13]. **Split CIFAR100** divides CIFAR100 into 20 tasks where each task has 2500 samples from 5 different classes [35]. We adapt the network architecture according to [13]. We set the maximum memory size λ as 2000, 1000, and 5000 for Split MNIST, Split CIFAR10, and Split CIFAR100, respectively. We set the batch size as 10, and β in Eq. (1) as 0.90, 0.15 and 0.16 for Split MNIST, Split CIFAR10, and Split CIFAR100, respectively. We employ classification accuracy as the performance criterion. We provide the detailed setting in **Appendix-C1 from SM**.



(a) Changing β.

(b) Dynamic expansion process

(c) Comparison to baselines

Figure 3. (a) The performance and number of experts of SEDEM trained under Split MNIST when changing β. (b) The expansion signals of the proposed SEDEM under Split CIFAR10. (c) Comparison to other dynamic expansion models under the same number of experts.

Methods	Split M-S	Parameters	Split M-C	Parameters
ER	10.89	208M	15.28	161M
ER + GMED	16.23	208M	21.26	161M
CoPE	22.45	208M	26.85	161M
CNDPM	47.64	237M	66.25	185M
SEDEM	**56.62**	189M	**78.56**	157M

Table 4. Classification accuracy of various models in the cross-domain setting.

Baselines. We consider the following baselines for TFCL : GSS [5], MIR [3], Incremental Classifier and Representation Learning (iCARL) [44], Reservoir [51], Dynamic-Online Cooperative Memorization (OCM) [65], CURL [43], Gradient Episodic Memory (GEM) [35], CNDPM [33], CoPE [13], ER + GMED and ER$_a$ + GMED [24] where ER is the Experience Replay (ER) [45] and ER$_a$ is the model using ER and data augmentation. More information about baselines is provided in **Appendix-C2** from SM.

5.2. Single Data Domain Classification

We investigate the effectiveness of the proposed Self-Evolved Dynamic Expansion Model (SEDEM) on the classification of a single data domain. We employ the Adam optimization algorithm [27] with a learning rate of 0.00001 and the average accuracy and standard deviation from five independent runs for Split MNIST, Split CIFAR10 and Split CIFAR100 are reported in Tab. 1. We observe that the dynamic expansion models such as CURL and CNDPM usually outperform most static models due to their scalability and generalization performance. The proposed SEDEM achieves the best performance in each dataset when compared to the baselines, demonstrating its effectiveness under the TFCL.

We also investigate the performance of various models on a dataset with more complex images, such as Split MiniImageNet [50]. Split MiniImageNet divides MiniImageNet into 20 tasks where each task contains the images of five classes [3]. We report the classification accuracy on Split MiniImageNet in Tab. 2. The number of experts of SEDEM for Split MNIST, Split CIFAR10, Split CIFAR100 and Split MImageNet is of 10, 6, 7 and 6, respectively. These results show that SEDEM outperforms all baselines on the challenging Split MiniImageNet dataset. We also provide the results of the model's complexity in **Appendix-E** from SM, which demonstrates that SEDEM employs fewer parameters compared with CNDPM.

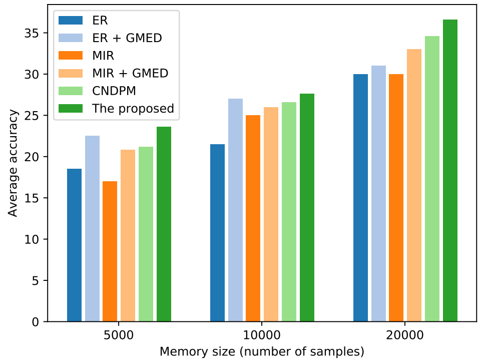


Figure 4. The performance on Split MinImageNet, achieved by various models when changing the memory buffer size.

5.3. Fuzzy Task Boundaries

In the real-world environment, a model would receive a data stream with fuzzy task boundaries [33]. In this section, we investigate the effectiveness of various models for this challenging setting. Following from [33], we swap randomly samples between two tasks for each data stream. We report the results on Split MNIST, Split CIFAR10 and Split MImageNet in Tab. 3. The results show that the proposed SEDEM outperforms other baselines by a large margin under the fuzzy task boundaries setting.

5.4. Classification for Multiple Data Domains

We evaluate the performance in a more challenging CL setting, when learning multiple data domains. We create a data stream Split M-S which combines Split MNIST and Split SVHN. Similarly, Split M-C combines Split MNIST

Methods	Split MNIST	Split CIFAR10	Split CIFAR100
SEDEM-CoPE	97.63	50.82	23.75
SEDEM-MIR	97.65	50.38	23.62
SEDEM-reservoir	97.98	50.35	22.97
SEDEM-NoRS	97.29	50.14	22.85
SEDEM-B1	97.42	52.98	22.74
SEDEM	**98.35**	**55.27**	**24.85**

Table 5. Assessing the proposed data selection in SEDEM.

and Split CIFAR10. The memory buffer filled according to Eq. (4), is of size λ=1000 for these databases. The results from Tab. 4 where 'Parameters' represents the number of parameters for each model. We consider a large network architecture for the static model for a fair comparison. We can observe that the proposed SEDEM significantly outperforms both the static and DEM model on the multi-domain setting while using fewer parameters.

5.5. Ablation Study

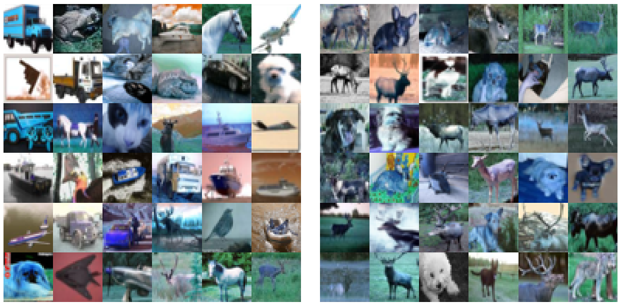
In this section, we evaluate the effectiveness of each module in SEDEM by performing a wide range of ablation studies (See more results in **Appendix-D from SM**).

Dynamic expansion : In Fig. 3-a we plot the performance and the number of experts for SEDEM trained on Split MNIST when changing the threshold β from Eq. (1). The results show that a large β encourages adding more experts, improving the performance. Meanwhile, a small β creates fewer experts, resulting in lower performance.

Sample selection results : We adapt other sample selection methods including CoPE, MIR and reservoir for SEDEM, resulting in several baselines such as SEDEM-CoPE, SEDEM-MIR and SEDEM-reservoir. We also consider SEDEM-NoRS which does not employ the sample selection mechanism. The classification accuracy for all these methods is provided in Tab. 5 where we observe that the proposed sample selection approach for SEDEM leads to a better performance than other methods.

Effects of the proposed DEKMM : We consider DEKMM with SEDEM-B1 which does not employ DEKMM and the results are provided in Tab. 5. These results show that the proposed DEKMM can further improve the performance of SEDEM on all datasets, demonstrating the positive forward transfer achieved by using DEKMM.

Dynamic expansion process: In Fig. 3-b, we plot the expansion signals (Left-Hand-Side (LHS) of Eq. (1)) in each training step, estimated by the proposed SEDEM trained on Split CIFAR10, where the expansion signal is zero when SEDEM has only a single expert. We can observe that the proposed SEDEM gives a low score (LHS of Eq. (1)) when facing the data distribution shift. Such a low score indicates that the current expert had learnt sufficient novel knowledge, and thus SEDEM performs the expansion to preserve the learnt knowledge while employing the additional capacity to handle the data distribution shift. We provide additional results for the dynamic expansion process of the SEDEM in **Appendix-D.1** from SM. These results show that each expert in the proposed SEDEM learns almost a unique underlying data distribution, which demonstrates that the proposed dynamic expansion mechanism allows the SEDEM to adapt to the data distribution shift well during the training. The performance, achieved by various dynamic expansion models with different number of experts on Split CIFAR10 is provided in Fig. 3-c. The proposed approach significantly outperforms other baselines when considering the same number of experts.



(a) Real samples (b) Memorized samples

Figure 5. Real and memorized samples randomly drawn from all previously seen data batches and the memory buffer.

Memory buffer size : We evaluate the performance of various models under different memory buffer sizes (λ) on the Split MinImageNet and plot the results in Fig. 4. The dynamic expansion models achieve better results than the static models in almost any memory buffer setting. The proposed SEDEM outperforms all baselines in each dataset, even when using an extremely small-scale memory buffer.

Samples in the memory buffer : We randomly draw training samples from all previously seen data batches at a certain training step ($\mathcal{S}_{4500}$). Those samples can represent the knowledge of all previously learnt experts. We also randomly draw samples from the memory buffer, which can represent the knowledge of the current expert. We plot those samples in Fig. 5, where we can observe that most memorized samples have different visual concepts, even when compared to the real samples. These results show that the proposed sample selection approach encourages the memory buffer to store novel samples.

6. Conclusion

This paper proposes a novel model for TFCL, namely the Self-Evolved Dynamic Expansion Model (SEDEM) which evaluates the diversity among experts as expansion signals, ensuring an appropriate model size. In addition, we propose a novelty-aware sample selection approach to further improve the performance. Moreover, we propose reusing the previously learnt representations for enhancing the forward knowledge transfer when learning new data.

References

[1] A. Achille, T. Eccles, L. Matthey, C. Burgess, N. Watters, A. Lerchner, and I. Higgins. Life-long disentangled representation learning with cross-domain latent homologies. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 9873–9883, 2018. 2

[2] Hongjoon Ahn, Sungmin Cha, Donggyu Lee, and Taesup Moon. Uncertainty-based continual learning with adaptive regularization. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 4394–4404, 2019. 2

[3] Rahaf Aljundi, Eugene Belilovsky, Tinne Tuytelaars, Laurent Charlin, Massimo Caccia, Min Lin, and Lucas Page-Caccia. Online continual learning with maximal interfered retrieval. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 11872–11883, 2019. 2, 3, 7

[4] Rahaf Aljundi, Klaas Kelchtermans, and Tinne Tuytelaars. Task-free continual learning. In *Proc. of IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 11254–11263, 2019. 1, 3, 4

[5] R. Aljundi, M. Lin, B. Goujaud, and Y. Bengio. Gradient based sample selection for online continual learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 11817–11826, 2019. 2, 3, 4, 7

[6] Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. Rainbow memory: Continual learning with a memory of diverse samples. In *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8218–8227, 2021. 1, 2

[7] Jihwan Bang, Hyunseo Koh, Seulki Park, Hwanjun Song, Jung-Woo Ha, and Jonghyun Choi. Online continual learning on a contaminated data stream with blurry task boundaries. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9275–9284, 2022. 2

[8] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010. 5

[9] Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2l: Contrastive continual learning. In *Proc. of the IEEE/CVF International Conference on Computer Vision (CVPR)*, pages 9516–9525, 2021. 2

[10] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. In *Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1812.00420*, 2019. 2

[11] A. Chaudhry, M. Rohrbach, M. Elhoseiny, T. Ajanthan, P. Dokania, P. H. S. Torr, and M.'A. Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019. 2

[12] C. Cortes, X. Gonzalvo, V. Kuznetsov, M. Mohri, and S. Yang. Adanet: Adaptive structural learning of artificial neural networks. In *Proc. of Int. Conf. on Machine Learning (ICML), vol. PMLR 70*, pages 874–883, 2017. 3

[13] Matthias De Lange and Tinne Tuytelaars. Continual prototype evolution: Learning online from non-stationary data streams. In *Proc. of the IEEE/CVF International Conference on Computer Vision (CVPR)*, pages 8250–8259, 2021. 3, 6, 7

[14] Danruo Deng, Guangyong Chen, Jianye Hao, Qiong Wang, and Pheng-Ann Heng. Flattening sharpness for dynamic gradient projection memory benefits continual learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:18710–18721, 2021. 2

[15] Mohammad Mahdi Derakhshani, Xiantong Zhen, Ling Shao, and Cees Snoek. Kernel continual learning. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 2621–2631. PMLR 139, 2021. 2

[16] Sayna Ebrahimi, Franziska Meier, Roberto Calandra, Trevor Darrell, and Marcus Rohrbach. Adversarial continual learning. In *Proc. European Conference on Computer Vision (ECCV), vol. LNCS 12356*, pages 386–402, 2020. 1, 3

[17] Evgenii Egorov, Anna Kuzina, and Evgeny Burnaev. BooVAE: Boosting approach for continual learning of VAE. *Advances in Neural Information Processing Systems*, 34:17889–17901, 2021. 2

[18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Proc. Advances in Neural Inf. Proc. Systems (NIPS)*, pages 2672–2680, 2014. 2

[19] Yanan Gu, Xu Yang, Kun Wei, and Cheng Deng. Not just selection, but exploration: Online class-incremental continual learning via dual view consistency. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7442–7451, 2022. 2

[20] Yiduo Guo, Bing Liu, and Dongyan Zhao. Online continual learning through mutual information maximization. In *Proc. International Conference on Machine Learning (ICML)*, pages 8109–8126. PMLR 162, 2022. 2

[21] Christian Henning, Maria Cervera, Francesco D'Angelo, Johannes Von Oswald, Regina Traber, Benjamin Ehret, Seijin Kobayashi, Benjamin F Grewe, and João Sacramento. Posterior meta-replay for continual learning. *Advances in Neural Information Processing Systems*, 34:14135–14149, 2021. 2

[22] Ching-Yi Hung, Cheng-Hao Tu, Cheng-En Wu, Chien-Hung Chen, Yi-Ming Chan, and Chu-Song Chen. Compacting, picking and growing for unforgetting continual learning. In *Advances in Neural Information Processing Systems*, pages 13647–13657, 2019. 3

[23] Julio Hurtado, Alain Raymond, and Alvaro Soto. Optimizing reusable knowledge for continual learning via metalearning. *Advances in Neural Information Processing Systems*, 34:14150–14162, 2021. 2

[24] Xisen Jin, Arka Sadhu, Junyi Du, and Xiang Ren. Gradient-based editing of memory examples for online task-free continual learning. In *Advances in Neural Information Processing Systems (NeurIPS), arXiv preprint arXiv:2006.15294*, 2021. 6, 7

[25] Haeyong Kang, Rusty John Lloyd Mina, Sultan Rizky Hikmawan Madjid, Jaehong Yoon, Mark Hasegawa-Johnson, Sung Ju Hwang, and Chang D Yoo. Forget-free continual learning with winning subnetworks. In *Proc. International Conference on Machine Learning (ICML)*, pages 10734–10750. PMLR 162, 2022. 3

[26] Ronald Kemker, Marc McClure, Angelina Abitino, Tyler Hayes, and Christopher Kanan. Measuring catastrophic forgetting in neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 2

[27] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1412.6980*, 2015. 7

[28] D. P. Kingma and M. Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2, 5

[29] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell. Overcoming catastrophic forgetting in neural networks. *Proc. of the National Academy of Sciences (PNAS)*, 114(13):3521–3526, 2017. 1, 2

[30] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Univ. of Toronto, 2009. 6

[31] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11):2278–2324, 1998. 6

[32] Sebastian Lee, Sebastian Goldt, and Andrew Saxe. Continual learning in the teacher-student setup: Impact of task similarity. In *International Conference on Machine Learning*, pages 6109–6119. PMLR, 2021. 2

[33] Soochan Lee, Junsoo Ha, Dongsu Zhang, and Gunhee Kim. A neural Dirichlet process mixture model for task-free continual learning. In *Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:2001.00689*, 2020. 2, 3, 7

[34] Z. Li and D. Hoiem. Learning without forgetting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017. 3

[35] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, pages 6467–6476, 2017. 2, 6, 7

[36] James Martens and Roger B. Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 2408–2417. JMLR.org, 2015. 2

[37] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. 1

[38] Cuong V Nguyen, Yingzhen Li, Thang D Bui, and Richard E Turner. Variational continual learning. In *Proc. of Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1710.10628*, 2018. 2

[39] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019. 1

[40] R. Polikar, L. Upda, S. S. Upda, and Vasant Honavar. Learn++: An incremental learning algorithm for supervised neural networks. *IEEE Trans. on Systems Man and Cybernetics, Part C*, 31(4):497–508, 2001. 3

[41] Qi Qin, Wenpeng Hu, Han Peng, Dongyan Zhao, and Bing Liu. Bns: Building network structures dynamically for continual learning. *Advances in Neural Information Processing Systems*, 34:20608–20620, 2021. 3

[42] J. Ramapuram, M. Gregorova, and A. Kalousis. Lifelong generative modeling. In *Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1705.09847*, 2017. 2

[43] Dushyant Rao, Francesco Visin, Andrei A. Rusu, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Continual unsupervised representation learning. In *Proc. Neural Inf. Proc. Systems (NIPS)*, pages 7645–7655, 2019. 2, 3, 7

[44] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. iCaRL: Incremental classifier and representation learning. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2001–2010, 2017. 2, 7

[45] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy P. Lillicrap, and Gregory Wayne. Experience replay for continual learning. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pages 348–358, 2019. 7

[46] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 1, 3

[47] Yujun Shi, Li Yuan, Yunpeng Chen, and Jiashi Feng. Continual learning via bit-level information preserving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16674–16683, 2021. 2

[48] H. Shin, J. K. Lee, J. Kim, and J. Kim. Continual learning with deep generative replay. In *Advances in Neural Inf. Proc. Systems (NIPS)*, pages 2990–2999, 2017. 2

[49] Rishabh Tiwari, Krishnateja Killamsetty, Rishabh Iyer, and Pradeep Shenoy. Gcr: Gradient coreset based replay buffer selection for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 99–108, June 2022. 2

[50] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. *Advances in neural information processing systems (NIPS)*, 29:3637–3645, 2016. 7

[51] Jeffrey S Vitter. Random sampling with a reservoir. *ACM Transactions on Mathematical Software (TOMS)*, 11(1):37–57, 1985. 7

[52] Liyuan Wang, Mingtian Zhang, Zhongfan Jia, Qian Li, Chenglong Bao, Kaisheng Ma, Jun Zhu, and Yi Zhong. Afec: Active forgetting of negative transfer in continual learning. *Advances in Neural Information Processing Systems*, 34:22379–22391, 2021. 2

[53] Shipeng Wang, Xiaorong Li, Jian Sun, and Zongben Xu. Training networks in null space of feature covariance for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 184–193, 2021. 1, 2

[54] Zhenyi Wang, Li Shen, Le Fang, Qiuling Suo, Tiehang Duan, and Mingchen Gao. Improving task-free continual learning by distributionally robust memory evolution. In *International Conference on Machine Learning*, pages 22985–22998. PMLR, 2022. 2

[55] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022. 2

[56] Yeming Wen, Dustin Tran, and Jimmy Ba. BatchEnsemble: an alternative approach to efficient ensemble and lifelong learning. In *Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:2002.06715*, 2020. 3

[57] T. Xiao, J. Zhang, K. Yang, Y. Peng, and Z. Zhang. Error-driven incremental learning in deep convolutional neural network for large-scale image classification. In *Proc. of ACM Int. Conf. on Multimedia*, pages 177–186, 2014. 3

[58] Qingsen Yan, Dong Gong, Yuhang Liu, Anton van den Hengel, and Javen Qinfeng Shi. Learning bayesian sparse networks with full experience replay for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 109–118, 2022. 2

[59] Fei Ye and Adrian G. Bors. Learning latent representations across multiple data domains using lifelong VAEGAN. In *Proc. European Conf. on Computer Vision (ECCV), vol. LNCS 12365*, pages 777–795, 2020. 2

[60] Fei Ye and Adrian G. Bors. Lifelong learning of interpretable image representations. In *Proc. Int. Conf. on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–6, 2020. 2

[61] Fei Ye and Adrian G. Bors. Lifelong infinite mixture model based on knowledge-driven Dirichlet process. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, pages 10695–10704, 2021. 2

[62] Fei Ye and Adrian G. Bors. Lifelong mixture of variational autoencoders. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–14, 2021. 2

[63] Fei Ye and Adrian G. Bors. Lifelong teacher-student network learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2

[64] Fei Ye and Adrian G. Bors. Lifelong twin generative adversarial networks. In *Proc. IEEE Int. Conf. on Image Processing (ICIP)*, pages 1289–1293, 2021. 2

[65] Fei Ye and Adrian G. Bors. Continual variational autoencoder learning via online cooperative memorization, 2022. 7

[66] Fei Ye and Adrian G Bors. Dynamic self-supervised teacher-student network learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5731–5748, 2022. 2

[67] Fei Ye and Adrian G Bors. Task-free continual learning via online discrepancy distance learning. *Advances in Neural Information Processing Systems*, 35:23675–23688, 2022. 2

[68] Fei Ye and Adrian G Bors. Continual variational autoencoder via continual generative knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 10918–10926, 2023. 2

[69] Fei Ye and Adrian G Bors. Learning dynamic latent spaces for lifelong generative modelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 10891–10899, 2023. 2

[70] Fei Ye and Adrian G Bors. Lifelong dual generative adversarial nets learning in tandem. *IEEE Transactions on Cybernetics*, 2023. 2

[71] Fei Ye and Adrian G Bors. Lifelong generative adversarial autoencoder. *IEEE Transactions on Neural Networks and Learning Systems*, 2023. 2

[72] Haiyan Yin, Ping Li, et al. Mitigating forgetting in online continual learning with neuron calibration. *Advances in Neural Information Processing Systems*, 34:10260–10272, 2021. 2

[73] M. Zhai, L. Chen, F. Tung, J He, M. Nawhal, and G. Mori. Lifelong GAN: Continual learning for conditional image generation. In *Proc. of the IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, pages 2759–2768, 2019. 2, 3

[74] Guanyu Zhou, Kihyuk Sohn, and Honglak Lee. Online incremental feature learning with denoising autoencoders. In *Artificial intelligence and statistics*, pages 1453–1461, 2012. 3