Proceedings Paper:
Ye, Fei and Bors, Adrian Gheorghe orcid.org/0000-0001-7838-0021 (2022) Continual Variational Autoencoder Learning via Online Cooperative Memorization. In: Proceedings of the European Conference on Computer Vision (ECCV). Lecture Notes in Computer Science (LNCS) . Springer , 531–549.

https://doi.org/10.1007/978-3-031-20050-2_31

Continual Variational Autoencoder Learning via Online Cooperative Memorization

Fei Ye and Adrian G. Bors

Department of Computer Science, University of York, York YO10 5GH, UK
{fy689,adrian.bors}@york.ac.uk

Abstract. Due to their inference, data representation and reconstruction properties, Variational Autoencoders (VAE) have been successfully used in continual learning classification tasks. However, their ability to generate images with specifications corresponding to the classes and databases learned during Continual Learning (CL) is not well understood and catastrophic forgetting remains a significant challenge. In this paper, we firstly analyze the forgetting behaviour of VAEs by developing a new theoretical framework that formulates CL as a dynamic optimal transport problem. This framework proves approximate bounds to the data likelihood without requiring the task information and explains how the prior knowledge is lost during the training process. We then propose a novel memory buffering approach, namely the Online Cooperative Memorization (OCM) framework, which consists of a Short-Term Memory (STM) that continually stores recent samples to provide future information for the model, and a Long-Term Memory (LTM) aiming to preserve a wide diversity of samples. The proposed OCM transfers certain samples from STM to LTM according to the information diversity selection criterion without requiring any supervised signals. The OCM framework is then combined with a dynamic VAE expansion mixture network for further enhancing its performance.

Keywords: VAE, Continual learning, Lifelong generative modelling

1 Introduction

One desired capability for an artificial intelligence system is to continually learn novel concepts without forgetting the knowledge learnt in the past. However, existing artificial systems are far away from such capabilities, characteristic of living organisms. A deep learning model which can recover the training data from a low-dimensional latent code space is the Variational Autoencoder (VAE) [25]. VAEs have been widely used in image synthesis [60, 62], semi-supervised learning [1, 63] and for image-to-image translation [38]. However, similar to other deep learning systems, VAEs suffer from degenerated performance when it is trained successively with new tasks, which is a result of catastrophic forgetting [42].

Existing works to relieve VAE's forgetting can be summarized as two categories. The first would usually train a generator [2, 44, 49], or store a few past

learnt samples [39] in a memory buffer which replays old samples together with learning new tasks to optimize the model. The methods from the second category would focus on dynamically adding new VAE components into a mixture model to adapt to the data distribution shift [35, 45] in which prior knowledge is preserved in the frozen network parameters and structures. These approaches have been extended for the case when the model is trained on non-stationary data streams without knowing the task information, a mechanism called Task Free Continual Learning (TFCL) [4, 5]. However, the theoretical analysis for VAE's forgetting behaviour under TFCL has not been studied before.

In recent years, some studies have provided the theoretical analysis for continual learning from different perspectives including the NP-hard problem [27], risk bound [58, 64], Teacher-Student framework [34, 57] and game theory [43]. However, all these approaches require strong assumptions such as clearly defining the task identities, which is not applicable when the task information is missing. In this paper, we bridge this gap by developing a new theoretical framework which formulates TFCL as a dynamic optimal transport (OT) problem, and derives the approximate bounds on the data likelihood. The motivation behind OT is twofold : 1) OT models evaluate distances between pairs of probability density functions [8] and can be used for deriving the approximate bound to the data likelihood (See Section 4); 2) OT can be estimated by employing sampling [18], which is suitable for analysis and verification. The proposed theoretical analysis also highlights that the sample diversity in the memory used for training is crucial for overcoming forgetting and would not require the category information.

Another contribution of this study, inspired by the above mentioned theoretical analysis, is to develop a new memorization approach aiming to store diverse samples for training a VAE through the TFCL. Other approaches have proposed diversifying the information for memorization by evaluating the similarity on the gradient information [3] or by assigning balanced samples to memory buffers according to their categories' information [6, 13]. However, most of these prior approaches require to access supervised signals, which are not available in unsupervised learning. Additionally, these approaches do not have theoretical guarantees and also ignore the data stream future information in the sample selection. Knowing both the past and future information was shown to improve time series prediction [22] and would be helpful for the sample selection.

In this paper, we address the aforementioned problems by : 1) Proposing a new learning paradigm called Online Cooperative Memorization (OCM) which consists of three components: a Long-Term Memory (LTM), a Short-Term Memory (STM) and a model (Learner). OCM implements a memorization mechanism which transfers the temporary information from the STM to LTM, according to a certain criterion. 2) A kernel-based information importance criterion for evaluating the similarity among the data stored in the STM for selecting diverse characteristic samples for LTM, without requiring a class label. The kernel evaluation of the similarity of a pair of data samples [15], defined as an inner product of the latent representations of each pair of the data stored in the memory, is shown to be efficient. This procedure ensures achieving an appropriate diversifi-

cation among the samples stored in the LTM. We summarize our contributions as follows : 1) Our work is the first to provide theory insights for the forgetting behaviour of VAE under TFCL. 2) We propose the Online Cooperative Memorization (OCM) that can be used in any VAE variant with minimal modification and can also be extended to a dynamic expansion mixture approach to further enhance performance. 3) We propose a new sample selection approach for dynamically transferring selected samples from the STM to LTM without requiring any supervised signal. To our best knowledge, this is the first work to explore the kernel-based distance for the sample selection under TFCL. 4) The proposed sample selection approach can be used in both supervised and unsupervised learning without modifying the selection strategy.

2 Related work

Continual learning. One of the most popular approaches is to use a regularization loss within the optimization procedure [14, 21, 23, 26, 36, 41, 47, 52, 56], where the network parameters which are important to the past learnt data are re-weighted when learning a new task, in order to attempt to preserve past knowledge. Other approaches would employ a small buffer to store a few past data [3, 10, 53] or would train a generator as a generative replay network that provides pseudo data samples for the future task learning [2, 44, 45, 49, 57–59, 66, 69]. However, these approaches can not guarantee the optimal performance on the past task since stored or generated samples can not represent the true underlying data distributions [64]. This issue can be solved by storing the information of past samples into the network's parameters which are then frozen when learning novel tasks [35, 64, 65, 67, 68].

Task free continual learning. Recent works have driven the attention to a more challenging scenario where task boundaries are unknown. Most approaches would focus on the sample selection approach that stores certain samples into a buffer to train the model. This approach was firstly investigated in [5] for training a classifier under TFCL and for training both classifiers and VAEs [4] using a new retrieving mechanism selecting called the Maximal Interfered Retrieval (MIR). The Gradient Sample Selection (GSS) [3] formulates the sample selection as a constrained optimization reduction. More recently, a Learner-Evaluator framework, called the Continual Prototype Evolution (CoPE) [13] stores the same number of samples for each class in the memory to enforce the balance replay. Different from these approaches, the proposed OCM does not require any supervised signals for the sample selection in both supervised and unsupervised learning.

Another approach for TFCL is based on the dynamic expansion mechanism [35], called the Continual Neural Dirichlet Process Mixture (CN-DPM), which introduces Dirichlet processes for the expansion of VAE components. This expansion mechanism was combined with the generative replay into the Continual Unsupervised Representation Learning (CURL) [45], for learning the shared and task-specific representations, befitting on the clustering task.

Optimal Transport (OT). The OT aims to search for a minimal effort solution to transfer the mass from one distribution to another. OT has been recently applied in the domain adaptation problems [12, 16] and was also used in auto-encoders to provide a flexible training loss for the VAE [54]. However, these models require to fully access all samples at all times, and are failing to capture the underlying data distributions under TFCL. In this paper, we formulate TFCL as the dynamic optimal transport problem which provides a new perspective for the forgetting behaviour of VAEs. To our best knowledge, this paper is the first work to employ OT for forgetting analysis under TFCL.

3 Preliminary

In this section, we firstly introduce the background of VAEs. Then we explain how TFCL can be seen as a dynamic optimal transport problem.

3.1 The Variational Autoencoder

The VAE [25] aims to jointly optimize the observed variable $\mathbf{x}$ and their corresponding encoded latent variables $\mathbf{z}$ within an unified optimization framework by maximizing the marginal log-likelihood $\log p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}\,|\,\mathbf{z})p(\mathbf{z})\,\mathrm{d}\mathbf{z}$. This integral involves the Normal prior distribution $p(\mathbf{z})$, which is intractable to optimize since it requires access to all $\mathbf{z}$. The VAE maximizes the Evidence Lower Bound (ELBO) on $\log p_\theta(\mathbf{x})$, while the distribution $p_\theta(\mathbf{z}\,|\,\mathbf{x})$ is approximated by a variational distribution $q_\omega(\mathbf{z}\,|\,\mathbf{x})$:

$$\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega) := \mathbb{E}_{z\sim q_\omega(\mathbf{z}\,|\,\mathbf{x})}\left[\log p_\theta(\mathbf{x}\,|\,\mathbf{z})\right] - KL\left[q_\omega(\mathbf{z}\,|\,\mathbf{x})\,||\,p(\mathbf{z})\right], \tag{1}$$

where $p_\theta(\mathbf{x}\,|\,\mathbf{z})$ is the decoder parameterized by θ and $\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)$ is a lower bound to $\log p_\theta(\mathbf{x})$. $KL(\cdot)$ represents the Kullback–Leibler (KL) divergence. Eq. (1) can be further extended when considering multiple samples, as the Importance Weighted Autoencoder (IWVAE) [9] :

$$\mathcal{L}^m_{IW}(\mathbf{x};\theta,\omega) := \mathbb{E}_{z_1,\cdots,z_m\sim q_\omega(\mathbf{z}\,|\,\mathbf{x})}\left[\log \frac{1}{m}\sum_{i=1}^{m} w_i\right], \tag{2}$$

where $w_i = p_\theta(\mathbf{x},\mathbf{z}_i)/q_\omega(\mathbf{z}_i\,|\,\mathbf{x})$ and m is the number of importance samples. Since we have $\mathcal{L}^m_{IW}(\mathbf{x};\theta,\omega) > \mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)$ for $m > 1$ [9], Eq. (2) can be used as the estimator for the data likelihood [50].

3.2 Formulate TFCL as a dynamic OT problem

Learning setting. Let $\mathcal{D}^S$ be a training set over the image space $\mathcal{X} \in \mathbb{R}^d$ with d dimensions, we assume that there are N training steps $\{t_1, \cdots, t_N\}$, for the part-by-part learning of $\mathcal{D}^S$, defined as $\mathcal{D}^S = \bigcup_{i=1}^{t_N} \mathbf{X}^i_b$, where $\mathbf{X}^i_b \cap \mathbf{X}^j_b = \varnothing$ for $i \neq j$. In each training step t_i, a model only observes a small batch of images $\mathbf{X}^i_b$ drawn from $\mathcal{D}^S$, without accessing all the prior batches $\{\mathbf{X}^1_b, \cdots, \mathbf{X}^{i-1}_b\}$. Once all training steps are finished, we evaluate the model on a testing dataset $\mathcal{D}^T$ by

using two main criteria (negative log-likelihood estimation and reconstruction quality). In the following, we introduce several definitions and notations.

Definition 1. *(Memory.) Let $\mathcal{M}_i$ represent a memory data buffer updated at the step t_i and $\mathbb{P}_{m_i}$ represent the probabilistic representation of the samples drawn from $\mathcal{M}_i$. Let $\mathbb{P}_{\mathbf{x}}$ represent the probabilistic measure defined by the samples drawn from $\mathcal{D}^S$.*

Definition 2. *(Model.) Let h^i be a VAE model trained on $\mathcal{M}_i$ at t_i. Let $\mathbb{P}_{\mathbf{z}}$ be a prior distribution (Normal distribution) on the latent variable space $\mathcal{Z}$.*

Definition 3. *(Decoder.) Let $\mathrm{G}_i \colon \mathcal{Z} \to \mathcal{X}$ be a generator (decoder in the h^i model trained at t_i). $\mathrm{G}_i(\mathbf{z}) = p_\theta(\mathbf{x} \,|\, \mathbf{z})$ in h^i is implemented as the Gaussian decoder $\mathcal{N}(\mathrm{G}_i^\star(\mathbf{z}), \sigma^2 \mathbf{I}_d)$, where $\mathrm{G}_i^\star$ is a deterministic generator, $\sigma > 0$ represents a small random variation for ensuring randomness, and $\mathbf{I}_d$ is the unit vector of dimension d. Let $\mathbb{P}_{\mathrm{G}_i}$ represent the probabilistic measure formed by samples drawn through the sampling process, $\mathbf{x} \sim p_\theta(\mathbf{x} \,|\, \mathbf{z}), \mathbf{z} \sim \mathbb{P}_{\mathbf{z}}$ of h^i.*

In the generative modelling, we usually consider two probabilistic measures $\mathbb{P}_{\mathbf{x}}$ and $\mathbb{P}_{\mathrm{G}_i}$ over two distinct spaces, denoted as $\Omega_{\mathbf{x}}$ and Ω_{G_i}, respectively. Let $\mathbf{T} \colon \Omega_{\mathrm{G}_i} \to \Omega_{\mathbf{x}}$ be a transport map if satisfying $\mathbf{T} \# \mathbb{P}_{\mathrm{G}_i} = \mathbb{P}_{\mathbf{x}}$ that transforms $\mathbb{P}_{\mathrm{G}_i}$ into $\mathbb{P}_{\mathbf{x}}$. For a given arbitrary measurable cost function $\mathcal{L}$, the optimal transportation problem can be defined by Monge's formulation, expressed by :

$$\mathbf{T}^* = \arg\min_{\mathbf{T}} \int_{\Omega_{\mathrm{G}_i}} \mathcal{L}(\mathbf{x}, \mathbf{T}(\mathbf{x}))\, \mathrm{d}\mathbb{P}_{\mathrm{G}_i}(\mathbf{x})\,, s.t. \mathbf{T} \# \mathbb{P}_{\mathrm{G}_i} = \mathbb{P}_{\mathbf{x}}\,. \tag{3}$$

According to the optimal transport theory [12], the above problem is solved by the Kantorovitch formulation [24] :

$$\mathrm{W}_{\mathcal{L}}^\star(\mathbb{P}_{\mathbf{x}}, \mathbb{P}_{\mathrm{G}_i}) = \inf_{\mathbb{P}_{\mathbf{x} \times \mathrm{G}_i}} \mathbb{E}_{(\mathbf{x}^r, \mathbf{x}^g) \sim \mathbb{P}_{\mathbf{x} \times \mathrm{G}_i}} [\mathcal{L}(\mathbf{x}^r, \mathbf{x}^g)]\,, \tag{4}$$

where $\mathbb{P}_{\mathbf{x} \times \mathrm{G}_i}$ represents the set of all probabilistic couplings on $\Omega_{\mathbf{x}} \times \Omega_{\mathrm{G}_i}$ with marginals $\mathbb{P}_{\mathbf{x}}$ and $\mathbb{P}_{\mathrm{G}_i}$. $\mathbf{X}^r$ and $\mathbf{X}^g$ are the samples drawn from $\mathbb{P}_{\mathbf{x} \times \mathrm{G}_i}$. Different from the traditional OT problem, $\mathrm{W}_{\mathcal{L}}(\mathbb{P}_{\mathbf{x}}, \mathbb{P}_{\mathrm{G}_i})$ would be changed over time (when i increases) because the model is trained on the dynamically evolved memory $\mathcal{M}_i$. We call Eq. (4) as the dynamic OT problem where the optimal solution is evolved each training time t_i. Eq. (4) has an upper bound when G_i is the Gaussian decoder [8, 54] :

$$\mathrm{W}_{\mathcal{L}}^\star(\mathbb{P}_{\mathbf{x}}, \mathbb{P}_{\mathrm{G}_i}) \leq \inf_{q_\omega(\mathbf{z})} \mathbb{E}_{\mathbb{P}_{\mathbf{x}}} \mathbb{E}_{q_\omega(\mathbf{z} \,|\, \mathbf{x})} [\mathcal{L}(\mathbf{x}, \mathrm{G}_i(\mathbf{z}))]\,, \tag{5}$$

where $q_\omega(\mathbf{z})$ is the marginal distribution of $q_\omega(\mathbf{z} \,|\, \mathbf{x})$ satisfying $q_\omega(\mathbf{z}) = p(\mathbf{z})$. We implement $\mathcal{L}(\mathbf{x}, \mathrm{G}_i(\mathbf{z})) = \|\mathbf{x} - \mathrm{G}_i(\mathbf{z})\|^2$ as the squared Euclidean cost function in which $\mathrm{W}_{\mathcal{L}}(\cdot)$ is the squared 2-Wasserstein distance [8].

4 Theoretical framework

ELBO is an important indicator of the VAE's performance and is used as its main optimization function [11]. In the following, we provide a new perspective

for analyzing the forgetfulness behaviour of VAEs during the continuous learning of several batches of data by formulating the ELBO's variation as a learning and forgetting process. The code and Supplemental Materials (SM) are available at https://github.com/dtuzi123/OVAE.

4.1 Analysis of forgetting in a single model

Firstly, we derive an upper bound to ELBO of the target domain $\mathbb{P}_{\mathbf{x}}$, based on the dynamic OT problem (Eq. (5)).

Theorem 1. *For a VAE model h^i trained at t_i, where $p_\theta(\mathbf{x}\,|\,\mathbf{z}) = \mathcal{N}(\mathrm{G}_i(\mathbf{z}), \sigma^2\mathbf{I}_d)$ is the Gaussian decoder and $\sigma = 1/\sqrt{2}$, we have :*

$$\inf_{q_\omega(\mathbf{z})=p(\mathbf{z})} \mathbb{E}_{\mathbb{P}_{\mathbf{x}}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)] \leq -\frac{1}{2}\log\pi - \mathrm{W}^\star_{\mathcal{L}}(\mathbb{P}_{\mathbf{x}}, \mathbb{P}_{\mathrm{G}_i})\,, \tag{6}$$

The detailed proof is provided in Appendix-A from Supplemental Materials (SM). Based on the results from Theorem 1, we derive a bound that explains the forgetting process of VAEs.

Theorem 2. *Let $\mathbb{P}_{m_i}$ and $\mathbb{P}_{\mathbf{x}}$ be the source and target domains. From Eq.* (6)*, we derive the bound on the ELBO between $\mathbb{P}_{m_i}$ and $\mathbb{P}_{\mathbf{x}}$ at the training step t_i :*

$$\begin{aligned}\mathbb{E}_{\mathbb{P}_{\mathbf{x}}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)] \leq\ & \mathbb{E}_{\mathbb{P}_{m_i}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)] \\ & + 2\mathrm{W}^\star_{\mathcal{L}}(\mathbb{P}_{m_i}, \mathbb{P}_{\mathrm{G}_i}) - \mathrm{W}^\star_{\mathcal{L}}(\mathbb{P}_{\mathbf{x}}, \mathbb{P}_{m_i}) + \widetilde{\mathrm{F}}(\mathbb{P}_{\mathrm{G}_i}, \mathbb{P}_{m_i})\,,\end{aligned} \tag{7}$$

where $\widetilde{\mathrm{F}}(\mathbb{P}_{\mathrm{G}_i}, \mathbb{P}_{m_i})$ is expressed as :

$$\begin{aligned}\widetilde{\mathrm{F}}(\mathbb{P}_{\mathrm{G}_i}, \mathbb{P}_{m_i}) =\ & \mathbb{E}_{\mathbb{P}_{m_i}}[D_{KL}(q_\omega(\mathbf{z}\,|\,\mathbf{x})\,||\,p(\mathbf{z}))] \\ & + \left|\mathbb{E}_{\mathbb{P}_{m_i}}\mathbb{E}_{q_\omega(\mathbf{z}\,|\,\mathbf{x})}[-\mathcal{L}(\mathbf{x}, \mathrm{G}_i(\mathbf{z}))] - \mathrm{W}^\star_{\mathcal{L}}(\mathbb{P}_{m_i}, \mathbb{P}_{\mathrm{G}_i})\right|.\end{aligned} \tag{8}$$

Remark. The detailed proof is provided in Appendix-B from SM. We have several observations from Theorem 2 : 1) Improving the performance on the source domain (ELBO on $\mathbb{P}_{m_i}$) would not lead to increasing ELBO on the target domain $\mathbb{P}_{\mathbf{x}}$ because the right hand side (RHS) of Eq. (7) involves the negative term, $-\mathrm{W}^\star_{\mathcal{L}}(\mathbb{P}_{\mathbf{x}}, \mathbb{P}_{m_i})$. 2) Since RHS of Eq. (7) is upper bounded to ELBO on $\mathbb{P}_{\mathbf{x}}$, a large $\mathrm{W}^\star_{\mathcal{L}}(\mathbb{P}_{\mathbf{x}}, \mathbb{P}_{m_i})$ decreases RHS of Eq. (7) and therefore leads to the degenerated performance, measured by ELBO, on $\mathbb{P}_{\mathbf{x}}$, corresponding to forgetting the knowledge at the training step t_i. This is usually caused by the memory $\mathcal{M}_i$ that fails to capture all information of $\mathbb{P}_{\mathbf{x}}$ during the initial training process (when i is small) or after the training ($i = t_N$).

The effect of the memory diversity. In practice, $\mathbb{P}_{\mathbf{x}}$ is divided into several separate distributions (target domains) $\{\mathbb{P}_{\mathbf{x}^1}, \cdots, \mathbb{P}_{\mathbf{x}^n}\}$ where each $\mathbb{P}_{\mathbf{x}^j}$ is the characteristic distribution of a data category. Under this setting we analyze the forgetting behaviour in the class-incremental scenario.

Lemma 1. *Let $\{\mathbb{P}_{\mathbf{x}^1}, \cdots, \mathbb{P}_{\mathbf{x}^n}\}$ and $\mathbb{P}_{m_i}$ be the target domains and source domain, respectively. The bound on ELBO between the source and target domain*

is derived as :

$$\begin{aligned}\sum_{j=1}^{n} \mathbb{E}_{\mathbb{P}_{\mathbf{x}^j}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)] \leq \sum_{j=1}^{n} \Big\{ 2\mathrm{W}^{\star}_{\mathcal{L}}(\mathbb{P}_{m_i}, \mathbb{P}_{\mathrm{G}_i}) \\ + \mathbb{E}_{\mathbb{P}_{m_i}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)] - \mathrm{W}^{\star}_{\mathcal{L}}(\mathbb{P}_{\mathbf{x}^j}, \mathbb{P}_{m_i}) \Big\} + n\widetilde{\mathrm{F}}(\mathbb{P}_{\mathrm{G}_i}, \mathbb{P}_{m_i})\,.\end{aligned} \tag{9}$$

Proof. We sum up the bounds between $\mathbb{P}_{\mathbf{x}^j}$ and $\mathbb{P}_{m_i}$, where $j = 1, \cdots, n$ and prove Lemma 1.

Remark. We have several observations from Lemma 1 : 1) To maximize ELBO on target domains $\{\mathbb{P}_{\mathbf{x}^1}, \cdots, \mathbb{P}_{\mathbf{x}^n}\}$, $\mathrm{W}^{\star}_{\mathcal{L}}(\mathbb{P}_{\mathbf{x}^j}, \mathbb{P}_{m_i}), j = 1, \cdots, n$ must be minimized, corresponding to the diverse samples replayed from $\mathbb{P}_{m_i}$. 2) We also provide new insights into the backward transfer [39] by using Eq. (9). When a memory $\mathcal{M}_i$ prefers to store samples from a few recent data distributions $\{\mathbb{P}_{\mathbf{x}^{n-1}}, \mathbb{P}_{\mathbf{x}^n}\}$, the model would lead to negative backward transfer on past target domains $\{\mathbb{P}_{\mathbf{x}^1}, \cdots, \mathbb{P}_{\mathbf{x}^{n-2}}\}$. Data diversity in memory can relieve this negative effect.

4.2 Forgetting analysis of the expanding VAE mixture model

In this section, we extend the forgetting analysis from a single VAE model to the Dynamic Expansion Model (DEM).

Definition 4. *Let $\mathbf{H} = \{h_1, \cdots, h_k\}$ be a dynamic expansion model trained at t_i, which has built k components during the learning, where each h_i is a VAE model. Let $\mathbf{q} = \{q_1, \cdots, q_k\}$ represent the training steps that each component converged on. For instance, h_i converged on $\mathcal{M}_{q_i}$ at t_{q_i}, is not updated in the following training steps. Then $\mathbb{P}_{\mathrm{G}_{q_i}}$ and $\mathbb{P}_{m_{q_i}}$ represent the generator distribution and the distribution of samples drawn from $\mathcal{M}_{q_i}$.*

Lemma 2. *Let $\{\mathbb{P}_{\mathbf{x}^1}, \cdots, \mathbb{P}_{\mathbf{x}^n}\}$ be a set of n target domains. From Definition 4, the bound on the ELBO for the dynamic expansion model is derived as :*

$$\sum_{j=1}^{n} \mathbb{E}_{\mathbb{P}_{\mathbf{x}^j}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)] \leq \sum_{i=1}^{n} \mathrm{F}^{\star}(\mathbb{P}_{\mathbf{x}^i})\,, \tag{10}$$

where $\mathrm{F}^{\star}(\mathbb{P}_{\mathbf{x}^i})$ is the selection function, defined as :

$$\begin{aligned}\mathrm{F}^{\star}(\mathbb{P}_{\mathbf{x}^i}) = \max_{j=1,\cdots,k} \Big\{ \mathbb{E}_{\mathbb{P}_{m_{q_j}}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)] \\ + 2\mathrm{W}^{\star}_{\mathcal{L}}(\mathbb{P}_{m_{q_j}}, \mathbb{P}_{\mathrm{G}_{q_j}}) - \mathrm{W}^{\star}_{\mathcal{L}}(\mathbb{P}_{\mathbf{x}^i}, \mathbb{P}_{m_{q_j}}) + \widetilde{\mathrm{F}}(\mathbb{P}_{\mathrm{G}_{q_j}}, \mathbb{P}_{m_{q_j}}) \Big\}\,.\end{aligned} \tag{11}$$

The proof is provided in Appendix-C from SM. To compare with a single model (Lemma 1), DEM would provide a maximum upper bound to the Left Hand Side (LHS) of Eq. (10) due to the selection process, Eq. (11). Additionally, DEM can relieve the negative backward transfer by preserving prior knowledge into the frozen components.

4.3 Mixture expansion with the task information

Although the proposed theoretical framework is only used for TFCL, it can be extended for the case where task labels are known. We also apply the proposed theoretical framework for analyzing the forgetting behaviour of existing approaches (See details in Appendix-F from SM).

Definition 5. *(**Learning setting.**) Let $\mathcal{T} = \{\mathcal{T}_1, \cdots, \mathcal{T}_c\}$ represent a set of task labels where c is the number of tasks and we consider that each i-th task is associated with a testing dataset $\mathcal{D}_i^T$ and a training dataset $\mathcal{D}_i^S$. Let $\mathbb{P}_{\mathbf{x}^i}$ and $\mathbb{P}_{\widehat{\mathbf{x}}^i}$ represent the empirical distributions for $\mathcal{D}_i^S$ and $\mathcal{D}_i^T$, respectively. Since the task label is given, a mixture model starts to learn the first task and then either builds a new component or selects an existing component to learn a new task after the task switch. When a certain component is selected to learn a new task, the Generative Replay Mechanism (GRM) is used to relieve forgetting.*

Definition 6. *(**Generative replay.**) Let $\mathbb{P}^j_{\widetilde{\mathbf{x}}}$ represent the distribution of samples drawn from the generating process of h_j. Let $f_t \colon \mathcal{X} \to \mathcal{T}$ be the true labelling function that returns the task label for the data sample. If the i-th task is trained by h_j, let $\mathbb{P}_{\widetilde{\mathbf{x}}^{(i,m)}}$ be the distribution of samples drawn from the process $\mathbf{x} \sim \mathbb{P}^j_{\widetilde{\mathbf{x}}}$ if $f_t(\mathbf{x}) = i$, where m represents that $\mathbb{P}_{\widetilde{\mathbf{x}}^{(i,0)}}$ is evolved to $\mathbb{P}_{\widetilde{\mathbf{x}}^{(i,m)}}$ through m generative replay processes [58]. Let $\mathbb{P}_{\widetilde{\mathbf{x}}^{(i,0)}}$ and $\mathbb{P}_{\widetilde{\mathbf{x}}^{(i,-1)}}$ represent $\mathbb{P}_{\mathbf{x}^i}$ and $\mathbb{P}_{\widehat{\mathbf{x}}^i}$ for simplicity.*

Theorem 3. *Let $\mathcal{A} = \{a_1, \cdots, a_n\}$ be a set where each a_i represents the index of the component that has trained only once. Let $\widetilde{\mathcal{A}} = \{\widetilde{a}_1, \cdots, \widetilde{a}_n\}$ be a set of task labels where each $\widetilde{a}_i$ represents the index of the task learned by the a_i-th component. Let $\mathcal{B} = \{b_1, \cdots, b_{k-n}\}$ be a set where each b_i represents the index of the component that is trained more than once. Let $\widetilde{b}_i = \{\widetilde{b}_i^1, \cdots, \widetilde{b}_i^m\}$ be a set of task labels for the b_i-th component. Let c_i^j represent the number of generative replay processes for the $\widetilde{b}_i^j$-th task, achieved by the b_i-th component. Let $\mathbb{P}_{\mathrm{G}^i}$ represent the generator distribution of the i-th component. We derive the bound for a mixture model with k components trained on c tasks as :*

$$\sum_{i=1}^{|\mathcal{A}|}\Big\{\mathbb{E}_{\mathbb{P}_{\widehat{\mathbf{x}}^{\widetilde{a}_i}}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)]\Big\} + \sum_{i=1}^{|\mathcal{B}|}\Big\{\sum_{q=1}^{|\widetilde{b}_i|}\big\{\mathbb{E}_{\mathbb{P}_{\widehat{\mathbf{x}}^{\widetilde{b}_i^q}}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)]\big\}\Big\} \leq \mathcal{R}_S + \mathcal{R}_M \tag{12}$$

where $|\cdot|$ denotes the cardinal of a set. $\mathcal{R}_S$ is estimated by components that are trained only once, defined as :

$$\begin{aligned}\mathcal{R}_S = \sum\nolimits_{i=1}^{|\mathcal{A}|}\Big\{&2\mathrm{W}^\star_{\mathcal{L}}(\mathbb{P}_{\mathbf{x}^{\widetilde{a}_i}}, \mathbb{P}_{\mathrm{G}^{a_i}}) + +\widetilde{\mathrm{F}}(\mathbb{P}_{\mathrm{G}^{a_i}}\mathbb{P}_{\mathbf{x}^{\widetilde{a}_i}}) \\ &+ \mathbb{E}_{\mathbb{P}_{\mathbf{x}^{\widetilde{a}_i}}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)] - \mathrm{W}^\star_{\mathcal{L}}(\mathbb{P}_{\widehat{\mathbf{x}}^{\widetilde{a}_i}}, \mathbb{P}_{\mathbf{x}^{\widetilde{a}_i}})\Big\}.\end{aligned} \tag{13}$$

$\mathcal{R}_M$ is estimated by components that are trained on more than one task, as :

$$\begin{aligned}\mathcal{R}_M = \sum\nolimits_{i=1}^{|\mathcal{B}|}\Big\{\sum\nolimits_{q=1}^{|\widetilde{b}_i|}\Big\{&\mathbb{E}_{\mathbb{P}_{\widetilde{\mathbf{x}}^{(\widetilde{b}_i^q, c_i^q)}}}[\mathcal{L}_{ELBO}(\mathbf{x};\theta,\omega)] + \sum\nolimits_{s=0}^{c_i^q}\big\{2\mathrm{W}^\star_{\mathcal{L}}(\mathbb{P}_{\widetilde{\mathbf{x}}^{(\widetilde{b}_i^q,s)}}, \mathbb{P}_{\mathrm{G}^{b_i}}) \\ &+ \widetilde{\mathrm{F}}(\mathbb{P}_{\mathrm{G}^{b_i}}, \mathbb{P}_{\widetilde{\mathbf{x}}^{(\widetilde{b}_i^q,s)}}) - \mathrm{W}^\star_{\mathcal{L}}(\mathbb{P}_{\widehat{\mathbf{x}}^{(\widetilde{b}_i^q,s-1)}}, \mathbb{P}_{\widetilde{\mathbf{x}}^{(\widetilde{b}_i^q,s)}})\big\}\Big\}\Big\}.\end{aligned} \tag{14}$$

Remark. The detailed proof is provided in Appendix-D from SM. Theorem 3 has the following observations : 1) If the number of components k is equal to the number of tasks, then $\mathcal{R}_M = 0$ and there is no forgetting. When the number of

components decreases, forgetting happens because the last term in the RHS of Eq. (14) is increased, leading to a decrease in the RHS of Eq. (12) (corresponding to the decrease of ELBO on all target domains). 2) If $k = 1$, then $\mathcal{R}_S$ is about only the last task, then $\mathcal{R}_M$ is increased significantly since the accumulated errors $\sum_{s=0}^{c_i^q}\{\mathrm{W}_{\mathcal{L}}^{\star}(\mathbb{P}_{\widehat{\mathbf{x}}^{(\tilde{b}_i^q,s-1)}}, \mathbb{P}_{\widetilde{\mathbf{x}}^{(\tilde{b}_i^q,s)}})\}$ in Eq. (14) increases. Learning early tasks would lead to more forgetting than when learning the recent tasks for $k = 1$ because early tasks would have more accumulated errors (c_i^q in $\mathcal{R}_M$ is large as i increases (See Appendix-D from SM)).

5 Methodology

Previous approaches have proposed to learn a diverse memory according to the category information. However, these approaches do not provide a theoretical guarantee for the accumulated memory's diversity. To our best knowledge, this paper is the first to provide a theoretical forgetting analysis and guarantees for existing TFCL models (See details in Appendix-F of SM). Additionally, the proposed theoretical framework demonstrates that the diversity of memory content can be achieved without knowing the category information (Lemma 1). Based on the conclusion of Lemma 1, we introduce a new memory approach which consists of three modules: LTM, STM and the Learner. The proposed approach does not require any task information or supervised signals for unsupervised learning. Firstly, we introduce the proposed OCM with the Learner implemented as a single VAE, and then we extend this into a dynamic expansion mechanism.

5.1 Online Cooperative Memorization (OCM)

Notations. Let $\mathcal{M}_i^l = \{\mathbf{x}_{i,j}^l\}_{j=1}^{n_i^l}$ and $\mathcal{M}_i^e = \{\mathbf{x}_{i,u}^e\}_{u=1}^{n_i^e}$ represent the samples stored in the LTM and STM, respectively, at the training step t_i while n_i^l and n_i^e represent the number of samples. Let $\mathcal{M}_{Max}^e$ represent the maximum number of samples which can be stored in $\mathcal{M}_i^e$.

The training procedure, presented in Fig. 1, consists of three main stages, as described in the following.

Stage 1 : Learning. At the training step t_i, STM stores a new batch of samples $\mathbf{X}_i^b$ into $\mathcal{M}_i^e$, while the model, consisting of a single VAE, is trained to update both $\mathcal{M}_i^e$ and $\mathcal{M}_i^l$ using Eq. (1). Once the training is finished, we perform the next step.

Stage 2: Evaluation. We perform this step if and only if $n_i^e \geq \mathcal{M}_{Max}^e$ in order to reduce the computational cost. The main goal of this stage is to evaluate the correlation between stored samples from STM and LTM. Firstly, we treat each stored sample as a node and introduce a graph relationship matrix $\mathbf{S}_i \in \mathbb{R}^{n_i^e \times n_i^l}$, whose elements $\mathbf{S}_i(j,u)$ represent the correlation between two samples $\mathbf{x}_{i,j}^e$ and $\mathbf{x}_{i,u}^l$, from STM and LTM respectively. Directly evaluating each $\mathbf{S}_i(j,u)$ in the high-dimensional data space is intractable since it would require overloaded computations [17] and auxiliary training [7, 37]. Since the model has been trained on

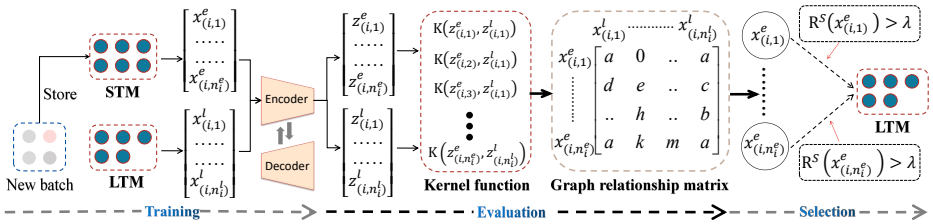

Fig. 1. The training of OCM consists of three stages : **(Learning.)** STM continually stores recent samples while the model is trained to adapt both LTM and STM; If STM is full, we perform the evaluation and selection stages, otherwise, we continually perform the learning stage. **(Evaluation.)** We obtain the feature vectors $\{\mathbf{z}^e_{(i,1)}, \cdots, \mathbf{z}^l_{(i,n^l_i)}\}$ from inputs $\{\mathbf{x}^e_{(i,1)}, \cdots, \mathbf{x}^l_{(i,n^l_i)}\}$ by using a VAE encoder, which is used for the evaluation of the sample similarity using the kernel from Eq. (15). This similarity information is preserved in the graph relationship matrix $\mathbf{S}_i$. **(Selection.)** We transfer the samples from STM to LTM using the proposed criterion Eq. (18) by means of $\mathbf{S}_i$ from (16).

(a) Real testing samples. (b) VAE-ELBO-Random. (c) VAE-ELBO-OCM.

Fig. 2. Image reconstruction compared to real images.

both past samples from LTM and the current samples from STM, it can be used as a discriminator. We then evaluate the distance between two samples based on the perceptual feature space of the learned model by using the Radial Basis Function (RBF) kernel :

$$\mathrm{K}(\mathbf{x}^e_{i,j}, \mathbf{x}^l_{i,u}) = \exp\left(-\frac{\left\|\mathbf{z}^e_{i,j} - \mathbf{z}^l_{i,u}\right\|^2}{2\alpha^2}\right), \tag{15}$$

where $\mathbf{z}^e_{i,j}$ and $\mathbf{z}^l_{i,u}$ are feature vectors extracted from $\mathbf{x}^e_{i,j}$ and $\mathbf{x}^l_{i,u}$ using the feature extractor implemented by the output layer of the encoder $q_\omega(\mathbf{z}\,|\,\mathbf{x})$ of the VAE model, as illustrated in Fig. 1. $\mathbf{S}_i(j,u) = \mathrm{K}(\mathbf{x}^e_{i,j}, \mathbf{x}^l_{i,u})$ and $\|\cdot\|^2$ is the squared Euclidean distance. α is the scale hyperparameter for the kernel and we set $\alpha = 10$ to ensure that the output of $\mathrm{K}(\cdot,\cdot)$ is within $[0, 1]$. Eq. (15) can be further accelerated by the matrix operation, expressed as :

$$\mathbf{S}_i = \mathrm{F}_{\exp}\Big(-(\mathbf{Z}^e_i(-\mathbf{Z}^l_i)^{\mathrm{T}}) \odot (\mathbf{Z}^e_i(-\mathbf{Z}^l_i)^{\mathrm{T}})/2\alpha^2\Big), \tag{16}$$

where $\mathbf{Z}^e_i \in \mathbb{R}^{n^e_i \times d_z}$ and $\mathbf{Z}^l_i \in \mathbb{R}^{n^l_i \times d_z}$ are the feature matrices corresponding to $\mathcal{M}^e_i$ and $\mathcal{M}^l_i$, where each row is a feature vector of dimension d_z. $(\cdot)^{\mathrm{T}}$ and $\odot$ are the transpose operation and Hadamard product, respectively. $\mathrm{F}_{exp}(\cdot)$ is the exponential function for each element in a matrix.

Stage 3: Sample selection. This stage also require satisfying $N^e_i \geq \mathcal{M}^e_{Max}$ to avoid excessive LTM growing. The main goal of this stage is to choose samples that are very different from those already stored in LTM. We achieve this by calculating the average similarity scores using kernels between each candidate

Table 1. The estimation of log-likelihood on all testing samples by using the IWVAE bound with 1000 importance samples.

	Split MNIST			**Split Fashion**			**Split MNIST-Fashion**		
Methods	Log	Memory	N	Log	Memory	N	Log	Memory	N
VAE-ELBO-Random	-150.79	3.0K	1	-280.54	3.0K	1	-247.46	3.0K	1
LIMix [64]	-146.23	2.0K	30	-262.52	2.0K	30	-238.63	2.0K	30
CNDPM [35]	-120.71	2.0K	30	-257.56	2.0K	30	-236.79	2.0K	30
VAE-ELBO-OCM	-132.07	1.6K	1	-250.74	1.6K	1	-215.62	2.0K	1
VAE-IWVAE50-OCM	-127.11	1.6K	1	-247.90	1.6K	1	-224.34	2.0K	1
Dynamic-ELBO-OCM	**-115.89**	1.1K	5	**-237.69**	1.3K	10	**-187.49**	1.4K	10

sample $\mathbf{x}^e_{i,j}$ and each sample from LTM using $\mathbf{S}_i$ from Eq. (16) :

$$\mathrm{R}^S(\mathbf{x}^e_{i,j}) = \frac{1}{n^l_i} \sum\nolimits_{k=1}^{n^l_i} \mathbf{S}_i(j,k)\,. \tag{17}$$

Eq. (17) refers to the distance between $\mathbf{x}^e_{i,j}$ and all samples contained in the LTM. In order to control the size of LTM, we introduce a threshold λ for the sample selection :

$$\mathrm{R}^S(\mathbf{x}^e_{i,j}) > \lambda \Rightarrow \mathcal{M}^l_i = \mathcal{M}^l_i \cup \mathbf{x}^e_{i,j}\,. \tag{18}$$

The choice for λ influences the diversity and memory size of LTM. Empirically, according to the ablation study in Appendix H.4 from SM, $\lambda \in [0.2, 0.5]$ can achieve the best performance resulting in a reasonable LTM size for most datasets. Once the selection is finished, $\mathcal{M}^e_i$ is cleared for storing novel samples during the next training step t_{i+1}.

Table 2. IS and FID scores under Split CIFAR10.

Methods	*IS*	*FID*	Memory	N
VAE-ELBO-Random	3.84	116.26	1.0K	1
CNDPM [35]	4.12	95.23	1.0K	30
LIMix [64]	3.02	156.46	1.0K	30
VAE-ELBO-OCM	4.13	98.76	0.5K	1
Dynamic-ELBO-OCM	**4.16**	**92.99**	0.4K	3

Table 3. The estimation of log-likelihood on "Cross domain"

Methods	Log	Memory	N
VAE-ELBO-Random	-239.71	3.0K	1
LIMix [64]	-226.63	2.0K	30
CNDPM [35]	-218.15	2.0K	30
VAE-ELBO-OCM	-201.31	2.0K	1
VAE-IWVAE50-OCM	-204.35	2.0K	1
Dynamic-ELBO-OCM	**-177.29**	1.5K	11

5.2 Combining OCM with expansion mechanism

According to Lemma 2 and Section 4.3, by dynamically expanding the model with new components would lead to better performance. Moreover, the extension mechanism reduces negative transfer when each component learns different underlying data distributions (see detailed analysis in Appendix-C of SM). This analysis inspires us to implement the extension mechanism from two aspects. First, we introduce an expansion criterion to detect the data distribution shift by comparing the loss value between the previously learned and newly seen samples, which ensures a suitable network architecture. Second, to encourage each component to learn different underlying data distributions, we clear STM and LTM when we dynamically add a new component to the mixture model.

The newly added component can be an independent VAE or one that shares its parameters with existing components. In the following, we describe the latter setting. Let $\mathrm{f}^e_{\omega_s}: \mathcal{X} \rightarrow \mathcal{Z}'$ and $\mathrm{f}^e_{\omega_i}: \mathcal{Z}' \rightarrow \mathcal{Z}$ be the shared module and the component-specific module for the encoding process, where i represents the component index and $\mathcal{Z}'$ is the feature space. Similar to the encoding process, we have two modules for the decoding process, $\mathrm{f}^d_{\theta_s}: \mathcal{Z} \rightarrow \mathcal{X}'$ and $\mathrm{f}^d_{\theta_i}: \mathcal{X}' \rightarrow \mathcal{X}$, where $\mathcal{X}'$ is the feature space. The encoding and decoding processes for the i-th component can be implemented by $q_{\theta_{s,i}}(\mathbf{z}\,|\,\mathbf{x}) = \mathrm{f}^e_{\omega_s} \odot \mathrm{f}^e_{\omega_i}(\mathbf{x})$ and $p_{\theta_{s,i}}(\mathbf{x}\,|\,\mathbf{z}) = \mathrm{f}^e_{\theta_s} \odot \mathrm{f}^e_{\theta_i}(\mathbf{z})$, respectively, where $\mathrm{f}^e_{\omega_s} \odot \mathrm{f}^e_{\omega_i}: \mathcal{X} \rightarrow \mathcal{Z}' \rightarrow \mathcal{Z}$ is the encoding process. The optimization for the i-th component corresponds to maximizing ELBO :

$$\mathcal{L}^i_{ELBO}(\mathbf{x};\theta,\omega) := \mathbb{E}_{q_{\omega_{s,i}}(\mathbf{z}\,|\,\mathbf{x})}\left[\log p_{\theta_{s,i}}(\mathbf{x}\,|\,\mathbf{z})\right] - KL\left[q_{\omega_{s,i}}(\mathbf{z}\,|\,\mathbf{x})\,||\,p(\mathbf{z})\right] \tag{19}$$

where $\mathbf{z} \sim q_{\omega_{s,i}}(\mathbf{z}\,|\,\mathbf{x})$ and the shared modules are only updated by using Eq. (19) for $i > 1$ in order to avoid forgetting.

Criterion for dynamic expansion. When a mixture model has multiple components, we evaluate the sample similarity from Eq. (15) by using an augmented feature extractor that concentrates features from each component. The training process for the new components from the dynamic expansion model is the same as the one described in Section 5.1 where we incorporate a criterion for the model expansion in **Step 3 : (Sample selection)** :

$$|\mathrm{R}_i - \mathrm{R}_{last}| > \lambda_2\,, \mathrm{R}_i = \frac{1}{N'}\sum\nolimits_{j=1}^{N'}\left\{\frac{1}{K}\sum\nolimits_{c=1}^{K}\{\mathcal{L}^c_{ELBO}(\mathbf{x}_j;\theta,\omega)\}\right\}, \tag{20}$$

where $\mathbf{x}_j$ is the j-th sample from the joint memory $\mathcal{M}^e_i \cup \mathcal{M}^l_i$. $N' = n^e_i + n^l_i$ and R_i is the loss evaluated on all memorized samples using the mixture model at the training step t_i. R_{last} is the most recent loss value. The pseudocode of the algorithm is provided in Appendix-H from SM.

6 Experiments

6.1 Experiment setting and datasets

Datasets. For the Log-likelihood evaluation, we have the following settings: 1) **Split MNIST/Fashion.** Split MNIST [33] into ten parts according to the category information and create a data stream by collecting these parts in a class-incremental way. This is also done for Fashion database; 2) **Split MNIST-Fashion.** Combine Split MNIST and Split Fashion into a data stream; 3) **Cross-Domain.** Combine Split MNIST-Fashion and unsorted samples from OMNIGLOT [31]. We adapt CIFAR10 [28] and Tiny-ImageNet [32] for the generative modelling task. Similar to Split MNIST, we divide CIFAR10 and Tiny-ImageNet into ten parts, namely Split CIFAR10 and Split Tiny-ImageNet, respectively. The details of dataset, hyperparameter and network architecture are provided in Appendix-H.1 of SM.

Evaluation Criteria. We use the Inception Score (IS) [48] and Fréchet Inception Distance (FID) [19] for the evaluation of reconstruction quality. For the density estimation task, we estimate the real sample log-likelihood by using IWVAE bound [9], as in Eq. (2), considering 5000 importance samples.

Baseline. We introduce several baselines used in experiments: 1) VAE-ELBO-OCM : We train a single VAE model with ELBO using the proposed OCM. 2) VAE-IWVAE50-OCM : We train a single VAE model with IWVAE using the proposed OCM where the number of importance samples is 50. 3) VAE-ELBO-Random : We train a single VAE model with a memory that randomly removes samples when it reaches the maximum memory size. 4) Dynamic-ELBO-OCM : We train a mixture model with ELBO using the proposed OCM. 5) CNDPM [35] : CNDPM uses Dirichlet process for the expansion of the mixture system; 6) LIMix [64] : We assign an episodic memory with a fixed buffer size for the LIMix model used for TFCL. The maximum number of components for various models is set to 30 to avoid memory overload.

Table 4. The classification accuracy of five indepdnent runs for various models on three datasets.

Methods	Split MNIST	Split CIFAR10	Split CIFAR100
finetune*	19.75 ± 0.05	18.55 ± 0.34	3.53 ± 0.04
GEM* [39]	93.25 ± 0.36	24.13 ± 2.46	11.12 ± 2.48
iCARL* [46]	83.95 ± 0.21	37.32 ± 2.66	10.80 ± 0.37
reservoir* [55]	92.16 ± 0.75	42.48 ± 3.04	19.57 ± 1.79
MIR* [4]	93.20 ± 0.36	42.80 ± 2.22	20.00 ± 0.57
GSS* [3]	92.47 ± 0.92	38.45 ± 1.41	13.10 ± 0.94
CoPE-CE* [13]	91.77 ± 0.87	39.73 ± 2.26	18.33 ± 1.52
CoPE* [13]	93.94 ± 0.20	48.92 ± 1.32	21.62 ± 0.69
CURL* [45]	92.59 ± 0.66	-	-
CNDPM* [35]	93.23 ± 0.09	45.21 ± 0.18	20.10 ± 0.12
Dynamic-OCM	**94.02 ± 0.23**	**49.16 ± 1.52**	**21.79 ± 0.68**

Table 5. IS and FID on ImageNet database.

Model	*IS*	*FID*
MVAE-Gau [61]	**6.84**	/
MVAE-Gau fixed [61]	6.30	/
MVAE-GS [61]	6.52	/
MSVI [30]	6.12	/
InfoVAE [70]	6.14	/
β-VAE [20]	5.05	/
VAE [25]	5.46	/
MAE [40]	5.87	/
VAE-ELBO-Random	3.15	145.36
VAE-ELBO-OCM	3.36	133.23

6.2 Log-likelihood evaluation

In this section, we implement each VAE model or component by using the Bernoulli decoder. All datasets are binarized according to the setting from [9]. The results for Split MNIST, Split Fashion, Split MNIST-Fashion and Cross-domain are provided in Tables 1 and 3, where "Memory" represents the number of samples N^l in LTM. The proposed OCM can improve the performance on the density estimation tasks even when using a small memory size compared to the random selection approach. Additionally, the expansion mechanism combined with the proposed OCM can further improve the performance with a reasonable memory use, especially when learning multiple datasets (Split MNIST-Fashion and Cross-Domain). We also find that the use of IWVAE bound (Eq. (2)) into the proposed OCM can also improve the performance on a single dataset. To compare with the expansion models, such as LIMix and CNDPM, a single model with OCM outperforms these models by using a few more stored samples such as 2.0K for LTM and 0.5K for STM vs 2.0K for LIMix and CNDPM, in Cross-Domain experiments. However, OCM with the expansion mechanism outperforms LIMix and CNDPM by using fewer mixture components.

6.3 Evaluation of the reconstruction quality

To evaluate the reconstruction quality, we use β-VAE loss [20] where $\beta = 0.01$ for all models in order to avoid the over-regularization issue [51]. We report the

IS and FID scores for the reconstruction quality in Table 2. We can observe that the proposed OCM with the expansion mechanism outperforms other baselines. The IS and FID for Tiny-ImageNet are reported in Appendix-H.3 from SM.

We also explore training a single VAE with OCM for learning ImageNet [29] under TFCL where the batch size is 64. The maximum size for STM and LTM is set to 512 and 2048, respectively, to avoid increasing the computational cost. We follow the settings from [61], as described in Appendix-H.3 from SM, after resizing all images to 64×64 pixels. The FID and IS results are provided in Table 5 and the results of all baselines (training on a single dataset) are cited from [61]. The visual results are shown in Fig. 2 where we can observe that the reconstruction of VAE-ELBO-Random is blurred when compared with VAE-ELBO-OCM. These results show that the proposed OCM outperforms the random selection approach in the large-scale dataset under TFCL.

6.4 Classification task

The proposed approach is mainly used in unsupervised learning. We also show that OCM can be used in classification tasks when we train a classifier with OCM on the labelled dataset. We adapt the setting and network architecture from [13] with a batch size of 10 and the memory size for Split MNIST, Split CIFAR10 and Split CIFAR100 is limited to 2K, 1K and 5K, respectively. We report the results in Table 4 where '*' means that the result is cited from [13]. The additional information about baselines and the proposed Dynamic-OCM is provided in Appendix-H.2 of SM. The number of required parameters is provided in Appendix-H.6 of SM. These results show that the proposed OCM outperforms the state-of-the-art methods in the classification task using fewer parameters.

6.5 Ablation study and theoretical results

A full ablation study is performed including testing the configuration for the threshold λ from Eq (18), STM memory size, batch size and λ_2 from Eq. (20). We also provide the empirical results for the theoretical analysis. These ablation results and their analysis are provided in Appendix-H.4 from SM.

7 Conclusion

We introduce a new theoretical framework for providing insights into the forgetting behaviour of deep models based on VAEs under TFCL. The theoretical analysis demonstrates that ensuring a diversity of data in the pre-training memory is crucial for relieving forgetting in continuous learning systems. Inspired by this result, we propose the Online Cooperative Memorization (OCM) that does not require any supervised signals and therefore can be used in an unsupervised fashion. The empirical results demonstrate the effectiveness of the proposed OCM method.

References

1. Abbasnejad, E., Dick, M., van der Hengel, A.: Infinite variational autoencoder for semi-supervised learning. In: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 5888–5897 (2017)
2. Achille, A., Eccles, T., Matthey, L., Burgess, C., Watters, N., Lerchner, A., Higgins, I.: Life-long disentangled representation learning with cross-domain latent homologies. In: Proc. Advances in Neural Inf. Proc. Systems (NeurIPS). pp. 9873–9883 (2018)
3. Aljundi, R., Lin, M., Goujaud, B., Bengio, Y.: Gradient based sample selection for online continual learning. In: Advances Neural Information Processing Systems (NeurIPS). vol. 33, pp. 11817–11826 (2019)
4. Aljundi, R., Belilovsky, E., Tuytelaars, T., Charlin, L., Caccia, M., Lin, M., Page-Caccia, L.: Online continual learning with maximal interfered retrieval. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 33, pp. 11872–11883 (2019)
5. Aljundi, R., Kelchtermans, K., Tuytelaars, T.: Task-free continual learning. In: Proc. of IEEE/CVF Conf. on Computer Vision and Pattern Recognition. pp. 11254–11263 (2019)
6. Bang, J., Kim, H., Yoo, Y., Ha, J.W., Choi, J.: Rainbow memory: Continual learning with a memory of diverse samples. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8218–8227 (2021)
7. Belghazi, M.I., Baratin, A., Rajeshwar, S., Ozair, S., Bengio, Y., Courville, A., Hjelm, D.: Mutual information neural estimation. In: Proc. Inter. Conference on Machine Learning (ICML), vol. PMLR 80. pp. 531–540 (2018)
8. Bousquet, O., Gelly, S., Tolstikhin, I., Simon-Gabriel, C.J., Schoelkopf, B.: From optimal transport to generative modeling: the VEGAN cookbook. arXiv preprint arXiv:1705.07642 (2017)
9. Burda, Y., Grosse, R., Salakhutdinov, R.: Importance weighted autoencoders. arXiv preprint arXiv:1509.00519 (2015)
10. Chaudhry, A., Rohrbach, M., Elhoseiny, M., Ajanthan, T., Dokania, P., Torr, P.H.S., Ranzato, M.: On tiny episodic memories in continual learning. arXiv preprint arXiv:1902.10486 (2019)
11. Chen, L., Dai, S., Pu, Y., Li, C., Su, Q., Carin, L.: Symmetric variational autoencoder and connections to adversarial learning. In: Proc. Int. Conf. on Artificial Intel. and Statistics (AISTATS) 2018, vol. PMLR 84. pp. 661–669 (2018)
12. Courty, N., Flamary, R., Tuia, D., Rakotomamonjy, A.: Optimal transport for domain adaptation. IEEE Trans. on Pattern Analysis and Machine Intelligence **39**(9), 1853–1865 (2016)
13. De Lange, M., Tuytelaars, T.: Continual prototype evolution: Learning online from non-stationary data streams. In: Proc. of the IEEE/CVF Int. Conference on Computer Vision (ICCV). pp. 8250–8259 (2021)
14. Egorov, E., Kuzina, A., Burnaev, E.: BooVAE: Boosting approach for continual learning of VAE. Advances in Neural Information Processing Systems (NeurIPS) **35**, 17889–17901 (2021)
15. Fang, P., Harandi, M., Petersson, L.: Kernel methods in hyperbolic spaces. In: Proc. of the IEEE/CVF Int. Conference on Computer Vision (ICCV). pp. 10665–10674 (2021)
16. Fatras, K., Séjourné, T., Flamary, R., Courty, N.: Unbalanced minibatch optimal transport; applications to domain adaptation. In: Int. Conf. on Machine Learning (ICML), vol. PMLR 139. pp. 3186–3197 (2021)

17. Goldberger, J., Gordon, S., Greenspan, H., et al.: An efficient image similarity measure based on approximations of kl-divergence between two gaussian mixtures. In: Proc. IEEE Int. Conf. on Computer Vision (ICCV). vol. 3, pp. 487–493 (2003)
18. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Proc. Advances in Neural Inf. Proc. Systems (NIPS). pp. 2672–2680 (2014)
19. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local Nash equilibrium. In: Proc. Advances in Neural Information Processing Systems (NIPS). pp. 6626–6637 (2017)
20. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: β-VAE: Learning basic visual concepts with a constrained variational framework. In: Proc. Int. Conf. on Learning Representations (ICLR) (2017)
21. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. In: Proc. NIPS Deep Learning Workshop, arXiv preprint arXiv:1503.02531 (2014)
22. Hua, Y., Zhao, Z., Li, R., Chen, X., Liu, Z., Zhang, H.: Deep learning with long short-term memory for time series prediction. IEEE Communications Magazine **57**(6), 114–119 (2019)
23. Jung, H., Ju, J., Jung, M., Kim, J.: Less-forgetting learning in deep neural networks. arXiv preprint arXiv:1607.00122 (2016)
24. Kantorovitch, L.: On the translocation of masses. Management science **5**(1), 1–4 (1958)
25. Kingma, D.P., Welling, M.: Auto-encoding variational Bayes. arXiv preprint arXiv:1312.6114 (2013)
26. 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., Hadsell, R.: Overcoming catastrophic forgetting in neural networks. Proc. of the National Academy of Sciences (PNAS) **114**(13), 3521–3526 (2017)
27. Knoblauch, J., Husain, H., Diethe, T.: Optimal continual learning has perfect memory and is NP-hard. In: Proc. International Conference on Machine Learning (ICML), vol PMLR 119. pp. 5327–5337 (2020)
28. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech. rep. (2009)
29. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Inf. Proc. Systems (NIPS). pp. 1097–1105 (2012)
30. Kurle, R., Günnemann, S., van der Smagt, P.: Multi-source neural variational inference. In: Proc. of AAAI Conf. on Artificial Intelligence. vol. 33, pp. 4114–4121 (2019)
31. Lake, B.M., Salakhutdinov, R., Tenenbaum, J.B.: Human-level concept learning through probabilistic program induction. Science **350**(6266), 1332–1338 (2015)
32. Le, Y., Yang, X.: Tiny imagenet visual recognition challenge. CS 231N **7**(7), 3 (2015)
33. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. of the IEEE **86**(11), 2278–2324 (1998)
34. Lee, S., Goldt, S., Saxe, A.: Continual learning in the teacher-student setup: Impact of task similarity. In: International Conference on Machine Learning (ICML), vol. PMLR 139. pp. 6109–6119 (2021)
35. Lee, S., Ha, J., Zhang, D., Kim, G.: A neural Dirichlet process mixture model for task-free continual learning. In: Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:2001.00689 (2020)

36. Li, Z., Hoiem, D.: Learning without forgetting. IEEE Trans. on Pattern Analysis and Machine Intelligence **40**(12), 2935–2947 (2017)
37. Liu, H., Gu, X., Samaras, D.: Wasserstein GAN with quadratic transport cost. In: Proc. of the IEEE/CVF Int. Conf. on Computer Vision (ICCV). pp. 4832–4841 (2019)
38. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: Advances in Neural Information Processing Systems. pp. 700–708 (2017)
39. Lopez-Paz, D., Ranzato, M.: Gradient episodic memory for continual learning. In: Advances in Neural Information Processing Systems. pp. 6467–6476 (2017)
40. Ma, X., Zhou, C., Hovy, E.: MAE: Mutual posterior-divergence regularization for variational autoencoders. In: Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1901.01498 (2019)
41. Nguyen, C.V., Li, Y., Bui, T.D., Turner, R.E.: Variational continual learning. In: Proc. of Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1710.10628 (2018)
42. Parisi, G.I., Kemker, R., Part, J.L., Kanan, C., Wermter, S.: Continual lifelong learning with neural networks: A review. Neural Networks **113**, 54–71 (2019)
43. Raghavan, K., Balaprakash, P.: Formalizing the generalization-forgetting trade-off in continual learning. Advances in Neural Information Processing Systems **34** (2021)
44. Ramapuram, J., Gregorova, M., Kalousis, A.: Lifelong generative modeling. In: Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1705.09847 (2017)
45. Rao, D., Visin, F., Rusu, A.A., Teh, Y.W., Pascanu, R., Hadsell, R.: Continual unsupervised representation learning. In: Advances Neural Inf. Processing Systems (NeurIPS). pp. 7645–7655 (2019)
46. Rebuffi, S.A., Kolesnikov, A., Sperl, G., Lampert, C.H.: iCaRL: Incremental classifier and representation learning. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 2001–2010 (2017)
47. Ren, B., Wang, H., Li, J., Gao, H.: Life-long learning based on dynamic combination model. Applied Soft Computing **56**, 398–404 (2017)
48. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training GANs. In: Proc. Advances in Neural Inf. Proc. Systems (NIPS). pp. 2234–2242 (2016)
49. Shin, H., Lee, J.K., Kim, J., Kim, J.: Continual learning with deep generative replay. In: Advances in Neural Inf. Proc. Systems (NIPS). pp. 2990–2999 (2017)
50. Sobolev, A., Vetrov, D.: Importance weighted hierarchical variational inference. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 33 (2019)
51. Takahashi, H., Iwata, T., Yamanaka, Y., Yamada, M., Yagi, S.: Variational autoencoder with implicit optimal priors. In: Proc. of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 5066–5073 (2019)
52. Tang, S., Chen, D., Zhu, J., Yu, S., Ouyang, W.: Layerwise optimization by gradient decomposition for continual learning. In: Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9634–9643 (2021)
53. Titsias, M.K., Schwarz, J., Matthews, A.G.d.G., Pascanu, R., Teh, Y.W.: Functional regularisation for continual learning with Gaussian processes. In: Proc. Int. Conf. on Learning Represenations (ICLR), arXiv preprint arXiv:1901.11356 (2019)
54. Tolstikhin, I., Bousquet, O., Gelly, S., Schoelkopf, B.: Wasserstein auto-encoders. In: Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1711.01558 (2018)

55. Vitter, J.S.: Random sampling with a reservoir. ACM Transactions on Mathematical Software (TOMS) **11**(1), 37–57 (1985)
56. Wang, S., Li, X., Sun, J., Xu, Z.: Training networks in null space of feature covariance for continual learning. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 184–193 (2021)
57. Ye, F., Bors, A.: Lifelong teacher-student network learning. IEEE Trans. on Pattern Analysis and Machine Intelligence (2021). https://doi.org/10.1109/TPAMI.2021.3092677
58. Ye, F., Bors, A.G.: Learning latent representations across multiple data domains using lifelong VAEGAN. In: Proc. European Conf. on Computer Vision (ECCV), vol. LNCS 12365. pp. 777–795 (2020)
59. Ye, F., Bors, A.G.: Lifelong learning of interpretable image representations. In: Proc. Int. Conf. on Image Processing Theory, Tools and Applications (IPTA). pp. 1–6 (2020)
60. Ye, F., Bors, A.G.: Mixtures of variational autoencoders. In: 2020 Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA). pp. 1–6 (2020)
61. Ye, F., Bors, A.G.: Deep mixture generative autoencoders. IEEE Transactions on Neural Networks and Learning Systems pp. 1–15 (2021). https://doi.org/10.1109/TNNLS.2021.3071401
62. Ye, F., Bors, A.G.: Infovaegan: Learning joint interpretable representations by information maximization and maximum likelihood. In: Proc. IEEE Int. Conf. on Image Processing (ICIP). pp. 749–753 (2021). https://doi.org/10.1109/ICIP42928.2021.9506169
63. Ye, F., Bors, A.G.: Learning joint latent representations based on information maximization. Information Sciences **567**, 216–236 (2021)
64. Ye, F., Bors, A.G.: Lifelong infinite mixture model based on knowledge-driven Dirichlet process. In: Proc. of the IEEE Int. Conf. on Computer Vision (ICCV) (2021)
65. Ye, F., Bors, A.G.: Lifelong mixture of variational autoencoders. IEEE Transactions on Neural Networks and Learning Systems pp. 1–14 (2021). https://doi.org/10.1109/TNNLS.2021.3096457
66. Ye, F., Bors, A.G.: Lifelong twin generative adversarial networks. In: Proc. IEEE Int. Conf. on Image Processing (ICIP). pp. 1289–1293 (2021)
67. Ye, F., Bors, A.G.: Learning an evolved mixture model for task-free continual learning (2022)
68. Ye, F., Bors, A.G.: Lifelong generative modelling using dynamic expansion graph model. In: AAAI on Artificial Intelligence. AAAI Press (2022)
69. Zhai, M., Chen, L., Tung, F., He, J., Nawhal, M., Mori, G.: Lifelong GAN: Continual learning for conditional image generation. In: Proc. of the IEEE/CVF Int. Conf. on Computer Vision (ICCV). pp. 2759–2768 (2019)
70. Zhao, S., Song, J., Ermon, S.: InfoVAE: Balancing learning and inference in variational autoencoders. In: Proc. AAAI Conf. on Artif. Intel. vol. 33, pp. 5885–5892 (2019)