Journal of Computer Science and Cybernetics, V.41, N.3 (2025), 211 – 224
DOI no. 10.15625/1813-9663/22071

ELASTIC: ENERGY-BASED LATENT SPACE ALIGNMENT WITH SECOND-ORDER INFLUENCES OF MEMORY SELECTION FOR CONTINUAL LEARNING

THANH HAI DANG

Faculty of Information Technology, VNU University of Engineering and Technology,
E3 Building, 144 Xuan Thuy Street, Cau Giay District, Ha Noi, Viet Nam

Abstract. Continual learning models encounter a significant challenge, namely catastrophic forgetting. After training on a new task they forget previously learned knowledge, causing a significant drop in models' performance on previous tasks. Replay-based methods, the most efficient approach for addressing catastrophic forgetting, select a small number of data (the coreset) from each learned task to store into an episodic memory buffer for rehearsal during subsequent stages. This selection significantly impacts the models' ability to perform well on new tasks while maintaining performance on previous tasks. In this paper, we propose a two-phase method to address catastrophic forgetting in continual learning. The first phase utilizes the second-order influence function to select an effective coreset from previously learned tasks. Even with this effective selection, there is still a problem that the hidden feature space is unexpectedly transformed across each task, causing the model to forget optimal hidden representations on previously learned tasks. To address this issue, the second phase employs an Energy-based Latent aligner for Incremental learning (ELI) to re-align the hidden feature representations of tasks towards the optimal region. Extensive experiments on three continual learning benchmark datasets, i.e. CIFAR-10, CIFAR-100, and Split miniImageNet, demonstrate that our proposed method outperforms several existing state-of-the-art continual learning models.

Keywords. Replay-based continual learning, influence functions, energy-based model.

1. INTRODUCTION

Continual learning, also known as lifelong learning, is a crucial sub-field of study within machine learning and artificial intelligence that mimics the human-being ability to learn seamlessly throughout life, accumulating and integrating new knowledge without forgetting past expertise. To this end, continual learning models are expected to continuously learn and adapt to new task's coming data without causing a loss of knowledge learned from previously tasks [1]. However, deep neural networks based continual learning models are susceptible to catastrophic forgetting phenomenon, first introduced by McCloskey and Cohen [2], in which learning new tasks can significantly deteriorate the model's performance on previously learned tasks. This catastrophic forgetting phenomenon occurs because the continual learning model's parameters are updated to accommodate the new coming task, thus becoming ineffective for earlier tasks.

Corresponding author.
E-mail addresses: hai.dang@vnu.edu.vn (T.H. Dang).

Numerous studies have proposed methods to mitigate catastrophic forgetting in continual learning scenarios. These methods can be broadly categorized into three types: (1) regularization methods [3] that minimize changes in network weights when learning for new task's coming data to retain priorly learned knowledge; (2) dynamic architecture methods [4] that involve adding or removing nodes or layers to adapt neural networks to new data; and (3) memory-based methods [5] that store a subset of data samples from previously learned tasks to be replayed during training on new tasks, preventing the model from forgetting previous knowledge. Sun et al., 2023 [6] introduced a novel continual learning approach called second-order influence regularization, which significantly mitigates the issue of catastrophic forgetting by leveraging data replay. However, when applying this method in continual learning scenarios, the latent feature space of the original model is susceptible to changes as new tasks are learned.

In this paper, we propose a two-phase method that combines memory-based rehearsal model (EBM) and latent space alignment to address catastrophic forgetting in continual learning scenarios. In the first phase, we employ a memory replay approach using the second-order influence function [7] to select an effective coreset of data points from previously learned tasks. This coreset is stored in an episodic memory and replayed during training on new tasks, enhancing the model's performance while learning new task's data. In the second phase, we employ an energy-based latent aligner (ELI) [8] that learns an energy manifold in the latent feature space of the model. ELI is used to re-align the hidden feature representations of previously learned tasks, preventing the model from forgetting optimal representations learned on prior tasks.

Our key contributions are as follows:

- We propose to enhance the second-order influence function based episodic memory selection [6] for rehearsal during continual learning by utilizing an energy-based latent aligner (ELI) [8], which learns an energy manifold in the latent feature space of the model. ELI is used to re-align the hidden feature representations of previously learned tasks, preventing the model from forgetting optimal representations learned on prior tasks due to the feature space transformations that occur during continual learning.
- We conduct extensive experiments on three continual learning benchmark datasets, CIFAR-10, CIFAR-100, and Split miniImageNet, demonstrating that our proposed method outperforms several existing state-of-the-art continual learning approaches in mitigating catastrophic forgetting and maintaining the model's performance across tasks.

The remainder of this paper is organized as follows: Section 2 provides an overview of related work in continual learning, influence functions, and energy-based models. Section 3 details our proposed method, including the second-order influence function for the coreset selection and the energy-based latent aligner for feature space alignment. Next, Section 4 presents the experimental setup, results, and analysis, demonstrating the effectiveness of our approach. Finally, Section 5 concludes the paper and discusses potential future research directions.

2. RELATED WORK

In this section, we provide an overview of relevant concepts and related work in class incremental continual learning, influence functions for memory selection, and energy-based models.

2.1. Class incremental learning

Class-incremental learning is a continual learning paradigm that aims to develop a cohesive classifier by assimilating data from various classes sequentially [7]. In CIL, the model learns a sequence of tasks $1, 2, ..., T$, where each task k contains a dataset used for training $D_k = \{(x_k^i, y_k^i)_{i=1}^{n_k}\}$, with n_k data samples. Here, $x_k^i \in \chi$ is an input data sample, and $y_k^i \in Y_k \subseteq \gamma(= \bigcup_{k=1}^{T} Y_k)$ is the corresponding label or class. All Y_k are distinct for each task k, meaning $Y_k \cap Y_{k'} = \emptyset$.

The goal of CIL is to learn a prediction function $f : \chi \rightarrow \mathcal{Y}$ that determines the label or class $y \in Y_k$ for an input data sample x. To this end, a single model is constructed for all tasks learned up to the current task. During inference, task identification is unnecessary, as the model has learned to differentiate between classes from all previous tasks. However, this characteristic poses a challenge in establishing decision boundaries between classes of new tasks and those of previous tasks, known as inter-task class separation. It arises when the model lacks access to the training set of previous tasks while learning new ones, causing a decrease in classification accuracy of the model.

2.2. Influence functions for memory selection

Influence functions [9] offer a cost-effective approximation for addressing the corset selection problem in continual learning by adjusting data sample weights, eliminating the need for costly leave-one-out retraining, which was required by previous exact solutions. A common approach to mitigating the issue of catastrophic forgetting in continual learning is to use a replay-based method, where a small set of previously trained data samples is stored into an episodic memory for further training in subsequent stages. However, selecting which data samples to store is a challenging problem that requires considering the interaction between consecutive selections.

Previous continual learning research has typically treated each selection process as an isolated event, focusing on optimizing performance within individual selections. However, this approach neglects the crucial fact that in continual learning scenarios, each previous selection directly impacts the input data for subsequent selections, thereby influencing future decisions. Failing to account for this interaction can lead to a gradual deterioration of coreset quality over prolonged selection procedures. Therefore, it is imperative to adopt a holistic perspective that addresses the interdependence of selections to maintain the integrity and effectiveness of the coreset throughout the continual learning process.

In 2023, Sun et al. [6] introduce an innovative method to regularize prior selections in continual learning by leveraging the expected second-order influence of data samples. This approach aims to maintain the model's performance on the known dataset $\mathcal{D}_{1:t}$ as it progresses through subsequent learning stages. The authors propose a straightforward yet effective solution involving the storage of a subset of data samples $\mathcal{M}_t$ in a buffer memory, serving as a compact representation of the previously encountered data $\mathcal{D}_{1:t}$. To ensure efficient memory utilization, the buffer size is limited to $|\mathcal{D}_t| \leq m$, where m is significantly smaller than n, the total number of samples in the training set. The key innovation in this method lies in the use of the second-order influence function to carefully select the most informative samples for inclusion in the coreset $\mathcal{D}_t$.

2.3. Energy-based models

Energy-based models (EBMs) [10] are probabilistic models that rely on constructing an energy function $E(x)$ that maps each point x in the input space to a scalar energy value. These energy values are then transformed into a probability density $p(x)$ through the Gibbs distribution, as shown in Equation (1)

$$p(y|x) = \frac{e^{-E(x,y)/T}}{\int_{y'} e^{-E(x,y')/T}} = \frac{e^{-E(x,y)/T}}{e^{-E(x)/T}}. \tag{1}$$

The temperature parameter T in the Gibbs distribution serves as a control for the smoothness of the probability distribution. Higher values of T result in a more uniform distribution, while lower values lead to a more concentrated distribution around the energy minima. The denominator $\int_{y'} e^{-E(x,y')/T}$ is known as the partition function, which plays a crucial role in normalizing the probability distribution. The concept of free energy $E(x)$ encapsulates the overall energy landscape of the model, providing a measure of the stability and likelihood of different configurations. By minimizing the free energy, EBMs can learn to assign lower energies to more probable configurations, effectively capturing the underlying patterns and dependencies in the data.

EBMs offer a flexible and intuitive approach to model complex distributions by learning an energy function that assigns low energies to desired configurations. The interplay between the energy function, partition function, and free energy enables EBMs to capture intricate relationships and dependencies in the data, making them a powerful tool for various machine learning tasks.

3. METHOD

3.1. Problem formulation

Consider a continual learning scenario where a model encounters a stream of tasks $\mathcal{T}_{1..n}$ with corresponding data sets $\mathcal{D}_{1..n}$ and at each time step t only current task data $\mathcal{D}_t$ is accessible. The goal is to strategically select representative data samples from $\mathcal{D}_{1..t}$ to store into a memory buffer $\mathcal{M}_t$ where $|\mathcal{M}_t| \leq k$ such that when replayed on $\mathcal{M}_t$ the model maintains its performance across all encountered tasks $\mathcal{T}_{1:t}$ in latter steps. Let $\mathcal{L}(x,\theta)$ be the loss of the model with parameters θ on training data sample x, the selection objective is formulated in Equation (2)

$$\begin{aligned} \min_{\mathcal{M}_t \subset \mathcal{M}_{t-1} \cup \mathcal{D}_t} \quad & \sum_{x \in \mathcal{M}_{t-1} \cup \mathcal{D}_t} \mathcal{L}(x, \hat{\theta}) \\ \text{s.t.} \quad & \hat{\theta} =_{\theta} \sum_{x \in \mathcal{M}_t} \mathcal{L}(x, \theta). \end{aligned} \tag{2}$$

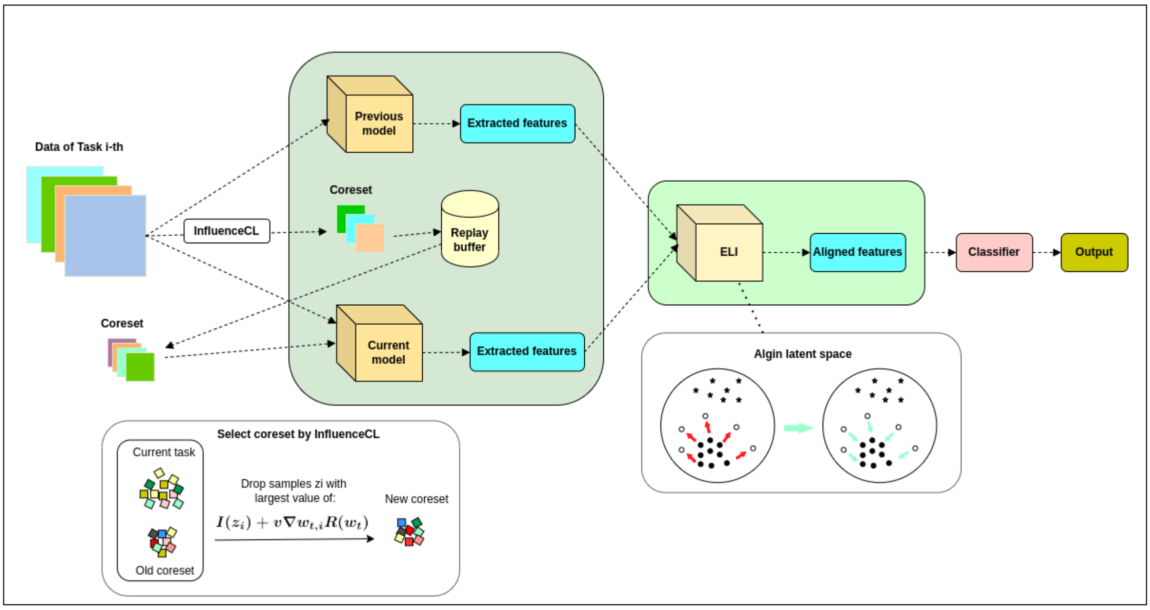

Figure 1: Overall architecture of our proposed model

3.2. Proposed method

Based upon recent advances in continual learning, our proposed model consists of two main phases, as depicted in Figure 1. The first phase involves continual learning based on the InfluenceCL memory replay method [6], which utilizes the second-order influence regularization approach to selectively replay important samples from previous tasks. This phase aims to maintain the model's performance on previously encountered tasks while incorporating new knowledge from incoming tasks. In the second phase, we utilize the Energy-based Latent Aligner (ELI) model [8] to tackle the changes in the latent feature space that occur during continual learning. The ELI model is designed to align the latent representations of the continually learned model with those of the original model, ensuring that the latent feature space remains consistent across tasks. By maintaining the stability of the latent space, the ELI model helps to preserve the previously acquired knowledge and prevents catastrophic forgetting.

3.2.1. InfluenceCL based memory replay

The InfluenceCL based memory replay method provides an efficient approximation for solving the good samples selection problem by perturbing sample weights, for which the previous exact solution requires expensive leave-one-out retraining. Following Sun et al., 2023 [6], we analyze both first-order and second-order influence of each candidate sample. This is quantified through parameter perturbation analysis using the model's loss gradient and Hessian information. By adding a small weight ϵ to each candidate sample z, the perturbed model parameters from the inner optimization in Equation (2) become

$$\hat{\theta}_{\epsilon,z} =_{\theta} \sum_{x \in \mathcal{M}_t} \mathcal{L}(x, \theta) + \epsilon \mathcal{L}(z, \theta). \tag{3}$$

The initial optimal parameters of the model at step t is $\hat{\theta}_{\epsilon,z}|_{\epsilon=0}$, which is denoted by $\hat{\theta}_t$.

As a result, the sample's first-order influence on the model's performance is given by [6]

$$\begin{aligned}\mathcal{I}(z) &= \sum_{x\in\mathcal{M}_{t-1}\cup\mathcal{D}_t} \frac{dL(x,\hat{\theta}_{\epsilon,z})}{d\epsilon}\Big|_{\epsilon=0} \\ &= -\sum_{x\in\mathcal{M}_{t-1}\cup\mathcal{D}_t} \nabla_\theta L(x,\hat{\theta}_t)^\top H_{\hat{\theta}_t}^{-1}\nabla_\theta L(x,\hat{\theta}_t)\end{aligned} \tag{4}$$

where $H_{\hat{\theta}_t} = \sum_{x\in\mathcal{M}_t}\nabla_\theta^2 L(x,\hat{\theta}_t)$ denotes the invertible Hessian matrix.

We could see that each selection at the prior step $t-1$ impacts on the influence of sample z at the current step t, thus determining the effectiveness of the current selection process. Taking into account this impact from a previous sample z', which is supposed to be up-weighted by ϵ, the influence of z turns into as follows

$$\mathcal{I}_{\epsilon,z'}(z) = -\left(\sum_{x\in\mathcal{M}_{t-1}\cup\mathcal{D}_t}\nabla_\theta L(x,\hat{\theta}_t) + \epsilon\nabla_\theta L(z',\hat{\theta}_t)\right)^\top (H_{\hat{\theta}_t} + \epsilon H_{\hat{\theta}_t,z'})^{-1}\nabla_\theta L(z,\hat{\theta}_t). \tag{5}$$

To this end, taking the derivative w.r.t. ϵ yields the second-order influence of z' and z, which is given by

$$\begin{aligned}\mathcal{I}^{(2)}(z',z) &= \frac{d\mathcal{I}_{\epsilon,z'}(z)}{d\epsilon}\Big|_{\epsilon=0} \\ &= -(\nabla_\theta L(z',\hat{\theta}_t) - H_{\hat{\theta}_t,z'}s_t)^\top H_{\hat{\theta}_t}^{-1}\nabla_\theta L(z,\hat{\theta}_t),\end{aligned} \tag{6}$$

where $s_t = H_{\hat{\theta}_t}^{-1}\sum_{x\in\mathcal{M}_{t-1}\cup\mathcal{D}_t}\nabla_\theta L(x,\hat{\theta}_t)$ is the inverse Hessian-vector product.

Second-order influences have two drawbacks, such as compromised sample diversity and amplified memory bias. To address this issue, we employ regularization techniques as in [6]. This regularization lessens the harmful effects of second-order influences without adding extra memory overhead and leverages gradient matching and diversity connections for improved results.

We note that the InfluenceCL based Memory Replay process suffers from the latent feature space instability. As new tasks are learned, latent representations drift from their optimal regions. To address this limitation, we integrate the InfluenceCL framework with an energy-based latent aligner [8]. This integration provides stronger protection against catastrophic forgetting by addressing both memory selection and feature space stability.

3.3. Energy-based latent aligner

In this proposed method, we perform energy-based modeling [8] in the latent space of the method. Let x denote an image sampled from the data distribution of the current task $\mathcal{T}_t$, i.e., $x \sim p_{\text{data}}^{\mathcal{T}_t}$. We obtain the latent representations of x from two different models $z^{\mathcal{T}_{t-1}} = F_\theta^{\mathcal{T}_{t-1}}(x)$, which represents the latent representation of x from the model trained till the previous task T_{t-1}, and $z^{\mathcal{T}_t} = F_\theta^{\mathcal{T}_t}(x)$, which represents the latent representation of x from the model trained till the current task T_t. Here, $F_\theta^{\mathcal{T}_{t-1}}$ and $F_\theta^{\mathcal{T}_t}$ denote the feature extraction functions of the models trained till tasks T_{t-1} and T_t, respectively, with parameters θ.

As illustrated in Figure 2, we first learn an energy manifold using three key components: images from the current task: $x \sim p_{\text{data}}^{\mathcal{T}_t}$, $z^{\mathcal{T}_{t-1}} = F_\theta^{\mathcal{T}_{t-1}}(x)$ and $z^{\mathcal{T}_t} = F_\theta^{\mathcal{T}_t}(x)$. We learn

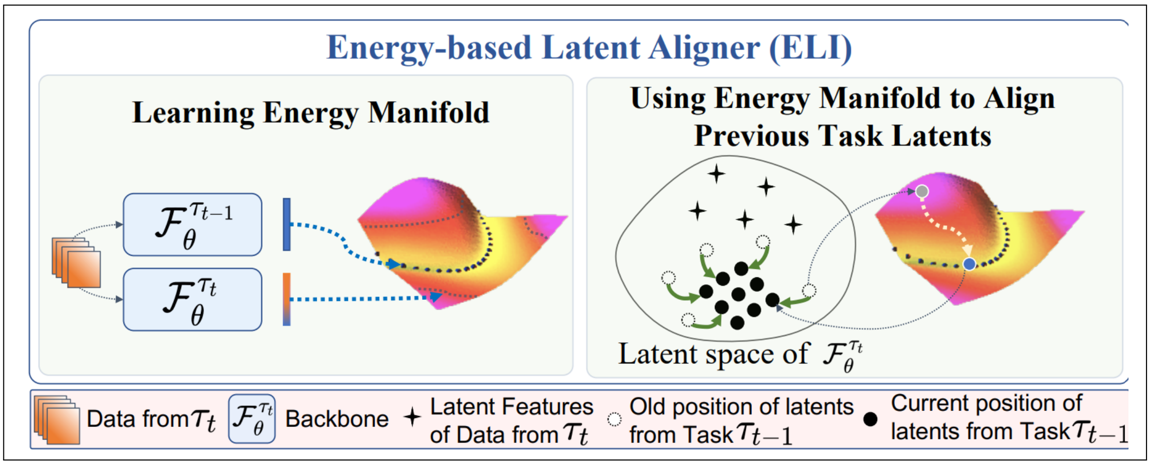
Energy-based Latent Aligner (ELI)
Learning Energy Manifold
$\mathcal{F}_\theta^{\mathcal{T}_{t-1}}$
$\mathcal{F}_\theta^{\mathcal{T}_t}$
Using Energy Manifold to Align Previous Task Latents
Latent space of $\mathcal{F}_\theta^{\mathcal{T}_t}$
Data from $\mathcal{T}_t$
$\mathcal{F}_\theta^{\mathcal{T}_t}$ Backbone
Latent Features of Data from $\mathcal{T}_t$
Old position of latents from Task $\mathcal{T}_{t-1}$
Current position of latents from Task $\mathcal{T}_{t-1}$

Figure 2: Energy-based Latent Aligner [8]

an energy-based model E_ψ in the latent space, where ψ represents the parameters of the energy model. The energy model learns to assign low energy values to latent representations $z^{\mathcal{T}_{t-1}}$ from the previous task's model. Conversely, it assigns high energy values to latent representations $z^{\mathcal{T}_t}$ from the current task's model. This energy-based modeling helps capture the desired characteristics of the latent space, where the representations from the previous task's model are considered more stable and preferred. Next, once the energy manifold E_ψ is learned, it is used to address the representational shift that occurs when instances from previous tasks are passed through the current model. Specifically, for an instance x belonging to a previous task $\mathcal{T}_{t-1}$, its latent representation $z^{\mathcal{T}_t} = F^\theta_{\mathcal{T}_t}(x)$ obtained from the current model $F^\theta_{\mathcal{T}_t}$ may have undergone a representational shift. As a result, $z^{\mathcal{T}_t}$ is likely to have higher energy values on the learned energy manifold E_ψ.

To mitigate this representational shift, we align the latent representations $z^{\mathcal{T}_t}$ to alternate locations in the latent space that minimize their energy on the manifold E_ψ. This alignment process involves finding the optimal latent representations $\hat{z}^{\mathcal{T}_t}$ that minimize the energy function

$$\hat{z}^{\mathcal{T}_t} = \arg\min_z E_\psi(z), \tag{7}$$

where $\hat{z}^{\mathcal{T}_t}$ represents the aligned latent representation of x from the current model.

The energy-based latent alignment step is performed in conjunction with the continual learning process, where the model is trained on the current task while leveraging the energy manifold to regularize the latent space. This integration allows the model to adapt to new tasks while maintaining the integrity of the latent representations learned from previous tasks.

3.3.1. Learning latent aligner

With a given latent feature vector $z \in \mathbb{Z}^{\mathrm{D}}$ in ELI [8], we learn an energy function $E_\psi(z) : \mathbb{R}^{\mathrm{D}} \to \mathbb{R}$ to map this to a scalar energy value. EBM is defined as a Gibbs distribution $p_\psi(\mathbf{z})$ over $E_\psi(z)$

$$p_\psi(z) = \frac{\exp(-E_\psi(z))}{\int_z \exp(-E_{\psi(z)})\, dz}.$$

The derivative of the above objective is as follows

$$\partial_\psi L(\psi) = \mathbb{E}_{z\sim p_{\text{true}}}[-\partial_\psi E_\psi(z)] + \mathbb{E}_{z\sim p_\psi}[\partial_\psi E_\psi(z)].$$

The term $\mathbb{E}_{z\sim p_{\text{true}}}[-\partial_\psi E_\psi(z)]$ shows that the energy for a sample z from the true data distribution p_{true} will be minimized, while the term $\mathbb{E}_{z\sim p_\psi}[\partial_\psi E_\psi(z)]$ ensures the higher energy for model's samples. In ELI, p_{true} is easy to sample from, as it represents the latent representations from the model trained on the previous tasks at any point in time. However, p_ψ is intractable to sample due to the normalization in the Gibbs distribution as in Equation (1). Therefore, we approximate samples with Langevin dynamics [11], which is a popular MCMC algorithm

$$z_{i+1} = z_i - \frac{\lambda}{2}\partial_z E_\psi(z) + \sqrt{\lambda}\omega_i, \quad \omega_i \sim \mathcal{N}(0, I),$$

where λ is the step size (or the learning rate) and ω captures data uncertainty.

4. EXPERIMENTS AND RESULTS

4.1. Experiment setup

Our proposed model utilizes the ResNet-18 architecture as the backbone for the original model. The model is optimized using the Stochastic Gradient Descent (SGD) method over 50 epochs per task, with a fixed batch size and replay batch size set to 32. The learning rate is set to 0.1 for both CIFAR-10 and CIFAR-100 datasets. During training, the model employs the cross-entropy loss function for replayed samples. The replay buffer is only updated in the last epoch of each task (i.e. epoch 50) to reduce computational complexity during model training.

The Energy-based Model (EBM) is implemented as a compact three-layer neural network, consisting of 64 neurons in the first two layers and a single neuron in the output layer. The EBM undergoes training for 1500 iterations, with mini-batches of size 128 and the learning rate of 0.0001 (following [8]). The sampling process from the EBM involves 30 Langevin iterations to ensure diverse and representative samples. The class-incremental learning method is used to evaluate the model's prediction results, requiring the model to differentiate between all previously trained classes at inference time without relying on task identification. To ensure the reliability and statistical significance of our findings, all experimental results are averaged over five independent runs.

4.2. Results

Tables 1 and 2 showcase the performance of our proposed model on the CIFAR-10 dataset with memory size settings of $m = 300$ and $m = 500$, respectively. The results consistently demonstrate that, irrespective of the memory setting, our model achieves substantial improvements in average performance across tasks when augmented with the Energy-based Latent Aligner (ELI). With $m = 300$, the average performance exhibits a notable improvement of 5.66%, while with $m = 500$, the improvement further increases to an impressive 7.43%. These results clearly highlight the effectiveness and robustness of our model on CIFAR-10.

To gain deeper insights into the performance of our model, we compare it with existing continual learning methods that utilize Influence Functions (IF) and those that do not, as summarized in Table 3 (data referenced from [6]). The results unequivocally demonstrate

Table 1: Average accuracy (%) of the SOTA InfluenceCL [6] (a) and our ELASTIC (b) models on the CIFAR-10 dataset with a memory size of $m = 300$.

a) InfluenceCL (SOTA)

Train Test on	T0	T1	T2	T3	T4
Task 0 (T0)	99.00	59.86	58.80	53.06	24.64
Task 1 (T1)		91.66	16.64	17.67	14.13
Task 2 (T2)			94.20	38.60	38.26
Task 3 (T3)				96.86	66.97
Task 4 (T4)					97.10
Average	99.00	75.76	56.55	51.55	48.22

b) InfluenceCL + ELI (Ours)

Train Test on	T0	T1	T2	T3	T4
Task 0 (T0)	99.00	93.88	56.31	57.93	47.24
Task 1 (T1)		91.66	85.01	10.55	15.36
Task 2 (T2)			94.20	89.93	24.96
Task 3 (T3)				96.86	84.73
Task 4 (T4)					97.10
Average	99.00	92.77	78.51	63.82	53.88

Table 2: Average accuracy (%) of the SOTA InfluenceCL [6] (a) and our ELASTIC (b) models on the CIFAR-10 dataset with a memory size of $m = 500$.

a) InfluenceCL (SOTA)

Train Test on	T0	T1	T2	T3	T4
Task 0 (T0)	98.99	66.96	62.35	64.13	24.84
Task 1 (T1)		91.70	21.60	18.67	22.94
Task 2 (T2)			94.52	37.43	43.66
Task 3 (T3)				97.51	67.50
Task 4 (T4)					97.12
Average	98.99	79.33	59.49	54.44	51.21

b) InfluenceCL + ELI (Ours)

Train Test on	T0	T1	T2	T3	T4
Task 0 (T0)	98.99	93.31	58.07	56.76	52.80
Task 1 (T1)		91.70	86.32	10.28	20.88
Task 2 (T2)			94.52	89.60	31.82
Task 3 (T3)				97.63	90.57
Task 4 (T4)					97.12
Average	98.99	92.51	79.64	63.57	58.64

the superiority of methods employing IF over those without IF, underscoring the positive impact of IF on data sample selection for subsequent continual learning stages. Notably, our method surpasses all these methods, showcasing the significant benefit of the ELI in synchronizing the latent feature space of the model after each task, thereby enabling more effective continual learning.

Table 3: Performance comparison in terms of accuracy of our proposed model with continual learning SOTA methods on three benchmark datasets, i.e. CIFAR-10, CIFAR-100 and Split miniImageNet.

Method type	Method	Cifar10 (%)		Cifar100 (%)		Split miniImageNet(%)	
		$m = 300$	$m = 500$	$m = 500$	$m = 1000$	$m = 500$	$m = 1000$
Non-IF	GEM [12]	37.51	36.95	15.91	22.79	-	-
	A-GEM [13]	20.02	20.01	9.31	9.27	10.69	10.69
	ER [14]	34.19	40.45	13.75	17.56	11.00	11.35
	GSS [15]	35.89	41.96	14.01	17.87	11.09	11.42
	ER-MIR [16]	38.53	42.65	13.49	17.56	11.07	11.32
	GDUMB [17]	36.92	44.27	11.11	15.75	6.22	7.15
	HAL [18]	24.45	27.94	8.20	19.59	-	-
	GMED [19]	38.12	43.68	14.56	18.67	11.03	11.73
IF	Vanilla IF [6]	41.76	47.14	17.49	22.75	12.08	14.64
	MetaSP [20]	43.76	50.10	19.28	25.72	12.74	14.54
	InfluenceCL [6]	48.22	51.21	21.15	27.36	13.28	16.68
	Ours	**53.88**	**58.64**	**30.61**	**33.95**	**15.01**	**17.53**

The continual learning model exhibits remarkably stable performance on the CIFAR-100 dataset, as evidenced by the results presented in Tables 4 and 5 with memory size settings of $m = 500$ and $m = 1000$, respectively. The average performance across ten tasks significantly improves by 9.68% and 6.59%, respectively, highlighting the model's ability to effectively

Table 4: Average accuracy (%) of the SOTA InfluenceCL (a) and our ELASTIC (b) models across each task on the CIFAR-100 dataset with a memory size of $m = 500$.

a) InfluenceCL (SOTA)

Train \ Test on	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9
Task 0 (T0)	87.12	46.30	35.54	29.22	23.76	18.96	20.36	17.54	13.22	16.44
Task 1 (T1)		82.11	38.74	20.87	21.80	21.62	20.66	20.36	13.10	13.10
Task 2 (T2)			86.28	37.96	20.50	21.34	19.42	14.42	13.06	14.40
Task 3 (T3)				85.54	32.71	17.68	18.12	15.10	10.42	8.30
Task 4 (T4)					86.28	32.76	19.78	15.26	13.70	12.76
Task 5 (T5)						88.36	40.54	26.00	17.96	11.42
Task 6 (T6)							87.62	18.80	14.66	9.04
Task 7 (T7)								86.36	26.18	9.40
Task 8 (T8)									90.04	23.04
Task 9 (T9)										91.42
Average	87.12	64.20	53.52	43.40	37.01	33.45	32.36	26.73	23.59	20.93

b) InfluenceCL + ELI (Ours)

Train \ Test on	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9
Task 0 (T0)	87.12	81.46	41.35	31.97	28.35	20.76	20.86	20.66	23.56	22.22
Task 1 (T1)		81.89	75.46	23.64	20.05	20.48	24.68	23.80	25.16	18.70
Task 2 (T2)			86.28	78.12	31.86	17.28	21.30	18.30	20.46	19.64
Task 3 (T3)				85.54	74.79	22.67	17.52	16.44	18.06	14.34
Task 4 (T4)					86.28	75.12	25.08	16.80	21.50	17.62
Task 5 (T5)						87.98	73.16	36.66	27.70	22.22
Task 6 (T6)							87.62	62.20	21.78	18.88
Task 7 (T7)								87.32	55.06	20.10
Task 8 (T8)									90.04	60.98
Task 9 (T9)										91.42
Average	87.12	81.67	67.70	54.82	48.26	40.72	38.60	35.27	33.70	30.61

retain and utilize knowledge from previous tasks. These results further reinforce our model's strong performance on CIFAR-100. When compared to methods using and not using IF (Table 3), our model achieves the highest performance, reaching an impressive 30.61% for $m = 500$ and 33.95% for $m = 1000$, demonstrating its superiority in continual learning scenarios.

The experimental results on the Split miniImageNet dataset demonstrate the superiority of our proposed ELASTIC model over the state-of-the-art InfluenceCL model in continual learning tasks. Tables 6 and 7 present the average accuracy of both models across each task with memory sizes of $m = 500$ and $m = 1000$, respectively. These results illustrate the evolution of average accuracy as the model progressively learns new tasks, offering a clear visualization of our model's superior performance and resilience on the Split miniImageNet dataset. The consistent improvement in accuracy across tasks demonstrates the efficacy and reliability of our approach in the continual learning setting.

From Table 6, it is evident that our ELASTIC model consistently outperforms the InfluenceCL model in terms of average accuracy across all tasks. With a memory size of $m = 500$, our model achieves an average accuracy of 15.57% on the final task (Task 4),

Table 5: Average accuracy (%) of the SOTA InfluenceCL (a) and our ELASTIC (b) models on the CIFAR-100 dataset with a memory size of $m = 1000$.

a) InfluenceCL (SOTA)

Test on / Train	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9
Task 0 (T0)	86.72	57.10	44.68	42.00	34.62	33.80	29.70	24.14	19.72	21.70
Task 1 (T1)		79.70	47.88	32.86	30.34	26.46	26.24	23.80	23.22	16.74
Task 2 (T2)			83.40	41.78	28.70	29.76	26.06	21.28	21.04	19.64
Task 3 (T3)				83.78	27.42	19.92	16.00	13.66	17.04	15.14
Task 4 (T4)					86.54	32.34	29.42	25.34	21.08	19.10
Task 5 (T5)						86.42	41.76	28.88	24.92	17.84
Task 6 (T6)							85.26	24.26	22.08	16.60
Task 7 (T7)								85.82	27.24	25.24
Task 8 (T8)									89.62	29.98
Task 9 (T9)										91.60
Average	86.72	68.40	58.65	50.11	41.52	38.12	36.35	30.90	29.55	27.36

b) InfluenceCL + ELI (Ours)

Test on / Train	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9
Task 0 (T0)	86.72	81.86	39.86	37.42	40.86	37.40	29.42	30.38	27.54	25.94
Task 1 (T1)		79.70	73.82	41.04	30.04	27.94	27.30	29.92	28.16	24.12
Task 2 (T2)			83.42	64.42	38.52	26.70	26.32	23.60	23.94	22.14
Task 3 (T3)				83.78	61.28	23.96	15.82	17.00	20.80	18.72
Task 4 (T4)					86.54	60.80	28.72	27.72	23.58	23.02
Task 5 (T5)						86.42	69.92	34.98	31.62	28.44
Task 6 (T6)							85.26	56.42	29.56	20.32
Task 7 (T7)								85.82	54.94	18.84
Task 8 (T8)									89.62	66.40
Task 9 (T9)										91.60
Average	86.72	80.78	65.70	56.67	51.45	43.87	40.39	38.23	36.64	33.95

Table 6: Average accuracy (%) of the SOTA InfluenceCL (left) and our ELASTIC (right) models cross each task on the Split miniImageNet dataset with a memory size of $m = 500$.

Test on / Train	T0	T1	T2	T3	T4
Task 0 (T0)	45.35	7.55	4.10	3.20	3.20
Task 1 (T1)		45.95	4.80	2.70	2.15
Task 2 (T2)			58.85	2.75	0.40
Task 3 (T3)				49.60	0.40
Task 4 (T4)					56.10
Average	45.35	26.75	22.58	14.56	13.45

Test on / Train	T0	T1	T2	T3	T4
Task 0 (T0)	45.35	22.30	10.75	6.70	6.20
Task 1 (T1)		45.95	21.40	5.70	4.40
Task 2 (T2)			58.85	11.70	2.50
Task 3 (T3)				49.60	3.65
Task 4 (T4)					56.10
Average	45.35	34.12	30.33	18.42	15.57

2.12% surpassing the SOTA InfluenceCL model's average accuracy of 13.45%. Furthermore, our model maintains higher average accuracy throughout the continual learning process, with improvements of 7.37%, 7.75%, and 3.86% on Tasks 1, 2, and 3, respectively, compared

Table 7: Average accuracy (%) of the SOTA InfluenceCL (left) and our ELASTIC (right) models on the Split miniImageNet dataset with a memory size of $m = 1000$.

Test on Train	T0	T1	T2	T3	T4
Task 0 (T0)	46.35	17.45	10.65	9.45	8.25
Task 1 (T1)		44.85	4.95	7.25	6.15
Task 2 (T2)			59.15	8.15	5.41
Task 3 (T3)				46.45	5.42
Task 4 (T4)					58.21
Average	46.35	31.15	24.92	17.83	16.68

Test on Train	T0	T1	T2	T3	T4
Task 0 (T0)	46.35	29.95	20.45	15.35	9.23
Task 1 (T1)		44.85	15.55	6.43	7.24
Task 2 (T2)			59.15	14.25	6.52
Task 3 (T3)				46.45	6.45
Task 4 (T4)					58.21
Average	46.35	37.40	31.72	20.62	17.53

to the InfluenceCL model. Similar trends can be observed in Table 7, where our ELASTIC model consistently outperforms the InfluenceCL model with a memory size of $m = 1000$. Our model achieves an average accuracy of 17.53% on Task 4, compared to 16.68% for the InfluenceCL model. Moreover, our model exhibits higher average accuracy across all tasks, with improvements of 6.25%, 6.80%, and 2.79% on Tasks 1, 2, and 3, respectively. Furthermore, the results demonstrate the scalability of our ELASTIC model. As evident from Tables 6 and 7, our model maintains its superior performance even when the memory size is increased from $m = 500$ to $m = 1000$. This indicates that our model can effectively utilize additional memory resources to enhance its continual learning capabilities.

The superior performance of our ELASTIC model can be attributed to the incorporation of the Energy-based Latent Aligner (ELI) component. The ELI component effectively aligns the latent feature space of the model after learning each new task, mitigating the forgetting of previous knowledge. By maintaining a more stable and consistent latent space, our model can better retain and utilize the information learned from earlier tasks, leading to improved accuracy and reduced catastrophic forgetting. It is worth noting that the performance gain of our ELASTIC model becomes more pronounced as the number of tasks increases. This highlights the effectiveness of our approach in handling the challenges of continual learning, particularly in scenarios with a larger number of sequential tasks.

5. CONCLUSION

Catastrophic forgetting poses a significant challenge in deep continual learning models. This issue arises because the model is trained on a sequence of tasks. When learning a new task, the parameter updates optimized for the new task render the model parameters ineffective for previous tasks. Another reason is that learning multiple tasks with different data distributions leads to changes in the hidden feature space over time, making these features ineffective for prediction. To address catastrophic forgetting, we employ two techniques to tackle these two causes. Firstly, we store a buffer of some data points from previous tasks to retrain when learning new tasks. We utilize the second-order influence function to select effective data points for this retraining. Secondly, we train an energy-based model to align the hidden feature space during model testing, enhancing model performance across tasks. Our comprehensive experiments on three continual learning benchmark datasets, i.e. CIFAR-10, CIFAR-100 and Split-miniImageNet datasets, provide compelling evidence for the effectiveness of our proposed model in continual learning tasks. The synergistic combination of the InfluenceCL memory replay method and the Energy-based Latent Aligner (ELI) model yields significant improvements in average performance across tasks, consistently outperforming state-of-the-art continual learning methods. These results underscore the critical importance of addressing changes in the latent feature space during continual learning to effectively mitigate catastrophic forgetting and achieve superior performance. Our model's ability to maintain stable and high performance across diverse datasets and memory settings highlights its robustness and potential for real-world applications.

REFERENCES

[1] Z. Chen and B. Liu, *Lifelong Machine Learning.* Morgan & Claypool Publishers, 2018.

[2] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," ser. Psychology of Learning and Motivation, G. H. Bower, Ed. Academic Press, 1989, vol. 24, pp. 109–165. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0079742108605368

[3] 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," 2016. [Online]. Available: https://arxiv.org/abs/1612.00796

[4] C. Fernando, D. Banarse, C. Blundell, Y. Zwols, D. Ha, A. A. Rusu, A. Pritzel, and D. Wierstra, "Pathnet: Evolution channels gradient descent in super neural networks," *arXiv preprint arXiv:1701.08734*, 2017.

[5] R. Aljundi, F. Babiloni, M. Elhoseiny, M. Rohrbach, and T. Tuytelaars, "Memory aware synapses: Learning what (not) to forget," 2017. [Online]. Available: https://arxiv.org/abs/1711.09601

[6] Z. Sun, Y. Mu, and G. Hua, "Regularizing second-order influences for continual learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[7] S. Mittal, S. Galesso, and T. Brox, "Essentials for class incremental learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3513–3522.

[8] K. Joseph, S. Khan, F. S. Khan, R. M. Anwer, and V. N. Balasubramanian, "Energy-based latent aligner for incremental learning," in *CVPR*, 2022, pp. 7452–7461.

[9] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," in *International Conference on Machine Learning.* PMLR, 2017, pp. 1885–1894.

[10] A. Gupta, S. Narayan, K. Joseph, S. Khan, F. S. Khan, and M. Shah, "Ow-detr: Open-world detection transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9235–9244.

[11] M. Welling and Y. W. Teh, "Bayesian learning via stochastic gradient langevin dynamics," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. Citeseer, 2011, pp. 681–688.

[12] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," in *NeurIPS*, 2017, pp. 6470–6479.

[13] A. Chaudhry, M. Ranzato, M. Rohrbach, and M. Elhoseiny, "Efficient lifelong learning with a-gem," *arXiv preprint arXiv:1812.00420*, 2018.

[14] A. Chaudhry, M. Rohrbach, M. Elhoseiny, T. Ajanthan, P. K. Dokania, P. H. Torr, and M. Ranzato, "On tiny episodic memories in continual learning," *arXiv preprint arXiv:1902.10486*, 2019.

[15] R. Aljundi, M. Lin, B. Goujaud, and Y. Bengio, "Gradient based sample selection for online continual learning," in *NeurIPS*, 2019, pp. 11816–11825.

[16] R. Aljundi, E. Belilovsky, T. Tuytelaars, L. Charlin, M. Caccia, M. Lin, and L. Page-Caccia, "Online continual learning with maximal interfered retrieval," in *NeurIPS*, 2019, pp. 11849–11860.

[17] A. Prabhu, P. H. Torr, and P. K. Dokania, "Gdumb: A simple approach that questions our progress in continual learning," in *ECCV*, 2020, pp. 524–540.

[18] A. Chaudhry, A. Gordo, P. Dokania, P. Torr, and D. Lopez-Paz, "Using hindsight to anchor past knowledge in continual learning," in *AAAI*, 2021, pp. 6993–7001.

[19] X. Jin, A. Sadhu, J. Du, and X. Ren, "Gradient-based editing of memory examples for online task-free continual learning," in *NeurIPS*, 2021, pp. 29193–29205.

[20] Q. Sun, F. Lyu, F. Shang, W. Feng, and L. Wan, "Exploring example influence in continual learning," in *NeurIPS*, 2022, pp. 27075–27086.

Received on December 11, 2024
Accepted on April 07, 2025