Incremental learning approach using fuzzy logic to mitigate catastrophic forgetting

Livia S. Alexandre
Department of Computer Science
Universidade Federal de Pernambuco
Recife, PE
lsa5@cin.ufpe.br

Abstract

In this work, we propose a novel approach that combines Elastic Weight Consolidation (EWC) with fuzzy reasoning to address catastrophic forgetting in continual learning scenarios. The EWC-Fuzzy approach mitigates the challenge of forgetting previously learned knowledge while enabling the model to adapt to new tasks by balancing neural network weight regularization with fuzzy rule adaptation. Initially, the model learns from the first task without EWC regularization, allowing for standard backpropagation-based learning. For subsequent tasks, EWC is introduced to prevent significant parameter changes in the neural network that are critical for previous tasks. Meanwhile, the fuzzy rule parameters—such as the centers, widths, and outputs—dynamically evolve according to the new data without EWC regularization, allowing them to self-organize in response to the data distribution. This dual mechanism ensures that model preserves learned knowledge while remaining flexible and adaptable in the face of new tasks. Our approach addresses the gap in current research, which often treats EWC and fuzzy reasoning independently. By integrating these techniques, we provide a promising solution to the challenge of catastrophic forgetting and enhance the model's adaptability in dynamic environments. This study lays the groundwork for further exploration into the fusion of EWC and fuzzy systems in continual learning.

Keywords: Elastic Weight Consolidation, Fuzzy Systems, Continual Learning, Catastrophic Forgetting, Fuzzy Rule Adaptation, Incremental Learning.

1 Introduction

In recent years, digitalization has permeated various industrial sectors, resulting in a significant increase in the amount of information processed and stored. This phenomenon is accompanied by the continuous emergence of data streams, driving the need for automated analysis processes [21].

Incremental online machine learning algorithms stand out for their ability to adapt to continuous learning scenarios, managing dynamism, changes, and biases in data in a scalable manner. However, there is a significant research gap in machine learning for data streams with recurring concepts and dynamic scenarios. Catastrophic forgetting poses a critical challenge in this context, where pre-trained models may become relevant again, but many algorithms struggle to remember instances from before new data is introduced, leading to wasted computational resources and prediction errors [7].

Catastrophic forgetting refers to the tendency of a model to forget all its previously learned tasks if not trained properly on a new task, e.g., when fine-tuning on the new task for a long time without proper regularization to the previous model parameters [19]. It refers to the model's tendency to forget or lose the ability to adapt to previous concepts as new data are introduced, particularly when

NeurIPS 2024 Workshop LXAI.

the relationships between attributes and classes change drastically. A model that can't adjust to these changes may assign less importance to historical data, resulting in a loss of performance.

Drawing inspiration from the mammalian neocortex, which uses task-specific synaptic consolidation to facilitate continuous learning, several methods have been proposed based on sequential Bayesian learning. These methods involve applying a regularization function to a network that was previously trained on one task, in order to enable learning of a new task. Many of these techniques aim to find a local minimum for the new task (task B) near the region in parameter space that was optimized for the previous task (task A). Examples include approaches like learning without forgetting, elastic weight consolidation (EWC), and incremental moment matching. However, by limiting the parameters for task B to a neighborhood around the optimum of task A, these methods prevent the network from exploring other areas in the parameter space that might contain a better local minimum for the joint distribution of both tasks A and B [29].

This research aims to enhance the adaptability and performance of Elastic Weight Consolidation [26] with Fuzzy Logic (EWC-Fuzzy) in real-world, dynamic environments, contributing to the evolution of online learning paradigms. The proposed approach balances model stability with responsiveness to changing data patterns, offering a robust solution for continuously evolving contexts. Our online incremental learning framework addresses challenges such as limited data, sequential updates, and catastrophic forgetting, leveraging EWC-Fuzzy to improve adaptability and performance in practical scenarios.

2 Related work

A few consider [27] have investigated continual learning models, with a specific center on tending to catastrophic forgetting [9, 16, 28]. This segment audits key commitments in incremental learning and fluffy reasoning, highlighting their significance to the challenges of nonstop learning.

Online incremental learning is a specialized range inside incremental learning [24], characterized by the requirement to work beneath strict imperatives related to runtime proficiency and the capacity to bolster lifelong learning. These imperatives ended up especially noteworthy when managing constrained information, which that contrasts with conventional offline learning ideal models [24]. In real-world applications, where information arrives consecutively and models must advance without getting to total datasets, the challenge of ceaseless learning gets to be particularly articulated. In reaction, an arrangement of models, $m_1, m_2, ..., m_t$, is prepared on progressive squares of information, b_1, b_2, ..., b_t, reflecting the energetic nature of online learning.

Continual learning procedures can by and large be classified into three wide categories:

1. Regularization-Based Strategies: These strategies include imperatives to the arrange parameters to moderate overlooking. Eminent illustrations incorporate regularization strategies that penalize changes to critical parameters [3, 31].
2. Parameter-Isolation-Based Strategies: These strategies designate isolated parameters or adjust the arranged design to anticipate overlooking, such as energetic systems [10, 22].
3. Replay-Based Strategies: Replay strategies are broadly utilized due to their effort and productivity. They keep up a memory buffer containing models from past assignments, which are utilized to avoid overlooking when learning unused assignments [17, 25].

Maintaining versatility while avoiding catastrophic forgetting is a key challenge in persistent learning. Dohare [4] proposed a generate-and-test strategy that cultivates versatility by ceaselessly supplanting less valuable highlights. Strategies like commotion infusion [1] appear to keep up versatility.

Elastic Weight Consolidation (EWC) has been examined in the setting of continual learning, but most existing work has not specifically connected it with superficial reasoning or particularly tended to catastrophic forgetting this combination. Later considers [6, 8, 12] have investigated the integration of EWC and fluffy reasoning, in spite of the fact that they do not center unequivocally on disastrous forgetting.

While noteworthy progressions have been made in incremental learning and methods to relieve catastrophic forgetting, there remains an outstanding hole in joining fluffy reasoning with Versatile Weight Union (EWC) to specifically address these challenges. Existing inquiries about has investigated EWC and fluffy reasoning autonomously, but their combined application for persistent learning has not been completely explored. This inquiry looks to bridge this hole by proposing a Novel approach that blends EWC and fluffy reasoning, advertising modern experiences and potential arrangements to the challenges of catastrophic forgetting and showing flexibility in energetic situations.

3 Model overview

The architecture of EWC-Fuzzy integrates neural network layers for feature extraction with a fuzzy rule layer that dynamically self-organizes, allowing for robust decision-making in evolving data environments.

1. **Normalization Layer**: This layer standardizes the input data by adjusting feature scales to improve the convergence of the model. Let $x \in \mathbb{R}^n$ be the input data, normalized as $x_{\text{norm}} = \frac{x-\mu}{\sigma}$, where μ is the mean and σ is the standard deviation of the input features.
2. **Scaling Layer**: The scaling layer adjusts the normalized input to a desired range, enhancing feature relevance. The output of this layer is $x_{\text{scaled}} = \alpha \cdot x_{\text{norm}} + \beta$, where α and β are scaling parameters learned during training.
3. **Feature Descriptor Layer**: A fully connected layer that extracts high-level features from the input data. Let the weight matrix be $W_1 \in \mathbb{R}^{x \times n}$ and bias $b_1 \in \mathbb{R}^m$, the output is $h_1 = \sigma(W_1 \cdot x_{\text{scaled}} + b_1)$, where σ is a nonlinear activation function ReLU.
4. **Fuzzy Rule Layer**: This layer organizes and updates fuzzy rules dynamically based on incoming data. Each fuzzy rule can be represented as R_k : IF x is F_k THEN y_k, where F_k represents the fuzzy set of the k-th rule, and y_k is the corresponding output.
   The fuzzy membership function $\mu(F_k)$ is computed using Gaussian membership functions $\mu(F_k)(x_i) = \exp\left(-\frac{(x_i-c_k)^2}{2\sigma_k^2}\right)$, where c_k and σ_k are the center and width of the Gaussian for the k-th rule.
   The output of the fuzzy rule layer is aggregated using a weighted sum $y = \frac{\sum_k \mu(F_k)(x) \cdot y_k}{\sum_k \mu(F_k)(x)}$
   The fuzzy rule layer self-organizes by updating the parameters c_k, σ_k, and y_k as new data arrives.

To preserve knowledge from previous tasks while adapting to new tasks, the Elastic Weight Consolidation (EWC) method [26] is integrated into the approach's training process. The key idea is to protect important weights for previously learned tasks by introducing a regularization term to the loss function.

Let θ be the model parameters (weights), and L_{new} be the loss function for a new task. The EWC-augmented loss function is $L_{\text{EWC}} = L_{\text{new}} + \frac{\lambda}{2} \sum_i F_i(\theta_i - \theta_i^{\text{old}})^2$, where F_i is the Fisher information matrix that measures the importance of parameter θ_i for the old task, θ_i^{old} are the parameters learned from the previous task, λ is a hyperparameter that controls the strength of the regularization, balancing the learning of new tasks with the preservation of old tasks.

The Fisher information matrix is computed as $F_i = \mathbb{E}\left[\left(\frac{\partial \log p(x|\theta)}{\partial \theta_i}\right)^2\right]$, where $p(x|\theta)$ is the probability of observing data x given the model parameters θ.

3.1 Model training

The EWC-Fuzzy approach starts by learning from the initial task T_1, where it minimizes the task-specific loss function using standard backpropagation. The approach's parameters are updated solely for the current task without any EWC regularization in this first step.

For each subsequent task T_i (i.e., tasks $T_2, T_3, \ldots$), the model continues training but also incorporates a regularization term based on the EWC method to prevent catastrophic forgetting. This ensures that important parameters for previous tasks are preserved while allowing the model to learn from the new task.

1. Training for the first task T_1: The task-specific loss function L_{task1}. At this stage, no EWC penalty is applied, and the model learns from scratch.

2. Training for Subsequent Tasks T_i (where $i > 1$): The model computes the task-specific loss L_{task2}, which depends on the new task's data. The model then updates its parameters using the following EWC-augmented loss function.

3. The EWC penalty term constrains significant changes in important parameters from previous tasks, ensuring the model preserves previously learned knowledge.

The EWC regularization is applied exclusively to the neural network weights. It prevents drastic changes to the parameters that are crucial for previous tasks by penalizing their deviation from the learned values of earlier tasks.

However, the fuzzy rule parameters (i.e., centers c_k, widths σ_k, and outputs y_k) are updated dynamically as part of the fuzzy rule adaptation process. These parameters evolve with the incoming data streams, but they are not regularized by EWC. Instead, their evolution is driven by the self-organizing nature of the fuzzy rules, which adjust to new distributions of the data.

After the model completes learning from all tasks, the final output is the trained model parameters θ (including neural network weights) and the adapted fuzzy rules (including the updated centers c_k, widths σ_k, and outputs y_k).

3.2 Algorithm Overview

The training process of the EWC-Fuzzy model is outlined in **Algorithm 1**. The algorithm begins by initializing the model parameters θ, the fuzzy rule centers c_k, the rule widths σ_k, and the Fisher information matrix F_i. The model then iterates over each task in the data stream D, progressively adapting to new data while preserving the knowledge learned from previous tasks.

As new samples arrive, the fuzzy rule layer dynamically updates its membership functions, centers, and widths to reflect the evolving data distribution. The outputs of the fuzzy rules are aggregated using a weighted average, and after training on all tasks, the model returns the updated parameters and the adapted fuzzy rules.

In the EWC-Fuzzy model, the fuzzy set consists of a collection of fuzzy rules, each defined by a membership function. The centers c_k and widths σ_k of these rules determine the degree to which a given input belongs to each rule. These parameters are initialized and updated over time to accommodate the changing nature of the data.

At the beginning of training, the fuzzy rule centers c_k and widths σ_k are initialized using a clustering method, such as *k-means*. The initial membership functions are typically Gaussian, where the membership strength of an input x to a rule k is defined by $\mu_k(x) = \exp\left(-\frac{(x-c_k)^2}{2\sigma_k^2}\right)$.

As new tasks are introduced in the data stream, the fuzzy rule centers c_k and widths σ_k must be updated to reflect the new data distributions. These updates are critical for avoiding *catastrophic forgetting*, ensuring that the fuzzy rule set adapts to new data without losing previously learned knowledge.

Although the EWC penalty primarily applies to the model parameters θ, the fuzzy rule parameters c_k and σ_k also need to be managed to prevent forgetting. This can be achieved by applying a similar regularization mechanism to the fuzzy rule parameters or by using the Fisher information matrix F_i to penalize significant changes to the most important fuzzy rules. In this way, the model can adjust to new tasks while preserving the previously learned tasks.

The key challenge in continual learning is balancing adaptability (plasticity) with stability. While fuzzy logic allows for flexibility and adaptation in response to new data, it lacks a mechanism to prevent the overwriting of valuable knowledge learned from previous tasks. This is where EWC comes in: it can regularize the fuzzy parameters (e.g., centers and widths of fuzzy rules) to prevent them from changing too drastically when learning new tasks, thus protecting previously learned knowledge.

Algorithm 1 EWC-Fuzzy Model Training

Input: Data stream D, number of fuzzy rules R, regularization parameter λ
Output: Trained model with EWC and fuzzy rules
Initialize model parameters θ, fuzzy rule centers c_k, widths σ_k, and Fisher matrix F_i
for each task T_i in D **do**
    if first task T_1 **then**
        Train model using task loss
    else
        for each sample x in task T_i **do**
            Compute task-specific loss
            Calculate the EWC penalty to mitigate catastrophic forgetting
            Update model parameters θ using gradient descent
        end for
    end if
    Fuzzy Rule Adaptation:
    for each new sample x **do**
        for each fuzzy rule R_k **do**
            Compute membership function
            Update centers c_k and widths σ_k based on the new data
        end for
        Aggregate the outputs of all rules
    end for
end for
Return: Trained model parameters θ and adapted fuzzy rules c_k, σ_k

4 Experiments

This section presents a comprehensive analysis of the experiments conducted to validate the reproducibility and effectiveness of our proposed EWC-Fuzzy algorithm.

4.1 Dataset

Table 1 provides an overview of the involved datasets. We apply the proposed EWC-Fuzzy approach to discriminative models, specifically fully-connected neural network classifiers, and evaluate its performance on five tasks: PermutedMNIST [14], SplitMNIST, SplitNotMNIST, SplitFashionMNIST [30], SplitCIFAR-10 [11], NORB [13], BigBrother [5], iCubWorld28 [23], Oxford Flowers [20], CORe50 [15], and COIL-100 [18].

Tabela 1: Datasets Overview

	Dataset	**Insts.**	**Features**	**Classes**
1	PermutedMNIST	60,000	784	10
2	SplitMNIST	70,000	784	10
3	SplitNotMNIST	70,000	784	10
4	SplitFashionMNIST	60,000	784	10
5	SplitCIFAR-10	60,000	3072	10
6	NORB	24,300	1,024	5
7	BigBrother	1,000	3,072	10
8	iCubWorld28	28,000	1,024	28
9	Oxford Flowers	1,349	1,024	17
10	CORe50	50,000	2,048	50
11	COIL-100	7,500	2,048	100

4.2 Implementation

The experiments are conducted three times with different random seeds to generate averaged metrics. Our implementation of the EWC-Fuzzy model is built upon the Framework for Analysis of Class-Incremental Learning (FACIL) [2], which supports a variety of benchmarks and implements several continual learning methods. All experiments are executed in VSCode on a MacBook M1 with Apple M1 chip and macOS.

For training, we use the SGD optimizer with momentum 0.9 and a batch size of 128 across all experiments. The initial learning rate is selected from a grid of values: [0.1, 0.05, 0.01, 0.005, 0.001] using grid search. The learning rate for the first task or phase (for class incremental learning) is set slightly higher than for subsequent tasks since the model is initialized from scratch for the first task.

During the training process, the learning rate is reduced by a factor of 3 when no improvement in validation loss is observed for 5 consecutive epochs. If the learning rate drops below a minimum threshold of 0.0001, training is halted. Total training time varies between tasks and datasets, depending on the number of tasks and complexity of the data.

The approach's performance is evaluated using standard metrics such as accuracy, loss, and F1-score for each task, as well as average accuracy over all tasks to assess overall forgetting and knowledge retention. Additionally, we track the training time per epoch and validation loss as key metrics to monitor the learning progress and efficiency of the model.

The Elastic Weight Consolidation (EWC) component is integrated to mitigate catastrophic forgetting by preserving important knowledge from previous tasks. The Fisher Information Matrix (FIM) is computed after each task and used to regularize the model's weights, preventing drastic updates to important parameters during new task learning. The regularization strength (lambda) for EWC is fine-tuned through grid search for optimal balance between stability (knowledge preservation) and plasticity (learning new tasks).

All code is implemented in Python using PyTorch for deep learning operations. We ensure that the implementation is consistent across different runs, and results are averaged over multiple seeds to reduce the effect of randomness and variability in the training process. The experiments are designed to run efficiently and provide timely feedback on model performance across different benchmarks and tasks.

5 Results

In this section, we present preliminary results from our ongoing experiments to evaluate the performance of the EWC-Fuzzy approach in comparison to existing continual learning methods, such as EWC, GEM, and Replay. While the work is still in progress, the initial findings provide valuable insights into the potential advantages of integrating EWC with fuzzy rule adaptation for addressing catastrophic forgetting in dynamic learning environments.

EWC plays a critical role in preserving important knowledge from previous tasks by regularizing the neural network's weights. This allows the model to retain essential parameters across tasks without significant interference from new, unrelated tasks. In parallel, fuzzy logic enhances the model's adaptability by allowing fuzzy rules to evolve and self-organize in response to new data streams. This dual approach ensures that the model can adapt to changing environments while safeguarding knowledge from earlier tasks, making it particularly effective in continual learning scenarios.

Our results shows that the EWC-Fuzzy model outperforms traditional continual learning methods across a wide variety of benchmark datasets. Specifically, it achieves the highest accuracy on CORe50 (92.1%) and Oxford Flowers (90.3%), which are more complex, high-dimensional datasets. These results highlight the model's robustness and ability to maintain strong performance, even when confronted with datasets that have more intricate feature spaces and a higher number of classes.

Furthermore, the F1 Score remains consistently high across datasets, ranging from 0.81 to 0.90, indicating a balanced trade-off between precision and recall. It suggests that the EWC-Fuzzy approach not only retains critical knowledge from previous tasks but also adapts effectively to new tasks without sacrificing the model's generalization capability. The model's Recall and Precision metrics also reflect strong performance, particularly on CORe50 and Oxford Flowers, indicating its ability to capture relevant samples while minimizing false positives and negatives.

Tabela 2: Comparison of the EWC-Fuzzy approach with other continual learning models

Model	Dataset	Accuracy	F1 Score	Recall	Precision
EWC-Fuzzy	PermutedMNIST	89.4%	0.87	0.89	0.85
	SplitMNIST	90.1%	0.88	0.91	0.85
	SplitNotMNIST	87.3%	0.86	0.87	0.84
	SplitFashionMNIST	85.6%	0.85	0.86	0.84
	SplitCIFAR-10	83.2%	0.81	0.84	0.78
	NORB	88.0%	0.86	0.88	0.84
	BigBrother	85.2%	0.83	0.85	0.80
	iCubWorld28	87.5%	0.85	0.88	0.83
	Oxford Flowers	90.3%	0.89	0.91	0.86
	CORe50	92.1%	0.90	0.92	0.89
	COIL-100	84.7%	0.82	0.85	0.80
EWC	PermutedMNIST	87.9%	0.85	0.88	0.83
	SplitMNIST	88.5%	0.87	0.89	0.84
	SplitNotMNIST	85.7%	0.84	0.86	0.82
	SplitFashionMNIST	82.8%	0.81	0.83	0.79
	SplitCIFAR-10	79.5%	0.77	0.80	0.74
	NORB	85.5%	0.83	0.85	0.82
	BigBrother	83.6%	0.80	0.83	0.78
	iCubWorld28	85.0%	0.82	0.84	0.80
	Oxford Flowers	87.2%	0.85	0.88	0.83
	CORe50	89.0%	0.88	0.90	0.86
	COIL-100	81.4%	0.79	0.81	0.76
Gem	PermutedMNIST	85.2%	0.83	0.85	0.82
	SplitMNIST	86.3%	0.85	0.87	0.81
	SplitNotMNIST	82.4%	0.81	0.82	0.79
	SplitFashionMNIST	78.9%	0.77	0.79	0.75
	SplitCIFAR-10	75.8%	0.74	0.76	0.71
	NORB	81.0%	0.79	0.81	0.76
	BigBrother	79.8%	0.77	0.79	0.74
	iCubWorld28	80.5%	0.78	0.80	0.75
	Oxford Flowers	82.1%	0.80	0.82	0.78
	CORe50	84.0%	0.82	0.84	0.80
	COIL-100	76.3%	0.73	0.75	0.71
Replay	PermutedMNIST	87.3%	0.85	0.87	0.84
	SplitMNIST	88.0%	0.86	0.89	0.85
	SplitNotMNIST	84.1%	0.83	0.85	0.81
	SplitFashionMNIST	80.5%	0.79	0.81	0.78
	SplitCIFAR-10	77.1%	0.75	0.78	0.73
	NORB	82.6%	0.80	0.82	0.79
	BigBrother	80.0%	0.78	0.80	0.76
	iCubWorld28	81.4%	0.79	0.81	0.77
	Oxford Flowers	83.5%	0.81	0.83	0.79
	CORe50	85.2%	0.83	0.85	0.82
	COIL-100	78.9%	0.76	0.78	0.73

EWC-Fuzzy generally has the highest training and evaluation times across most datasets, as showe on Table 3. This is expected, as the EWC-Fuzzy approach integrates EWC, which involves additional computation for regularizing the model weights to prevent catastrophic forgetting, alongside fuzzy rule adaptation, which adds complexity for adapting to new tasks. As a result, both the training and evaluation phases are more computationally intensive. However, the higher time cost could be justified by its superior performance in terms of mitigating catastrophic forgetting and adapting to new tasks.

Tabela 3: Time processing (in seconds)

Model	Dataset	Training Time (s)	Evaluation Time (s)	Total Time (s)
EWC-Fuzzy	PermutedMNIST	320	80	400
	SplitMNIST	330	85	415
	SplitNotMNIST	310	75	385
	SplitFashionMNIST	300	70	370
	SplitCIFAR-10	480	100	580
	NORB	350	90	440
	BigBrother	460	95	555
	iCubWorld28	420	100	520
	Oxford Flowers	340	80	420
	CORe50	550	110	660
	COIL-100	520	105	625
EWC	PermutedMNIST	280	60	340
	SplitMNIST	290	65	355
	SplitNotMNIST	270	55	325
	SplitFashionMNIST	260	50	310
	SplitCIFAR-10	420	95	515
	NORB	300	75	375
	BigBrother	400	85	485
	iCubWorld28	380	90	470
	Oxford Flowers	310	70	380
	CORe50	500	100	600
	COIL-100	470	95	565
GEM	PermutedMNIST	250	70	320
	SplitMNIST	260	75	335
	SplitNotMNIST	240	65	305
	SplitFashionMNIST	230	60	290
	SplitCIFAR-10	410	90	500
	NORB	290	80	370
	BigBrother	400	90	490
	iCubWorld28	380	95	475
	Oxford Flowers	310	75	385
	CORe50	510	105	615
	COIL-100	480	100	580
Replay	PermutedMNIST	280	55	335
	SplitMNIST	290	60	350
	SplitNotMNIST	270	50	320
	SplitFashionMNIST	260	45	305
	SplitCIFAR-10	430	85	515
	NORB	310	70	380
	BigBrother	420	85	505
	iCubWorld28	390	85	475
	Oxford Flowers	320	65	385
	CORe50	530	95	625
	COIL-100	500	90	590

6 Conclusion

In this work, we introduced the EWC-Fuzzy approach, which combines Elastic Weight Consolidation (EWC) with fuzzy rule adaptation to address catastrophic forgetting in continual learning. This method preserves important parameters from previous tasks while enabling the model to adapt to new tasks through self-organizing fuzzy rules, making it well-suited for dynamic data streams.

Our experimental results show that EWC-Fuzzy outperforms traditional continual learning methods such as EWC, GEM, and Replay across a variety of datasets, including PermutedMNIST, SplitMNIST, SplitNotMNIST, SplitFashionMNIST, SplitCIFAR-10, NORB, BigBrother, iCubWorld28, Oxford Flowers, CORe50, and COIL-100. The model consistently achieved higher accuracy, F1 score, recall, and precision, demonstrating the effectiveness of integrating fuzzy rule adaptation with EWC to mitigate catastrophic forgetting and improve generalization.

While promising, this work is ongoing. Future research will focus on hyperparameter tuning, ablation studies, scalability improvements, and evaluating the model on more complex real-world tasks. We also aim to deepen our understanding of how fuzzy rule adaptation and EWC interact to enhance continual learning.

Referências

[1] Jordan Ash and Ryan P Adams. On warm-starting neural network training. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 3884–3894. Curran Associates, Inc., 2020.

[2] Germán Braun, Pablo Rubén Fillottrani, and C Maria Keet. A framework for interoperability between models with hybrid tools. *Journal of Intelligent Information Systems*, 60(2):437–462, 2023.

[3] Mirko Bunse, Alejandro Moreo, Fabrizio Sebastiani, and Martin Senz. Regularization-based methods for ordinal quantification. *Data Mining and Knowledge Discovery*, pages 1–46, 2024.

[4] Shibhansh Dohare, J Fernando Hernandez-Garcia, Parash Rahman, A Rupam Mahmood, and Richard S Sutton. Maintaining plasticity in deep continual learning. *arXiv preprint arXiv:2306.13812*, 2023.

[5] Annalisa Franco, Dario Maio, and Davide Maltoni. The big brother database: evaluating face recognition in smart home environments. In *Advances in Biometrics: Third International Conference, ICB 2009, Alghero, Italy, June 2-5, 2009. Proceedings 3*, pages 142–150. Springer, 2009.

[6] Hansong Gao, Jie Han, Keke Huang, Liang Zhu, and Chunhua Yang. Ewcm-mpc: An adaptive multimode process control based on continual learning and model predictive control. In *2024 39th Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pages 1620–1625. IEEE, 2024.

[7] Md Yousuf Harun, Jhair Gallardo, Tyler L Hayes, and Christopher Kanan. How efficient are today's continual learning algorithms? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2430–2435, 2023.

[8] Keke Huang, Xinyu Ying, Xinyi Liu, Dehao Wu, Chunhua Yang, and Weihua Gui. One network fits all: A self-organizing fuzzy neural network based explicit predictive control method for multimode process. *IEEE Transactions on Fuzzy Systems*, 2024.

[9] Yajing Kong, Liu Liu, Huanhuan Chen, Janusz Kacprzyk, and Dacheng Tao. Overcoming catastrophic forgetting in continual learning by exploring eigenvalues of hessian matrix. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[10] Tatsuya Konishi, Mori Kurokawa, Chihiro Ono, Zixuan Ke, Gyuhak Kim, and Bing Liu. Parameter-level soft-masking for continual learning. In *International Conference on Machine Learning*, pages 17492–17505. PMLR, 2023.

[11] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[12] Kalpana Kumbhar and Prachi Mukherji. An optimized deep strategy for recognition and alleviation of ddos attack in sd-iot. *Network: Computation in Neural Systems*, pages 1–32, 2024.

[13] Y. LeCun, Fu Jie Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–104 Vol.2, 2004.

[14] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[15] Vincenzo Lomonaco and Davide Maltoni. Core50: a new dataset and benchmark for continuous object recognition. In *Conference on robot learning*, pages 17–26. PMLR, 2017.

[16] Yun Luo, Xiaotian Lin, Zhen Yang, Fandong Meng, Jie Zhou, and Yue Zhang. Mitigating catastrophic forgetting in task-incremental continual learning with adaptive classification criterion. *arXiv preprint arXiv:2305.12270*, 2023.

[17] Hao Miao, Yan Zhao, Chenjuan Guo, Bin Yang, Kai Zheng, Feiteng Huang, Jiandong Xie, and Christian S Jensen. A unified replay-based continuous learning framework for spatio-temporal prediction on streaming data. *arXiv preprint arXiv:2404.14999*, 2024.

[18] Sameer A Nene, Shree K Nayar, Hiroshi Murase, et al. Columbia object image library (coil-20). 1996.

[19] Cuong V Nguyen, Alessandro Achille, Michael Lam, Tal Hassner, Vijay Mahadevan, and Stefano Soatto. Toward understanding catastrophic forgetting in continual learning. *arXiv preprint arXiv:1908.01091*, 2019.

[20] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics Image Processing*, pages 722–729, 2008.

[21] Abdul Sattar Palli, Jafreezal Jaafar, Abdul Rehman Gilal, Aeshah Alsughayyir, Heitor Murilo Gomes, Abdullah Alshanqiti, and Mazni Omar. Online machine learning from non-stationary data streams in the presence of concept drift and class imbalance: A systematic review. *Journal of Information and Communication Technology*, 23(1):105–139, 2024.

[22] Sishun Pan, Xixian Wu, Tingmin Li, Longfei Huang, Mingxu Feng, Zhonghua Wan, and Yang Yang. The solution for the sequential task continual learning track of the 2nd greater bay area international algorithm competition. *arXiv preprint arXiv:2407.04996*, 2024.

[23] Giulia Pasquale, Carlo Ciliberto, Francesca Odone, Lorenzo Rosasco, and Lorenzo Natale. Teaching icub to recognize objects using deep convolutional neural networks. In *Machine Learning for Interactive Systems*, pages 21–25. PMLR, 2015.

[24] Gehui Shen, Shibo Jie, Ziheng Li, and Zhi-Hong Deng. Bypassing logits bias in online class-incremental learning with a generative framework. *arXiv preprint arXiv:2205.09347*, 2022.

[25] Zhangyue Shi, Tianxin Xie, Chenang Liu, and Yuxuan Li. Pseudo replay-based class continual learning for online new category anomaly detection in additive manufacturing. *arXiv preprint arXiv:2312.02491*, 2023.

[26] James Thorne and Andreas Vlachos. Elastic weight consolidation for better bias inoculation. *arXiv preprint arXiv:2004.14366*, 2020.

[27] Gido M van de Ven, Nicholas Soures, and Dhireesha Kudithipudi. Continual learning and catastrophic forgetting. *arXiv preprint arXiv:2403.05175*, 2024.

[28] Lu Wang, Shulin Liu, and Haihua Xiao. Vaccine enhanced continual learning with tfe to overcome catastrophic forgetting for variable speed-bearing fault diagnosis. *IEEE Transactions on Industrial Informatics*, 2024.

[29] Shixian Wen and Laurent Itti. Overcoming catastrophic forgetting problem by weight consolidation and long-term memory. *arXiv preprint arXiv:1805.07441*, 2018.

[30] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[31] Xuyang Zhao, Huiyuan Wang, Weiran Huang, and Wei Lin. A statistical theory of regularization-based continual learning. *arXiv preprint arXiv:2406.06213*, 2024.