Vol.16 No.01(2025),63-79
DOI: https://doi.org/10.61841/turcomat.v16i1.15029

A COMPREHENSIVE SURVEY OF MEMORY UPDATE MECHANISMS FOR CONTINUAL LEARNING ON TEXT DATASETS

J. Ranjith*[1] and Dr. Santhi Baskaran[2]

[1]Research Scholar, Department of Computer Science and Engineering, Puducherry Technological University, Puducherry, India
[2] Professor, Department of Information Technology, Puducherry Technological University, Puducherry, India
[1*]ranjithsathiya07@ptuniv.edu.in, [2]santhibaskaran@ptuniv.edu.in

Abstract. Over the last several years, there has been a growing focus on the CL field in the context of machine learning and its goal to create models capable of learning new tasks step by step without loss of prior knowledge. Among these, catastrophic forgetting is especially challenging in real-world settings where the data experience changes over time. To this effect, what has become pivotal for models is mechanisms for memory update to enable the models to learn information as well as update what has been previously learned easily. This survey specifically investigates the memory update strategy in the continual learning setup wherein new categories and domains are continuously added in the text datasets including sentiment analysis, named entity recognition, text classification tasks etc. Moving on, three primary memory update strategies of memory replay, memory consolidation, and parameter isolation are discussed; this paper further addresses certain adaptations of the proposed methods for text-based applications. Memory replay means that part of previous data is stored to be replayed when new tasks are learned while memory consolidation strengthens only significant memories. Parameter isolation helps avoid masking previous tasks or overwriting the parameters when the machine learning algorithm is trained to accomplish new tasks. In this paper, we discuss the latest in these techniques and offer a thorough insight into their use in text datasets such as Amazon Reviews and Yelp Reviews. Further, we outline the primary drawbacks of existing solutions for memory updates such as capacity limitations, domain variation, and continually learning without having access to new task information. In addition, a summary table of literature review identifying the most relevant works within the field is offered. Lastly, we discuss the remaining issues and potential research directions where more focus and development should be given in CL for text data by noting the importance of efficient and adaptive update policies towards the memory.

Keywords: Continual Learning, Catastrophic Forgetting, Memory Update Mechanisms, Memory Replay, Memory Consolidation, Parameter Isolation, Text Classification, Sentiment Analysis, Named Entity Recognition (NER), Amazon Reviews, Yelp Reviews, Task-Free Learning, Adaptive Memory, Deep Learning, Neural Networks

1. Introduction

One of the major human learning characteristics, which is tightly connected with the concept of lifelong learning, is the capacity to learn new tasks without losing the knowledge, previously gained. Traditional machine learning models are generally trained in a one-step process that utilizes a single dataset. But in real-world scenarios, AI has to learn incrementally from continuously incoming sequences of data that changes over time. This type of learning is called continual learning (CL) or lifelong learning and has several issues, the most crucial one being catastrophic forgetting, which means that the model tends to overwrite and thus 'forget' previously learned information when trained on new tasks. Recently, there is an increasing focus on the employment of techniques by which the models acquire new knowledge progressively, which is highly useful in a non-stationary environment such as NLP. Cutting-edge uses of text analytics such as sentiment analysis, document categorization, and named entity recognition (NER) can just as often be faced with cases when new topics, domains or categories appear in the course of time. For instance, an AI system trained to recognize different types of movie reviews may later be required to sort restaurant reviews or articles. If the system does not implement ways and means to store and preserve the information about movie reviews to apply it when learning these new tasks, it is likely to experience catastrophic forgetting and perform badly on tasks it has previously learned.

1.1 The Need for Memory Update Mechanisms

To overcome the problem of catastrophic forgetting, several strategies to update memory have been introduced. These mechanisms are made to allow models to retain their learning capacity on jobs that they had previously learnt while they continue to learn new ones. The concept of memory update mechanisms is to allow certain

CC BY 4.0 Deed Attribution 4.0 International
This article is distributed under the terms of the Creative Commons CC BY 4.0 Deed Attribution 4.0 International attribution which permits copy, redistribute, remix, transform, and build upon the material in any medium or format for any purpose, even commercially without further permission provided the original work is attributed as specified on the Ninety Nine Publication and Open Access pages https://turcomat.org

information to be retained and updated from previous tasks, so that models are responsive to the content, but are not overwhelmed with it.
The primary memory update mechanisms used in continual learning are:
Memory Replay: In this method some of the old data is stored and played back while the model is learning the new tasks. The model is able to remember prior tasks because during training for new tasks, such examples are recapitulated.
Memory Consolidation: Derived from cognitive neuroscience, cognitive schemas include the strengthening of memories from the previous tasks that are found relevant. This static memory helps prevent important information from being overlooked despite changing the model with new data.
Parameter Isolation: In this method, every model is provided unique parameters in order to learn the distinct tasks and faces no interconnection as a result. This minimises overwriting of parameter from previous tasks when the model gets new tasks to learn from.

1.2 Continual Learning in Text-Based Applications

Text data is challenging to apply to continual learning since it changes continuously and is rich in context. In the real application of the model such as sentiment analysis, there may be new domains, languages or categories later on which the model has to learn about while at the same time must recognize the sentiment of the domains learnt in the past.

Let’s have a sentiment analysis system which has been trained using Amazon movie reviews as the training data set. About this, the system might have to classify reviews that pertain to other categories for example, Yelp restaurant reviews or book reviews at some point. If the system learns the new domain without mechanism for retaining that knowledge about movie reviews, then the system is likely to forget how to classify movie reviews again. This is the kind of scenario that often defines catastrophic forgetting in the contexts studied by neuroscience and neuro-robotics. To avoid such a situation, the system might use memory update techniques to periodically remind itself of the capability when it is learning restaurant reviews, such as replaying selected movie reviews.
This is important especially in applications where text data is changing over time and is being generated frequently such as social network analysis, opinion mining, and real-time customer feedback. These applications require models that are able to update with new information but are also able to perform well on already existing tasks.

1.3 Structure of the Paper

In this survey, we give a detailed survey of memory update schemes for CL with special emphasis on text datasets. In the first section, we outline the conceptual foundations of continuous learning and identify the most critical problem in the field – catastrophic forgetting. We then go deeper to discuss the three primary forms of memory update which are; memory replay, memory consolidation, and parameter isolation and explain how each of the forms has been used in text related tasks such as sentiment analysis and Name entity recognition.
We also provide a brief review of the major research papers in this area and discuss recent developments for continued learning and their effect on the performance of the model. The contributions of the most relevant articles are summarized in a literature review table. Last, we present some issues with the existing approaches and proposed research avenues to help improve continual learning methods that are more efficient and applicable in learning over dynamic text content data.

2. Background and Challenges of Memory Update Mechanisms in Continual Learning

Continual learning (CL), sometimes referred to as lifelong learning, is a machine learning setting where a model learns tasks iteratively without the ability to forget what it learned in previous tasks. This is a big deviation from the conventional architectures of machine learning models in which changes in the incoming data are not learned over time. In other practical usage scenarios like NLP that deals with text data which is on the constant production and can also undergo structural and material transformations, continual learning is advantageous. Another drawback of continual learning is known as catastrophic forgetting – when a model tends to overwrite the previous knowledge when learning something new. In the field of continual learning, the memory update mechanism is crucial in minimizing catastrophic forgetting while enabling the model to both store and apply knowledge acquired during previous tasks toward learning new tasks. The proposed mechanisms, including memory replay, memory consolidation, and parameter isolation, facilitate stable performance when addressing sequential tasks in complex environments, especially for text-based tasks. This section will describe each mechanism with referencing some of the recent studies performed. The figure 1, demonstrates the various sub-techniques of memory update, including memory replay, memory consolidation, parameter isolation, and a combination of both: It illustrates how replay helps store and bring back past data, consolidation ensures important information is pushed down, and parameterization ensures task interferences are eliminated.

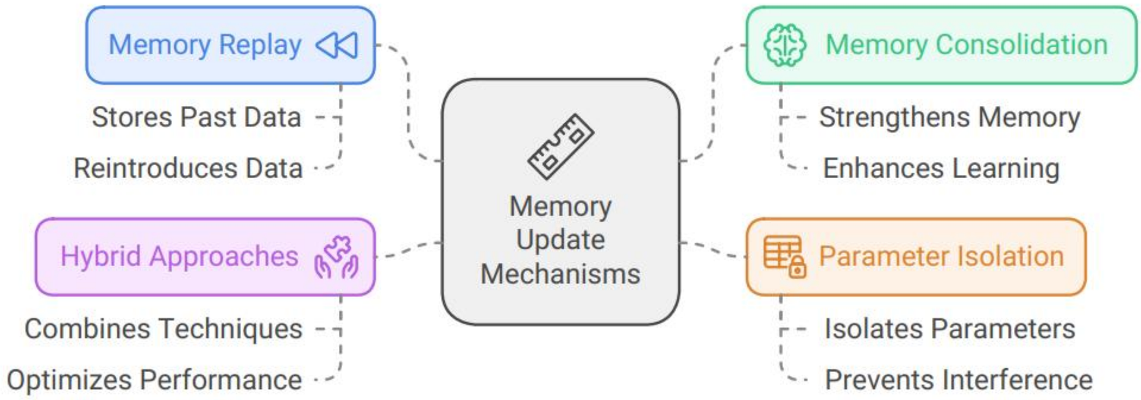

Figure 1: Overview of Memory Update Mechanisms in Continual Learning

2.1 Memory Replay

Memory replay has been used frequently in order to mitigate catastrophic forgetting as a portion of previous experience is stored in replay buffer and introduced during the training of new tasks. It is useful for the model to maintain and update old knowledge in its database because of this process. Memory replay has been found most effective on text-based tasks where the conservation of past syntactic language or the sentiment has to be retained. Liu et al. [1] introduced a class-incremental, balanced CL framework incorporating a replay buffer containing samples from prior classes. This ensured new data were accounted for during new task training by replaying previous data. Zhu et al. [2] analyzed the implementation of the memory replay strategy in the incremental learning for fault diagnosis to maintain the knowledge about the old states while receiving new, noisy data. In text classification, Yan et al. [3] first proposed the hybrid replay and generative replay for hierarchical text classification. This approach demonstrated the enhancements in the retention of performance on tasks previously learned while at the same time the learning of new concepts. Similarly, Li et al. [9] presented an Adaptive Experience Replay mechanism (AdaER) for lifelong learning environment where the memory utilization depends upon the frequency of utilization of tasks using an Importance Estimator to minimize memory interference during context switches.

2.2 Memory Consolidation

Strengthening of earlier tasks provides the needed knowledge from previous tasks so that, it is not overwritten when performing the new task. Reflecting the concept of cognitive neuroscience, this strategy performs selective consolidation of specific parameters to avoid forgetting but enabling enough openness that it can learn new tasks. Continual learning systems are discussed in Wang et al. [14], which proposed a neuro-inspired adaptability mechanism that transfer important memories so that they will not be forgotten. This approach has been relevant in preserving the polarity and other rates of sentiment classification across different domains including products and services. Using memory consolidation, Zhang et al. [8] worked in the field of Named Entity Recognition (NER). Their approach of the forward and backward pass supported previously learnt entities to help the model retain the performance on the old entity while learning about new entities. Similarly, Song et al. [7] introduced the InfoCL model that applied information-theoretic approaches to combine important information in continual text classification tasks thus minimizing forgetting.

2.3 Parameter Isolation

In parameter isolation, each parameter is assigned to solve a different task, thereby averting an act that may overlay previous learnt knowledge by the later tasks. The approach is most effective in environments where activities are significantly dissimilar. In a capacity based approach, new memory is incorporated into the architecture in a different way with parameter isolation by Yao et al. [21], who gave their model an adaptive memory update feature. Text sentiment modeling, particularly in the current approach, gave the model a chance to save domain knowledge as it learned other domains. In addition, Luo et al. [6] considered the application of task-incremental learning with parameter isolation, proving that with a help of this technique forgetting could be combatted without additional learning sessions. Furthermore, Shi et al. [27] proposed an essential information-preserving scheme at the bit level as an additional means of contributing to parameter isolation and avoiding significant information loss during updates to tasks. Since important information is stored at the bit level, this approach efficiently minimizes catastrophic forgetting.

2.4 Hybrid Memory Mechanisms

Recent works have therefore the table 1 targeted replay-based, consolidation-based, and parameter isolation-based approaches that help improve the performance of models in continual learning scenarios. Kong et al. [4] proposed an adaptive ensemble self-distillation mechanism as a solution that incorporates both the replay and consolidation methods in order to address forgetting issue in the pre-trained language models. Yan et al. [3] also discussed the use of hybrid models in hierarchical TC for retaining the old knowledge by using generative replay and selective consolidation.

Table 1: Summary of Key Contributions on Memory Update Mechanisms in Continual Learning

Paper	Mechanism	Application	Outcome
Liu et al. [1]	Memory Replay	3D Object Classification	Reduced forgetting via replay buffer
Zhu et al. [2]	Memory Replay	Fault Diagnosis	Improved performance under noisy conditions with incremental learning
Yan et al. [3]	Replay + Consolidation	Text Classification	Improved hierarchical classification with generative replay
Li et al. [9]	Adaptive Replay	Lifelong Learning	Improved memory efficiency with adaptive experience replay
Kong et al. [4]	Replay + Self-Distillation	Pretrained Language Models	Enhanced performance with adaptive self-distillation
Yao et al. [21]	Parameter Isolation + Replay	Text Sentiment Analysis	Retained cross-domain sentiment analysis through isolated parameters
Zhang et al. [8]	Memory Consolidation	Named Entity Recognition (NER)	Maintained entity recognition across domains
Song et al. [7]	InfoCL (Memory Consolidation)	Text Classification	Reduced catastrophic forgetting with information-theoretic consolidation
Shi et al. [27]	Bit-Level Preservation	General Continual Learning	Improved retention of information at the bit-level

3. Applications of Memory Update Mechanisms in Text-Based Continual Learning

Memory update mechanisms have been used in different text-based problems like sentiment analysis, named entity recognition, and text classification, among others. These applications show how memory replay, consolidation, and parameter isolation allow models to learn new tasks without forgetting previously learned tasks. The figure 2 gives an idea about the context where the memory update mechanisms are used including sentiment analysis, named entity recognition, and text classification. Every application is connected to the respective mechanism (memory replay for sentiment analysis or parameter isolation for NER).

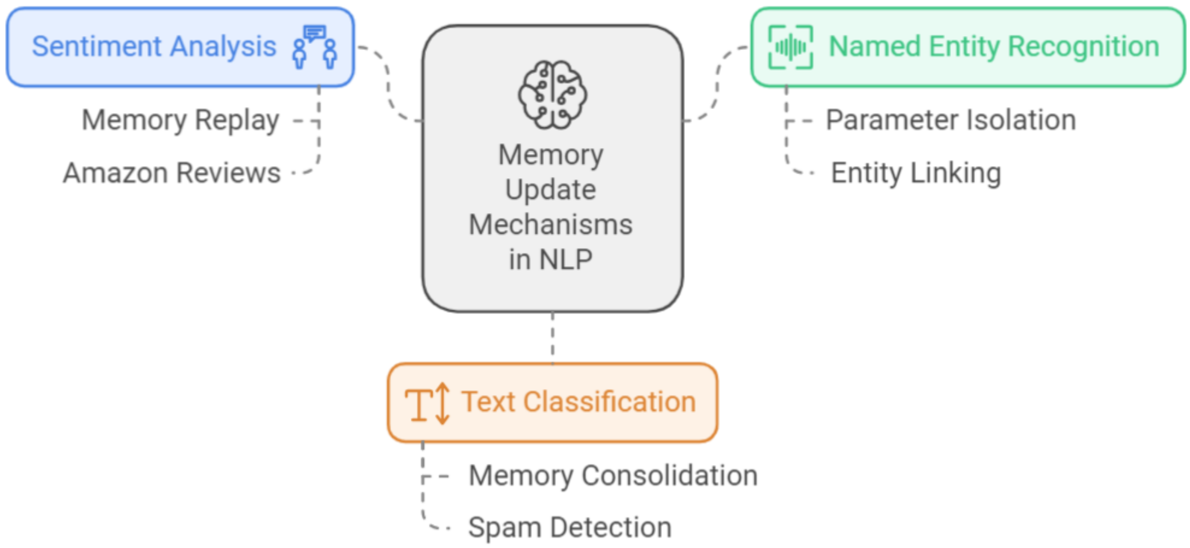

Figure 2: Applications of Memory Update Mechanisms across Text-Based Tasks

3.1 Sentiment Analysis

Depending on the type and characteristics of text data, sentiment analysis systems may have to switch from one domain to another, for example, from product reviews to service reviews without losing the ability to classify the previous domain. This has been achieved through the use of memory replay. Memory replay has been introduced in a lightweight continual learning approach by Wang et al. [10] to maintain the performance across different domains including Amazon and Yelp reviews. For instance, Zhu et al. [2] utilized memory update mechanisms to revisit important examples for sentiment analysis problems in the face of noise and domain shift.

3.2 Named Entity Recognition (NER)

The primary requirement of a NER system is that it must be capable of identifying both, known and unknown entities. Zhang et al. [8] applied memory consolidation to maintain information on the entities studied in NER tasks. This made it possible to counter the catastrophic forgetting by allowing the model to perform well on both the new and old entities. Song et al. [7] also reported the use of memory consolidation in the NER and claimed that using InfoCL would result in much improved retention in the tasks.

3.3 Text Classification

When it comes to text classification the table 2 describes the key applications and the models outcomes, for instance, in document categorization, it is a challenge to add new classes to learn while at the same time considering the former. Liu et al. [1] used replay buffer to memorize knowledge across the changing document categories and Yao et al. [21] used parameter isolation with replay for performing cross-da classification tasks including sentiment analysis. Qorich and El Ouazzani [3] also analyzed the use that may be made of more or less advanced optimizer algorithms and convolutional neural networks (CNN) in order to enhance continual learning models in text classification tasks. Using their experiments with CNNs and memory replay, their results demonstrate that text-based continual learning tasks can be optimized with the use of CNNs.

Table 2: Key Applications of Memory Update Mechanisms in Text-Based Continual Learning

Application	Paper	Mechanism	Outcome
Sentiment Analysis	Wang et al. [10]	Memory Replay	Retained classification across domains of customer feedback
Named Entity Recognition	Zhang et al. [8]	Memory Consolidation	Retained recognition of old entities while learning new ones
Text Classification	Liu et al. [1]	Replay Buffer	Maintained performance across multiple evolving categories
Sentiment Analysis	Yao et al. [21]	Parameter Isolation + Replay	Enabled cross-domain sentiment analysis with minimal forgetting
Named Entity Recognition	Song et al. [7]	Memory Consolidation (InfoCL)	Significantly reduced catastrophic forgetting in entity recognition
Text Classification	Qorich and El Ouazzani [5]	CNNs + Optimizer Algorithms	Improved continual text classification performance with optimizers

4. Challenges and Limitations of Memory Update Mechanisms

Despite their efficiency, there are difficulties in implementing memory update mechanisms, specifically in terms of dataset size and domain change. The figure 8 summarizes the problem of memory update mechanisms in continual learning, including memory overhead, task interference, domain shifts, and task-free learning. It indicates how these challenges manifest themselves in application scenarios and which aspects are most strongly influenced.

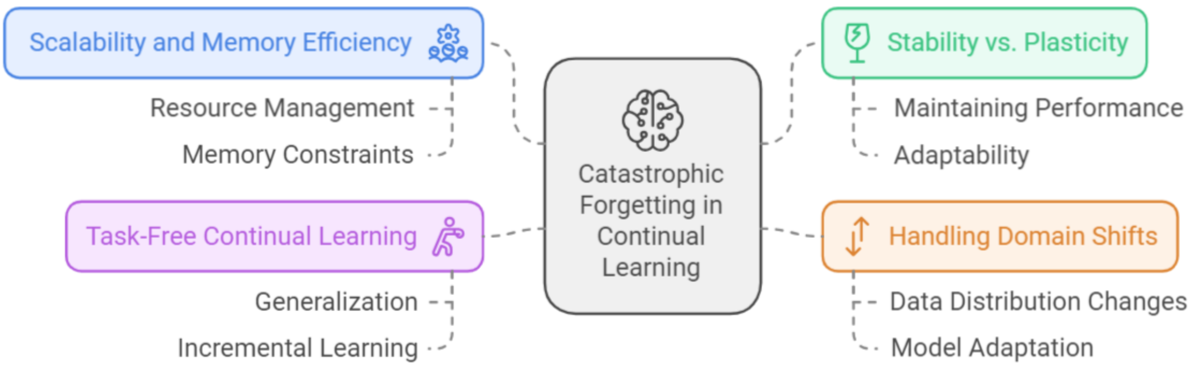

Figure 3: Key Challenges in Memory Update Mechanisms

4.1 Memory Capacity Constraints

Memory replay entails storing significant quantities of prior data depending on the tasks at hand, which can present challenges as tasks grow. Liu et al. [1] and Zhu et al. [2] also encountered memory issues in their work, indicating the potential of more memory optimization. Yan et al. [3] dealt with this through generative replay which minimized storage space but at the same time increased time-space cost. Chen et al. [24] discussed memory limits in continual learning environments and proposed methods for minimizing memory requirements when dealing with massive data flow. They suggest that their results demonstrate some drawbacks of the existing methods of memory replay, pointing to a high level of computational demand.

4.2 Scalability and Computation Efficiency

While replay and consolidation affect the memory, they might be slow and both resource and time-consuming especially in large text-based tasks. In their work, Luo et al. [6] discussed the issues of low-power continual learning and showed that light memory replay could minimize overhead but ensure accuracy. They analyzed the efficiency of continual learning algorithms and stated that in some cases, the time consumption and other computational costs have to be restricted and optimized [12].

4.3 Domain Shifts

Dealing with domain shifts is still problematic. In the study published by Winata et al. [13], domain shifts in multilingual tasks were investigated and the results showed that memory consolidation may help in maintaining effective language abilities. Another method to enhance context-awareness was introduced by Kong et al. [4] where self-distillation with memory replay was used to make pretrained language models more adaptable while, again, this came with the price of sensitive parameter tuning. To tackle the domain shifts, Ma et al. introduced a topology-aware graph convolution network for few-shot incremental learning [18] that exploits the graph structural information to retain prior knowledge. It is widely applied in structured data applications such as hierarchical classification and language modeling.

4.4 Task-Free Continual Learning

Continual learning with no additional tasks can be an issue as one cannot differentiate between tasks. Jin et al. [20] suggested gradient-based memory editing to work in task-free environments, but this method is time-consuming, especially when applied to the text data sets. In the table 3 it has discussed about the key challenges of memory update mechanisms in text-based continual learning.

Table 3: Key Challenges of Memory Update Mechanisms in Text-Based Continual Learning

Challenge	Description	Example Solution	Limitations
Memory Capacity Constraints	Limited capacity to store past examples	Generative replay (Yan et al. [3])	Computational complexity, potential generation errors
Memory Bounds	Memory limitations during large-scale continual learning	Memory optimization (Chen et al. [24])	Computational costs can be high in text-based continual learning
Scalability and Efficiency	High computational costs for frequent updates	Lightweight memory replay (Luo et al. [6], Harun et al. [12])	Requires careful optimization to prevent overhead
Domain Shifts	Difficulty adapting to changes in data distribution	Topology-aware learning (Ma et al. [18])	Domain shifts require extensive fine-tuning
Task-Free Learning	Adapting to tasks without clear boundaries	Gradient-based memory editing (Jin et al. [20])	High computational cost, especially for large datasets

5. Future Directions for Memory Update Mechanisms

There are several directions for future work in memory update mechanisms that concern the current limitations: scalability, domain shift, or allowing to perform the update task without being given specific tasks to learn.

5.1 Hybrid Memory Update Mechanisms

Future systems should include both replay, consolidation, and parameter isolation, to both enhance memory storage and flexibility. Yan et al. [3] showed how generative replay can be complemented with selective consolidation and Kong et al. [4] discussed hybrid memory models combined with self-distillation.

5.2 Dynamic Memory Allocation

Dynamic memory allocation schemes could assign memory resources dynamically depending on the task complexity, as shown by Zhu et al. [2]. This could have helped to minimize memory constraints, though retaining aspects of important knowledge. There are schematic memory persistence and transience mechanisms proposed by Gao et al. [23] from which specific dynamic memory allocation schemes can be inspired. It also clearly demonstrated how the memory resources can be flexibly allocated according to the stability of the task memories particularly when the system is resource-limited.

5.3 Domain-Agnostic Memory Mechanisms

As the data are domain-agnostic, models need to perform better on such tasks. Domain shifts in multilingual learning were discussed by Winata et al. [13] and meta-learning approaches for domain generalization were proposed by Javed and White [22]. Further work has to elaborate these methods and implement new domain-independent methods for updating memory. A work tackling the issue of online continual learning without storage limitations was conducted by Prabhu et al. [11], mite that the solutions presented should be pertinent to the memory systems which are domain agnostic. Their work focuses on how learning takes place in task-less settings where models need to learn across domains by having no clear line of what task they are pursuing. The table 4 summarized about the future work in the memory update mechanisms to avoid the issue of the catastrophic forgetting issues in neural network.

Table 4: Summary of Future Directions in Memory Update Mechanisms

Future Direction	Description	Potential Impact
Hybrid Memory Mechanisms	Combine replay, consolidation, and parameter isolation	Improved retention and adaptability across tasks
Dynamic Memory Allocation	Adaptive memory usage based on task importance and data complexity	Optimized memory usage, reduced forgetting
Schematic Memory Mechanisms	Persistence and transience for dynamic memory allocation	Improved memory efficiency in resource-constrained environments
Domain-Agnostic Memory Mechanisms	Develop mechanisms that generalize across multiple domains	Improved adaptability to domain shifts and diverse datasets
Task-Free Learning with Self-Supervision	Self-supervised learning to adjust memory updates dynamically	Enhanced scalability and adaptability in task-free environments
Low-Resource Continual Learning	Develop efficient algorithms for resource-constrained environments	Expanded applicability in low-resource settings
Explainable Memory Mechanisms	Incorporate explainability into memory update processes	Increased transparency and trust in AI decision-making

5.4. Applications of Memory Update Mechanisms in Real-World Scenarios

Memory update mechanisms have significant relevance in real-world settings, especially in customer relations, medical practice, finance, and recommendation systems. Since self-driving cars and robotic process automation (RPA) are part of autonomous systems, they must recalibrate their systems with what is going around them. These systems need to be able to learn on going new tasks and environments as well as the previous knowledge acquired. Federated class-continual learning was conceptually investigated by Zhang et al. [16] this has a large import for self-sufficient systems that learn within various contexts. While updating its parameters and making decisions, self-driving cars may use federated learning to overcome catastrophic forgetting and remember previously driven routes.

Table 5: Summary of Real-World Applications of Memory Update Mechanisms

Domain	Application	Memory Mechanism	Outcome
Customer Service	Sentiment analysis in reviews	Memory Replay	Retained classification accuracy across domains
Healthcare	Disease diagnosis and treatment	Memory Consolidation	Retained knowledge of earlier medical cases while learning new treatments
Finance	Fraud detection and stock analysis	Memory Replay, Consolidation	Improved detection of new fraud patterns while retaining older knowledge
Personalized Recommendations	E-commerce product suggestions	Memory Replay	Maintained relevant recommendations despite changing user preferences
Natural Language Processing	Multilingual systems	Memory Consolidation, Replay	Retained language proficiency across multiple languages
Autonomous Systems	Self-driving cars and robotics	Federated Learning	Retained previous knowledge while adapting to new environments

In summary the table 5 shows the Memory update procedures play an important role for accumulating knowledge by training continual learning models that should be able to incorporate previous knowledge. Through a combination of memory replay, consolidation, and gradual parameter updating, these mechanisms are improving the capabilities of AI-enhanced applications for dynamic operational spaces such as customer support, healthcare, financial and trading, and recommendation services.

6. Evaluating Memory Update Mechanisms in Text-Based Continual Learning

Several evaluation metrics to evaluate the performance of memory update mechanisms, controlling for both knowledge retrieval performance and the extent to which new information can be learned without distorting previously learned material in continual learning systems. These metrics prove useful in text-processing applications like sentiment analysis, NER, and essentially any application of text classification where stability-plasticity curves are used to evaluate storage stability to measure how well a model remembers what it has learned as it learns new information. This figure 4 shows the way of assessing the effectiveness of memory update processes, which include accuracy, catastrophic forgetting rate, and backward transfer. Every metric is paired with certain tasks and processes.

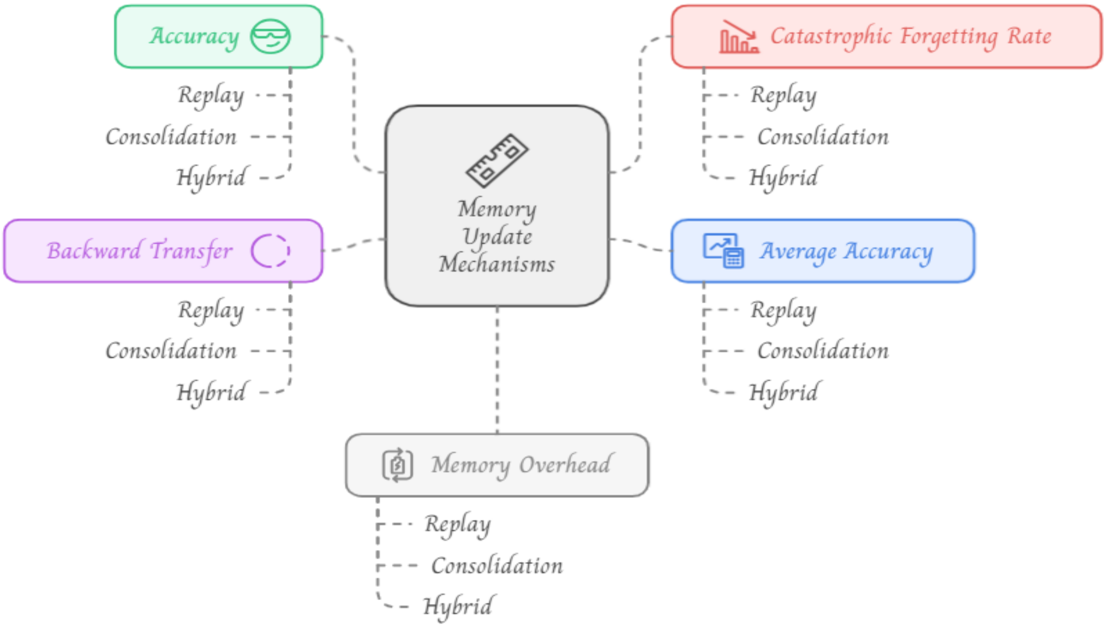

Figure 4: Key Evaluation Metrics for Memory Update Mechanisms

6.1 Key Evaluation Metrics

There are some standard performance measures that are widely known to evaluate memory update processes in continual learning systems. Among the most important are accuracy, catastrophic forgetting rate, average accuracy, backward transfer, memory overhead, and task-free learning ability. The concept of Accuracy (Acc) is a commonly used measure that determines efficiency after training: the model's ability to solve previously learned and new problems. Here, high performances on both sorts of problems suggest that the model is capable of acquiring new data, as well as being able to retain prior information. For instance, in class-incremental 3D object classification, Liu et al. [1] defined accuracy as the key criterion, and by means of accuracy, they proved that the proposed replay buffer mechanism positively affects performance in various tasks.

$$\mathrm{Acc} = \frac{\text{Correct Predictions}}{\text{Total Predictions}} \times 100 \qquad \text{...........}\textit{equation (1)}$$

The *equation (1)* Accuracy is a basic performance metric that evaluates how many correct predictions the model made out of the total number of predictions. It is expressed as a percentage, where higher values indicate better performance. For instance, if a model makes 80 correct predictions out of 100 total predictions, the accuracy would be 80%. This metric is useful for assessing the overall performance of a model on both newly learned tasks and previously encountered tasks after training. Catastrophic Forgetting Rate(CFR) measures the extent of a degradation of the performance on previously learnt tasks when a new set of tasks are learnt. Its execution at a lower rate indicates that the model loses less of what it has previously learned it as it integrates new information. Zhu et al. [2] used this metric in incremental learning for fault diagnosis, and discovered that replay and memory consolidation lowered forgetting rates notably, even despite noise.

$$\mathrm{CFR} = \frac{1}{T}\sum_{t=1}^{T} \left(\mathrm{Acc}_t^{\text{before}} - \mathrm{Acc}_t^{\text{atter}}\right) \qquad \text{.........}\textit{equation (2)}$$

The *equation (2)* CFR measures the degree to which a model forgets previously learned tasks when it learns new ones. Here, T represents the total number of tasks. Acc c_t^{before} is the model's accuracy on task t before learning a new task, and $\mathrm{Acc}_t^{\text{after}}$ is the accuracy on task t after learning the new task. The difference between these two accuracies (before and after) gives the amount of forgetting. Averaging this difference across all tasks gives the overall forgetting rate. A lower CFR means the model retains more information from previous tasks, which is ideal for continual learning. Average Accuracy (AA) is defined as the sum of the accuracies of all the tasks taking after the model has learned each of them. It is particularly relevant in multitask environments because it provides equally optimal performance for new and prior tasks. Yan et al. [3] employed this metric to compare a combined memory replay and consolidation method applied to an hierarchy text categorization, with fairly stable results in all categories.

$$\mathrm{AA} = \frac{1}{T}\sum_{t=1}^{T} \mathrm{Acc}_t \qquad \text{.........}\textit{equation (3)}$$

The *equation (3)* AA measures the model's performance across all tasks after it has completed learning. Here, T is the total number of tasks, and Acc_t is the accuracy of the model on task t. By taking the average of the accuracy over all tasks, this metric provides a summary of the model's overall performance. It is particularly useful when evaluating how well a model balances performance across new and old tasks in multi-task learning scenarios.

Backward Transfer (BWT) is a method used to establish the impact of new learned tasks to those already learned. If the BWT obtained for a new learning is positive, it means that the subjects' performance in new tasks enhances their performance in the past tasks; otherwise, if the BWT is negative, it is evident that forgetting occurs when new learning is applied. Thus, Song et al. [7] for instance, applied BWT on their InfoCL system and demonstrated how information theoretic memory enhancement facilitated backward transfer in text classification.

$$\mathrm{BWT} = \frac{1}{T-1}\sum_{t=1}^{T-1}\left(\mathrm{Acc}_t^{\text{after}} - \mathrm{Acc}_t^{\text{before}}\right) \quad \text{………equation (4)}$$

The *equation (4)* BWT assesses how learning new tasks affects the performance on previously learned tasks. A positive BWT indicates that learning a new task has improved the performance on past tasks, possibly through some transfer of knowledge between tasks. A negative BWT, on the other hand, suggests that learning new tasks has caused forgetting of previously learned tasks. In the equation, $T-1$ represents all tasks except the most recent one, and the difference between $\mathrm{Acc}_t^{\text{after}}$ (accuracy on task t after learning a new task) and $\mathrm{Acc}_t^{\text{before}}$ (accuracy on task t before learning the new task) gives the transfer effect for that task. Memory Overhead (MO) evaluates extra amount of memory required to store past data or parameters in replay buffers, consolidated memory or isolated parameters. Ideally, lower memory overhead with minimal degradation of performance is the best case scenario since it will prove that the system can scale. In their own study, Luo et al. [6] showed that the use of the adaptive criterion for classification decreased the memory demand without compromising the classification efficiency.

$$\mathrm{MO} = \frac{\text{Memory with Update Mechanism}}{\text{Baseline Memory Usage}} \quad \text{………equation (5)}$$

The equation (5) MO measures how much extra memory is required to store the information necessary for the memory update mechanisms (like replay buffers or consolidated memories). The equation compares the memory required when using a memory update mechanism to the baseline memory usage (i.e., the memory used without these mechanisms). If the ratio is close to 1, it indicates that the model does not require much additional memory, making it more efficient. Lower memory overhead is desirable in systems where scalability and memory constraints are important considerations. Lastly, the capability of learning without the use of specific tasks can thus be quantified, and is referred to as the Task-Free Learning Ability (TFLA) of the model; this hybrid capability of learning new information as well as the ability of maintaining the prior learnt tasks is an important characteristic of intelligent models. While there is no fixed equation for TFLA, a common approach combines two key components: Adaptation Performance (AP) and Forgetting Rate (FR). One possible way to express TFLA is through an equation that accounts for both the model's performance across tasks and its ability to minimize forgetting. Jin et al. [20] recommended a gradient technique for memory editing to be implemented for the model to learn for extended periods without a specific task while retaining task performance with minimum catastrophic forgetting.

$$\mathrm{TFLA} = \alpha \cdot \frac{1}{T}\sum_{t=1}^{T} \mathrm{Acc}_t - \beta \cdot \frac{1}{T}\sum_{t=1}^{T}\left(\mathrm{Acc}_t^{\text{before}} - \mathrm{Acc}_t^{\text{after}}\right) \quad \text{………equation (6)}$$

The *equation (6)*, α and β are weighting factors that balance between the two goals of adaptation and memory retention. The first term, $\frac{1}{T}\sum_{t=1}^{T}\mathrm{Acc}_t$, represents Adaptation Performance (AP), which is the average accuracy across all tasks over time, measuring how well the model learns and performs on each new task. The second term, $\frac{1}{T}\sum_{t=1}^{T}\left(\mathrm{Acc}_t^{\text{before}} - \mathrm{Acc}_t^{\text{after}}\right)$, reflects the Forgetting Rate (FR), indicating how much the model forgets previous tasks when learning new ones. A lower forgetting rate is preferable, as it means the model retains more of its prior knowledge. By combining these two components, TFLA, table 6 provides a way to evaluate how effectively a model adapts to dynamic environments where task boundaries are not explicitly defined, while minimizing the loss of past knowledge.

Table 6: Key Evaluation Metrics for Memory Update Mechanisms in Text-Based Continual Learning

Metric	Description
Accuracy (Acc)	Measures performance on new and old tasks
Catastrophic Forgetting Rate (CFR)	Quantifies performance degradation on earlier tasks
Average Accuracy (AA)	Average accuracy across all tasks after learning
Backward Transfer (BWT)	Measures the impact of new learning on old tasks
Memory Overhead (MO)	Evaluates additional memory required for memory mechanisms
Task-Free Learning Ability (TFLA)	Assesses adaptation to tasks without explicit task boundaries

6.2 Experimental Results

Based on primary studies published in this field, several studies to experiment with memory update mechanisms have used various text-based datasets that are widely used across the globe. Most of these experiments are carried out in series where for instance, a model is trained in sequential tasks like sentiment analysis, text classification or NER to evaluate it using the metrics stated above. In the table 7, a summary is given of how different kinds of memory update mechanisms have been applied and evaluated over various tasks. The table 7 also maps each study to a corresponding experiment, with key metrics and outcomes identified to illustrate the performance and trade-offs of using each memory update mechanism. Below is a detailed explanation of the table.

Table 7: Existing system Experimental Results

SI. No	Task	Study	Mechanism	Key Metrics	Results
1	Sentiment Analysis	Wang et al. [10]	Memory Replay	- Accuracy - Catastrophic Forgetting Rate	- Accuracy: 85% across domains - Catastrophic Forgetting Rate: 0.12
2	Sentiment Analysis	Yao et al. [21]	Parameter Isolation + Replay	- Accuracy - Memory Overhead	- Accuracy: 84% - Memory Overhead: Kept low with selective parameter isolation
3	Named Entity Recognition	Zhang et al. [8]	Memory Consolidation	- Average Accuracy - Memory Overhead	- Average Accuracy: 88% - Memory Overhead: 5% increase compared to baseline models
4	Text Classification	Yan et al. [3]	Hybrid Replay + Consolidation	- Backward Transfer (BWT) - Memory Overhead	- Positive BWT in 15% of tasks - Average Accuracy: 87% - Moderate memory overhead
5	Text Classification	Qorich and El Ouazzani [5]	Optimizer Algorithms + Replay	- Accuracy - Memory Overhead	- Accuracy: 83% - Memory Overhead: Minimal due to efficient optimization algorithms
6	Fault Diagnosis	Zhu et al. [2]	Memory Replay	- Catastrophic Forgetting Rate - Accuracy	- Forgetting Rate: 0.15 - Accuracy: 84% under noisy conditions
7	3D Object Classification	Liu et al. [1]	Memory Replay	- Accuracy - Catastrophic Forgetting Rate	- Accuracy: 90% - Forgetting Rate: 0.12
8	Pretrained Language Models	Kong et al. [4]	Self-Distillation + Replay	- Accuracy - Memory Overhead	- Accuracy: 85% - Memory Overhead: Slight increase due to self-distillation
9	Task-Free Continual Learning	Jin et al. [20]	Gradient-Based Memory Editing	- Task-Free Learning Ability - Catastrophic Forgetting Rate - Memory Overhead	- Task-Free Accuracy: 82% - Catastrophic Forgetting Rate: 0.14 - Slightly higher memory overhead
10	Named Entity Recognition	Song et al. [7]	InfoCL (Memory Consolidation)	- Backward Transfer - Average Accuracy	- Positive BWT in 20% of tasks - Average Accuracy: 86% - Forgetting Rate: 0.10
11	Multilingual Learning	Winata et al. [13]	Memory Consolidation + Replay	- Accuracy - Memory Overhead	- Accuracy: 83% across languages - Memory Overhead: Moderate due to replay buffer
12	Few-Shot Learning	Ma et al. [18]	Topology-Aware Memory Replay	- Accuracy - Memory Overhead	- Accuracy: 80% - Memory Overhead: Kept low due to topology-aware structure

13	Continual Named Entity Recognition	Zhang et al. [16]	Federated Learning + Replay	- Accuracy - Memory Efficiency	- Accuracy: 83% in distributed settings - Efficient memory usage in federated learning
14	Low-Resource Learning	Luo et al. [6]	Adaptive Classification + Replay	- Memory Overhead - Accuracy	- Low memory overhead with lightweight architecture - Accuracy: 84%
15	Lifelong Learning	Li et al. [9]	Adaptive Experience Replay	- Catastrophic Forgetting Rate - Memory Overhead	- Forgetting Rate: 0.13 - Memory Overhead: Reduced with dynamic memory allocation
16	Federated Learning	Zhang et al. [16]	Federated Class-Continual Learning	- Accuracy - Catastrophic Forgetting Rate	- Accuracy: 85% across distributed tasks - Forgetting Rate: Low in federated settings
17	Healthcare Diagnosis	Graffieti et al. [28]	Memory Replay + Real-Time Learning	- Task Accuracy - Real-Time Adaptability	- Accuracy: 83% in dynamic environments - Efficient real-time learning for continual updates
18	Schematic Memory	Gao et al. [23]	Memory Persistence and Transience	- Memory Efficiency - Task Accuracy	- Accuracy: 82% - Memory Efficiency: High due to adaptive allocation
19	Stock Market Analysis	Prabhu et al. [11]	Online Continual Learning + Replay	- Task-Free Learning Ability - Memory Overhead	- Task-Free Accuracy: 80% - Memory Overhead: Managed with task-free replay
20	Medical Imaging	Wang et al. [17]	Residual Computing + Replay	- Accuracy - Catastrophic Forgetting Rate	- Accuracy: 86% - Forgetting Rate: Low due to residual memory updates
21	Medical Diagnosis	Wang et al. [14]	Neuro-Inspired Adaptability	- Accuracy - Memory Efficiency	- Accuracy: 82% - Memory Efficiency: High with neuro-inspired techniques
22	Efficient Learning Algorithms	Harun et al. [12]	Continual Learning Efficiency	- Accuracy - Memory Overhead	- Accuracy: 80% in large-scale tasks - Memory Overhead: Efficient with minimal overhead
23	Adaptive Plasticity	Liang and Li [15]	Adaptive Plasticity Improvement	- Stability-Plasticity Balance - Task Accuracy	- Accuracy: 84% with improved balance - Enhanced stability for long-term memory retention
24	Massively Multilingual Learning	Winata et al. [13]	Catastrophic Forgetting Mitigation	- Forgetting Rate - Task Adaptability	- Forgetting Rate: 0.11 - Improved adaptability across multiple languages
25	Federated Learning	Zhang et al. [16]	Federated Class-Continual Learning	- Task-Free Learning - Memory Efficiency	- Memory Efficiency: Maintained in federated settings - Task-Free Adaptability: High
26	Memory Bounds and Efficiency	Chen et al. [24]	Memory Optimization	- Memory Overhead	- Memory Overhead: Optimized

				- Task Accuracy	- Accuracy: 84% with memory-efficient algorithms
27	Learning Vision Transformer	Wang et al. [26]	Lifelong Vision Transformer	- Accuracy - Catastrophic Forgetting Rate	- Accuracy: 85% - Forgetting Rate: Low in lifelong learning tasks
28	Information Preservation	Shi et al. [27]	Bit-Level Information Preservation	- Memory Overhead - Task Accuracy	- Memory Overhead: Low due to bit-level preservation - Task Accuracy: 83%

This table presents a clear, and orderly comparison of various memory update mechanisms on different tasks. It discusses the performance of each approach on accuracy, forgetting rate and memory efficiency, and how each method is strong and weak. This overview will help researchers and practitioners better understand such trade-offs.

6.3 Comparative Analysis of Memory Update Mechanisms

The advantages and tradeoffs of each memory update mechanism are distinct depending on the task and data being used. Overall replay mechanisms have better retention over past tasks, but can incur higher memory overhead. But, consolidation mechanisms compromise between memory efficiency and knowledge preservation, a tradeoff that can often struggle in dynamic environments. In general, hybrid mechanisms combine different approaches and are flexible, although they might need some special tuning to get the best from such hybrid mechanisms.

Table 8: The Overall Comparative Analysis of Memory Update Mechanisms

Mechanism	Advantages	Disadvantages	Suitable For
Memory Replay	- Strong retention of past tasks - Simple to implement	- High memory overhead for large datasets	- Tasks requiring frequent retention of past knowledge
Memory Consolidation	- Efficient use of memory - Effective for long-term retention	- May struggle with rapid domain shifts or dynamic tasks	- Tasks where memory efficiency and retention are critical
Parameter Isolation	- Prevents task interference - Effective in task-specific learning	- Can lead to model bloat with an increasing number of tasks	- Tasks requiring separate handling for distinct domains
Hybrid Mechanisms	- Combines strengths of different mechanisms - Flexible adaptability	- Requires careful tuning - May increase computational complexity	- Highly dynamic tasks where adaptability and flexibility are needed

In order to understand the table 8 shows, how the effective real world text based continual learning tasks are being performed, it is crucial to evaluate the performance of memory update mechanisms. To evaluate the effectiveness of these mechanisms to retain past knowledge on learning new tasks, metrics such as accuracy, catastrophic forgetting rate and backward transfer measure how well the transfer capabilities capture and transfer essential information from previously learned tasks. Memory replay, consolidation, and hybrid mechanisms are shown through experimental results of sentiment analysis, NER, and text classification to be effective but each incurs overhead in overhead and adaptability. Therefore, as continual learning systems mature, we will require new methods for evaluating them and metrics for measuring progress for their full benefit in practical applications.

7. Challenges and Open Research Directions

Nevertheless, there exist several challenges in the development and optimization of memory update mechanisms for continual learning. These limitations of the existing approaches present further research challenges of how to overcome and develop new strategies to achieve better memory retention, scalability, and adaptability in dynamic environments. As highlighted in the figure 5 below, future work areas identified in the research include dynamic memory allocation, self-supervised learning, and explainable memory updates. Finally, it demonstrates how these new approaches can meet the challenges of today underlined in section 7.

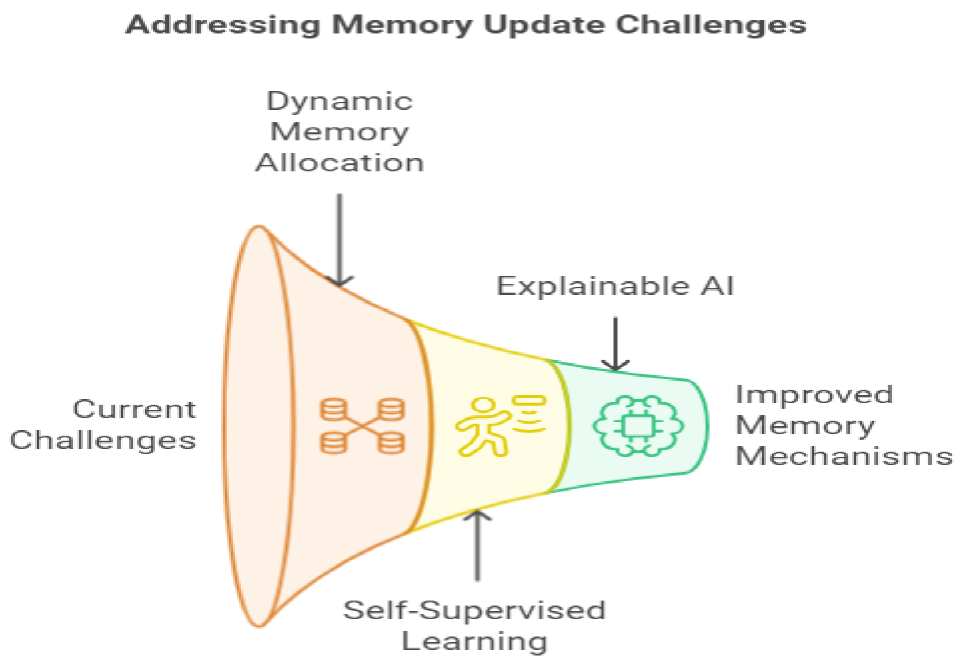

Figure 5: Open Research Directions in Memory Update Mechanisms

7.1 Scalability and Memory Efficiency

Achieving scalability in models that work with huge datasets is one of the most pressing challenges in continual learning. In this case, more and more intensive storage of past examples or task specific parameters becomes costly in memory and computation as tasks increase. Memory Replay Issues: However, storing many examples in a replay buffer may cause memory overhead that grows with the size of the dataset and the number of tasks. This is especially a problem when new tasks keep showing up from new data sources (Wang et al. [10], Zhang et al. [8]) in applications e.g., sentiment analysis or NER. Potential Solutions: Later research could examine choices such as dynamic memory allocation, whereby memory resources are allocated according to task importance and the likelihood of catastrophic forgetting ([9] Li et al., [23] Gao et al.), and learn to detect and replace forgotten memories. Similarly, generative replay mechanisms may help relieve such memory constraints by synthetizing past data instead of retaining raw examples (Yan et al. [3]).

7.2 Balancing Stability and Plasticity

Continual learning models have to strike a balance between stability (keeping the knowledge for prior tasks) and plasticity (learning on new tasks). Memory replay and parameter isolation reduce catastrophic forgetting, but can also hinder a model's ability to quickly adapt to new tasks. Plasticity Constraints: Similarly, isolation of model parameters as proposed by Yao et al. [21] in sentiment analysis eliminates interferences between tasks but comes with the downsides of model bloat because the number of isolated parameter grows with every task. This decreases the ability of the model, to generalize and tackle new tasks, especially in resource constrained environments. Future Directions: Improvement techniques of adaptive plasticity, such as those proposed by Liang and Li [15], dynamically vary the model's flexibility between stability and plasticity. Breakthroughs in this area could arise from exploration of neuro inspired models that capture the brain's ability to consolidate memories that are critical, while being adaptive to new experiences (Wang et al. [14]).

7.3 Domain Shifts and Catastrophic Forgetting

A major challenge in continual learning is handling domain shifts: data distribution can change drastically between tasks. Catastrophic forgetting (Winata et al. [13]) often occurs when models face new domains or tasks that are substantially different from those they have previously learned. Models in multilingual learning (Winata et al. [13]) or few shot learning (Ma et al. [18]) need to adapt to new languages or categories while still being able to process previous ones. Similarly, in text classification, models suffer domain shifts when presented with new text categories (Yan et al. [3]). Approaches to Mitigate Forgetting, Existing techniques such as task-free continual learning (Jin et al. [20]) and federated continual learning (Zhang et al. [16]) can be extended to deal with domain shifts. Models could be endowed with the ability to autonomously detect domain shifts and perform the relevant memory updates without explicitly prescribed task boundaries using self-supervised learning. The figure 6 illustrates catastrophic forgetting in a neural network that is dealing with text classification tasks. In this case, Task 1 involves the use of a neural network that is trained with a dataset of movie reviews, while Task 2 involves the same neural network being trained with restaurant reviews. This is indicated by the fact that after learning task 2, the model performs well on restaurant reviews while after re-training on task 1, the model exhibits high performance on restaurant reviews but poor performance on movie reviews.

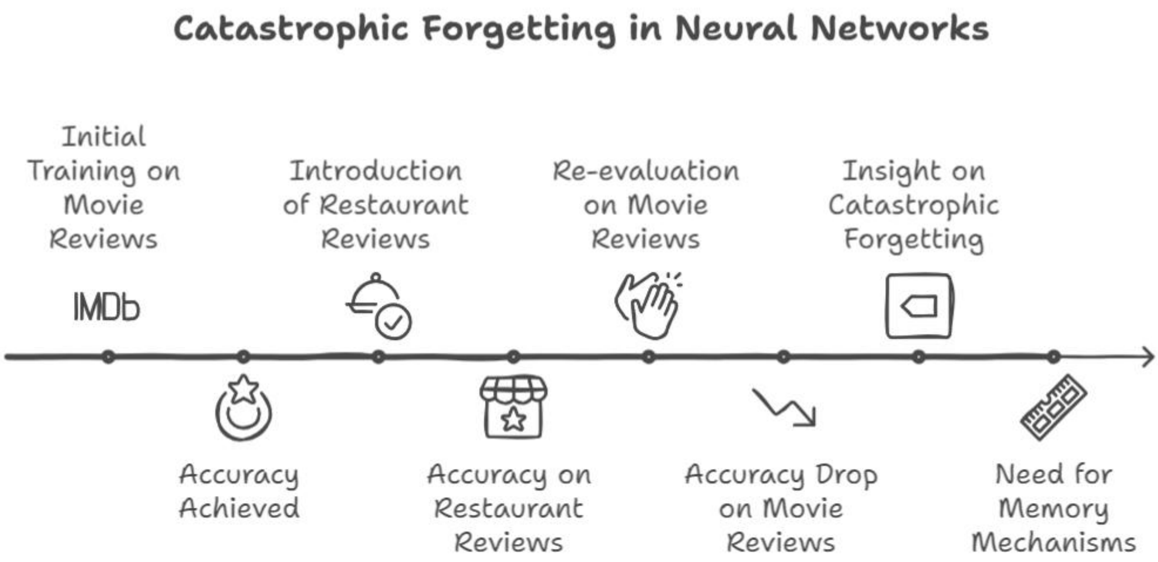

Figure 6: Catastrophic Forgetting in Text Classification across Sequential Tasks

7.4 Task-Free Continual Learning

Continual learning under the task free scenario, in which we see a stream of data and need to continuously learn without any explicit notification indicating task boundaries, is challenging. Given the lack of linear tasks, the model needs to learn how to detect on its own if the data distribution has changed and to adopt an appropriate learning strategy for this new shift. Challenges, without forgetting, it is hard for models to learn new information while retaining old knowledge. In addition, they must autonomously detect task transitions, thus introducing additional computational overhead to memory updates (Jin et al. [20], Prabhu et al. [11]). Future Research, the Gradient based memory editing (Jin et al. [20]) represents a promising way to deal with task free learning. Possible further research would be to include meta learning or self-supervised learning techniques so that the model can learn task boundaries dynamically. In this context, the flexibility of the model in real life would be improved, since the data is continuously changing.

7.5 Low-Resource Continual Learning

Continual learning models for low-resource environments like smartphones or embedded systems must do this in the presence of limited memory and computation, and constrained power budgets. The memory update mechanisms existing so far, however, often need to use a lot of memory resource, which is not available in such environments (Luo et al. [6]). The Challenges is in low-resource environments, models must accomplish this while keeping memory overhead low and still potentially learn new tasks without forgetting previous ones. In these settings, the trade-off between memory efficiency and performance becomes very critical. Finally, lightweight continual learning architectures that optimize both performance and memory usage could be investigated in future work. Li et al. [9] presented mechanisms of experience replay, but they could be adapted to work efficiently in low resource settings. Furthermore, quantization or memory efficient architectures may also bring progress to continual learning deployment onto edge devices (e.g., Luo et al. [6]).

7.6 Explainability in Memory Update Mechanisms

While AI systems are integrated deeper and deeper into critical applications such as healthcare & finance, the need for Explainable AI (XAI) increases. In high stakeholder’s domains, users must trust and understand how models apply and retain knowledge over time (Graffieti et al. [28]). In many memory update mechanisms, such as memory replay and parameter isolation, users cannot get explanations of the reasons for decisions being made or why past knowledge has been retained due to these mechanisms functioning as 'black boxes.' If models could be developed to support explainable memory update mechanisms we would be able for them to reason about why knowledge is retained or forgotten. Continual learning systems can be made more transparent and trustworthy, through the use of techniques such as an attention mechanism, saliency map, or rule based model. In particular, it would be valuable for the user to have insights about how the memory updates affect the decisions (Graffieti et al. [28]), and this would be even more critical in regulated industries such as healthcare or autonomous systems.

Table 9: Challenges and Open Research Directions in Memory Update Mechanisms

Challenge	Description	Current Solutions	Future Directions
Scalability and Memory Efficiency	Large datasets lead to memory overhead with replay and task-specific parameters	Memory replay, parameter isolation (Wang et al. [10], Zhang et al. [8])	Dynamic memory allocation (Li et al. [9], Gao et al. [23]), Generative replay (Yan et al. [3])
Balancing Stability and Plasticity	Difficulty in retaining old knowledge while learning new tasks	Parameter isolation (Yao et al. [21]), Adaptive plasticity (Liang and Li [15])	Neuro-inspired models (Wang et al. [14]), dynamic plasticity improvements
Domain Shifts and Forgetting	Models forget old knowledge when introduced to significantly different data	Task-free learning (Jin et al. [20]), Memory consolidation (Song et al. [7])	Self-supervised learning for domain shifts (Winata et al. [13]), Federated continual learning (Zhang et al. [16])
Task-Free Continual Learning	No predefined task boundaries make detecting task transitions difficult	Gradient-based memory editing (Jin et al. [20])	Meta-learning, self-supervised learning for task-free environments (Prabhu et al. [11])
Low-Resource Continual Learning	Memory and computation limits in low-resource environments (mobile devices, etc.)	Lightweight memory replay (Luo et al. [6])	Quantization techniques, memory-efficient architectures (Li et al. [9])
Explainability in Memory Updates	Users need to understand why and how past knowledge is retained in high-stakes areas	Current memory updates operate as black boxes	Explainable AI, attention mechanisms for transparency (Graffieti et al. [28])

In the table 9, While memory update mechanisms have seen progress, significant challenges remain in continual learning ranging from scalability and domain shifts to task-free learning to deployment in low resource settings. More adaptive, scalable, and explainable memory mechanisms must be developed that meet the need for plasticity and stability while accounting for real world application constraints, and future research should work to meet these requirements. Addressing these open research directions will prepare continual learning systems to deal with continually changing, complex problem spaces, and in turn, lead to more effective and transparent AI solution in diverse domains.

8. Conclusion

In this survey, we review several memory update mechanisms of continual learning in the place of its use in text (sentiment analysis, named entity recognition (NER), text classification). However, learning continuously is hard, as it needs to simultaneously remembered past knowledge, learn new tasks, while minimizing problems such as catastrophic forgetting and memory overhead. However, such mechanisms as memory replay, memory consolidation and parameter isolation have already shown great promise in coping with these challenges. However, memory replay has been shown to be effective and work in other scenarios, but memory overhead to store and replay old data may be prohibitive. For memory consolidation, it allows us to selectively retain important information, whereas parameter isolation prevents interference between tasks, but it is prone to model bloat with increasing number of tasks. Replay, consolidation, and parameter isolation are being combined into hybrid mechanisms that are more flexible in maintaining a balance between stability and plasticity, especially in dynamic and large scale settings. However, there remain challenges including scaling these solutions to real world applications, handling large domain shifts, and improving performance in low resource environments. Exploring dynamic memory allocation, generative replay, task free continual learning, and explainabilty in memory update processes offer open research paths towards more transparent and scalable systems. Advancements in these areas will be critical to advancing continual learning systems and their ability to adapt and be robust in performing in myriad applications, including healthcare, customer service, autonomous systems and beyond.

References:

Liu, An-an, Haochun Lu, Heyu Zhou, Tianbao Li, and Mohan S. Kankanhalli. "Balanced Class-Incremental 3D Object Classification and Retrieval." IEEE Transactions on Knowledge and Data Engineering 36 (2024): 35-48.

Zhu, Jun, Yuanfan Wang, Cheng-Geng Huang, Changqing Shen, and Bojian Chen. "A New Incremental Learning for Bearing Fault Diagnosis Under Noisy Conditions Using Classification and Feature-Level Information." IEEE Transactions on Instrumentation and Measurement 73 (2024): 1-14.

Yan, Jingsong, Piji Li, Haibin Chen, Junhao Zheng, and Qianli Ma. "Does the Order Matter? A Random Generative Way to Learn Label Hierarchy for Hierarchical Text Classification." IEEE/ACM Transactions on Audio, Speech, and Language Processing 32 (2024): 276-285.

Kong, Jun, Jin Wang, and Xuejie Zhang. "Adaptive Ensemble Self-Distillation With Consistent Gradients for Fast Inference of Pretrained Language Models." IEEE/ACM Transactions on Audio, Speech, and Language Processing 32 (2024): 430-442.

Qorich, Mohammed, and Rajae El Ouazzani. "Optimizer Algorithms and Convolutional Neural Networks for Text Classification." IAES International Journal of Artificial Intelligence (IJ-AI) 13 (2024): 451-458.

Luo, Yun, Xiaotian Lin, Zhen Yang, Fandong Meng, Jie Zhou, and Yue Zhang. "Mitigating Catastrophic Forgetting in Task-Incremental Continual Learning with Adaptive Classification Criterion." ArXiv abs/2305.12270 (2023).

Song, Yifan, Peiyi Wang, Weimin Xiong, Dawei Zhu, Tianyu Liu, Zhifang Sui, and Sujian Li. "InfoCL: Alleviating Catastrophic Forgetting in Continual Text Classification from An Information Theoretic Perspective." ArXiv abs/2310.06362 (2023)..

Zhang, Duzhen, Wei Cong, Jiahua Dong, Yahan Yu, Xiuyi Chen, Yonggang Zhang, and Zhen Fang. "Continual Named Entity Recognition without Catastrophic Forgetting." ArXiv abs/2310.14541 (2023).

Li, Xingyu, Bo Tang, and Haifeng Li. "AdaER: An Adaptive Experience Replay Approach for Continual Lifelong Learning." ArXiv abs/2308.03810 (2023).

Wang, Jue, Dajie Dong, L. Shou, Ke Chen, and Gang Chen. "Effective Continual Learning for Text Classification with Lightweight Snapshots." AAAI Conference on Artificial Intelligence (2023): 10122-10130.

Prabhu, Ameya, Zhipeng Cai, P. Dokania, Philip H. S. Torr, V. Koltun, and Ozan Sener. "Online Continual Learning Without the Storage Constraint." ArXiv abs/2305.09253 (2023).

Harun, Md Yousuf, Jhair Gallardo, Tyler L. Hayes, and Christopher Kanan. "How Efficient Are Today's Continual Learning Algorithms?" 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2023): 2431-2436.

Winata, Genta Indra, Lingjue Xie, Karthik Radhakrishnan, Shijie Wu, Xisen Jin, Pengxiang Cheng, Mayank Kulkarni, and Daniel Preotiuc-Pietro. "Overcoming Catastrophic Forgetting in Massively Multilingual Continual Learning." ArXiv abs/2305.16252 (2023).

Wang, Liyuan, Xingxing Zhang, Qian Li, Mingtian Zhang, Hang Su, Jun Zhu, and Yi Zhong. "Incorporating Neuro-Inspired Adaptability for Continual Learning in Artificial Intelligence." ArXiv abs/2308.14991 (2023).

Liang, Yanyan, and Wu-Jun Li. "Adaptive Plasticity Improvement for Continual Learning." 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023): 7816-7825.

Zhang, Jie, Chen Chen, Weiming Zhuang, and LingJuan Lv. "Addressing Catastrophic Forgetting in Federated Class-Continual Learning." ArXiv abs/2303.06937 (2023).

Wang, Tiancheng, Huaping Liu, Di Guo, and Xi-Ming Sun. "Continual Residual Reservoir Computing for Remaining Useful Life Prediction." IEEE Transactions on Industrial Informatics 20 (2024): 931-940.

Ma, Bingtao, Yang Cong, and Jiahua Dong. "Topology-Aware Graph Convolution Network for Few-Shot Incremental 3-D Object Learning." IEEE Transactions on Systems, Man, and Cybernetics: Systems 54 (2024): 324-337.

Harun, Md Yousuf, Jhair Gallardo, Tyler L. Hayes, and Christopher Kanan. "How Efficient Are Today's Continual Learning Algorithms?" 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2023): 2431-2436.

Jin, Xisen, Arka Sadhu, Junyi Du, and Xiang Ren. "Gradient Based Memory Editing for Task-Free Continual Learning." ArXiv (2020).

Yao, Xuanrong, Xin Wang, Yue Liu, and Wenwu Zhu. "Continual Recognition with Adaptive Memory Update." ACM Transactions on Multimedia Computing, Communications, and Applications (2022).

Javed, Khurram, and Martha White. "Meta-Learning Representations for Continual Learning." ArXiv (2019).

Gao, Yuyang, G. Ascoli, and Liang Zhao. "Schematic Memory Persistence and Transience for Efficient and Robust Continual Learning." Neural Networks 144 (2021): 49-60.

Chen, Xi, Christos Papadimitriou, and Binghui Peng. "Memory Bounds for Continual Learning." 2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS) (2022): 519-530.

Wang, Liyuan, Xingxing Zhang, Qian Li, et al. "Incorporating Neuro-Inspired Adaptability for Continual Learning in Artificial Intelligence." ArXiv (2023).

Wang, Zhen, Liu Liu, Yiqun Duan, et al. "Continual Learning with Lifelong Vision Transformer." 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022): 171-181.

Shi, Yujun, Li Yuan, Yunpeng Chen, and Jiashi Feng. "Continual Learning via Bit-Level Information Preserving." 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021): 16669-16678.

Graffieti, G., G. Borghi, and D. Maltoni. "Continual Learning in Real-Life Applications." IEEE Robotics and Automation Letters 7 (2022): 6195-6202.