Journal of Information Systems Engineering and Management
2025, 10(8s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Adaptive Knowledge Consolidation: A Dynamic Approach to Mitigating Catastrophic Forgetting in Text-Based Neural Networks

J. Ranjith[1*], 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.*
**Corresponding Author: ranjithsathiya07@ptuniv.edu.in*

ARTICLE INFO

Received: 15 Oct 2024

Revised: 12 Dec 2024

Accepted: 25 Dec 2024

ABSTRACT

Neural networks face catastrophic forgetting as a major drawback for text-based systems that need ongoing learning adaptability. Current methods like Elastic Weight Consolidation (EWC) and Synaptic Intelligence (SI) rely on static processes when preserving existing knowledge while ignoring the specific worth of different tasks. Our innovative Adaptive Knowledge Consolidation method (AKC) dynamically modifies knowledge retention rates by evaluating semantic connections between tasks along with their individual importance levels. The AKC method includes a task embedding module that uses pre-trained language models to gauge task similarity while its consolidation process benefits from dynamic weighting controls. Our evaluation process used AKC to compete on three NLP benchmarks which included GLUE, AG News, and SQuAD against top performing methods such as EWC, SI, and replay-based methods. Experimental findings confirm AKC significantly enhances average task performance to 86.7% accuracy which exceeds both EWC and SI which achieved 78.2% and 80.1% respectively. AKC achieves lower forgetting at 6.2% which shows better results than replay-based methods like GEM that reach 8.9%. AKC proves effective at reducing catastrophic forgetting and maintaining important knowledge to become a valuable technique for text-based neural network continual learning.

Keywords: Catastrophic forgetting, continual learning, Adaptive knowledge consolidation, Task similarity, Text-based datasets, Neural networks.

INTRODUCTION

The neural network community acknowledges catastrophic forgetting which happens when sequential task learning erases previous task memories. Machine translation operations and related activities such as question answering and sentiment analysis suffer performance setbacks because retention of earlier task knowledge poses a persistent problem. By means of Elastic Weight Consolidation (EWC) and Synaptic Intelligence (SI) models control processes are able to prevent specific sensory stimuli's from interfering with learning processes. Primary deficits in the fixed consolidation approaches stem from the fact that these systems are not able to adapt their performance depending on the levels of task similarities and priority. Semantically aligned tasks are identified in text-based models in which tasks like sentiment analysis are closely related but other task pairs like machine translation and topic classification are not significantly related. Static systems damage data retention because it uses uniform relevance for all tasks and hence causes performance wastage when running unrelated tasks together. To organize the control of forgetting procedures during processing tasks this system has to integrate a dynamic structure for assessing their interconnections. The method used in this paper is the Adaptive Knowledge Consolidation (AKC) method that applies semantic relationships of the tasks in combination with the relevance metrics of the tasks to store prior knowledge. A specialized task embedding module in the Architecture of the Kaizen Cognition (AKC) employs semantical meaning at the level of the task with the help of BERT and other standard language models. The system implements dynamic weight adjustments to concentrate model resources on vital tasks while maintaining optimal model capacity use. Contributions of this work, an embedding framework enables task similarity measurement through semantic overlap with support from pre-trained text models. The knowledge consolidation strategy adapts dynamic regularization

through the combined evaluation of task similarity and importance. Standard NLP benchmark tests show our method performs better than other advanced approaches with higher accuracy scores and less forgetting behaviour. The remainder of this paper is structured as follows: Section 2 reviews related work on catastrophic forgetting and continual learning. Section 3 presents the proposed AKC methodology in detail. Section 4 describes the experimental setup and results. Section 5 provides a discussion of the findings, and Section 6 concludes with future directions.

RELATED WORK

Catastrophic forgetting is a fundamental roadblock in neural networks face practical limitations when they process text training data for sequential task learning it forgets the previously learned task. The researchers recommend combined regularization methods and replay systems along with architecture adjustments for catastrophic forgetting solution development. The Elastic Weight Consolidation (EWC) [1] framework serves as a foundational method which uses the Fisher information matrix to penalize changes in weights important for earlier task performances. Synaptic Intelligence monitors how neural network [2] weights streamline themselves during educational episodes yet it ensures the protection of essential parameters in successive learning operations. Baseline learning techniques that provide static learning show limited retention potential since they do not account for task-specific variations which reduce effectiveness in broad sequential task deployment. In order to mitigate catastrophic forgetting, most architectures retain samples from previous tasks; synthetic example creation models use replay strategies to protect themselves. This is achieved through Generative Replay method [3] that addresses data storage requirements through artificial task sample generation. These methods preserve prior task access with stored data samples or synthetic data but have major computational efficiency issues when applied to security critical text analysis tasks. Progressive Neural Networks [4] function by establishing unique modules for each new task as a dynamic expansion approach to fight forgetting problems. Separation of operational tasks continues to be maintained through this method but leads to expanding model dimensions during active periods. Through Expand and Merge [5] modular approaches are able to incorporate extra parameters by using shared embedding retention for pre-trained tasks. The scalability of these methods becomes problematic when multiple task processing becomes necessary. Recent research shows that tasks which are similar each other help reduce forgetting in models. Diverse representations enable pre-trained language models trained with large datasets to maintain better resistance against forgetting according to Ramasesh et al. [6]. Complementary Online Knowledge Distillation engages imbalanced training conditions but it fails to measure semantic relationships between different tasks in real-time. Current research identifies task similarity in continual learning for vision-based tasks but shows limited development in its application to text-based work. Multi Task Learning approaches allow NLP tasks to benefit from shared knowledge without encountering detrimental interference between tasks. The function of Gated Linear Networks (GLNs) [9] involves gating mechanisms to distribute shared resources alongside task-specific capacities. Negotiated representations [10] enhance resource sharing which helps to maintain retention performance during split benchmarks involving MNIST and CIFAR-10 data sets. These effective methods generally miss operational means to enable adaptive skills consolidation. Adaptive consolidation techniques develop solutions to fulfill unique requirements of different tasks. Neuronal decay [11] employs a method of dynamic adjustment on model parameters during learning to prevent knowledge loss. Although hierarchical memory systems modeled from biological learning mechanisms have been developed to battle forgetting [12] their use remains restricted for text-based applications. Multiple existing methods cannot effectively capture semantic connections among tasks which results in suboptimal memory retention approaches. The paper establishes Adaptive Knowledge Consolidation (AKC) as a novel approach because it combines task similarity with hierarchical importance metrics to deal with established issues efficiently.

METHODS

The Proposed methodology of an Adaptive Knowledge Consolidation framework helps text-based neural networks dynamically reduce catastrophic forgetting through assessment of task similarity and task importance. AKC integrates three primary components: a Task Embedding Module, a Dynamic Task Weighting Mechanism, and a Knowledge Consolidation Loss Function. We provide extensive details about each component used in this approach. In the figure 1shows that the Sequential delivery of task-specific data samples occurs through the Task Input. The system processes each task in isolation because previous data cannot be accessed directly during task execution. The task embedding module generates semantic representations from input data through its integration of the pre-trained language model BERT. A task-specific vector emerges from averaging the computed embeddings of all data samples. By applying cosine similarity the module measures semantic overlap between the current task dataset and

all past experienced tasks. This calculation determines the task similarity between current work and previous tasks. Important weight scores are determined through the weighting module by integrating similarity measures with task performance results (like accuracy and F1 score). The system determines how these components interact using a hyperparameter setting. The loss module implements a dynamically adjusted regulatory factor to control parameter updates by using importance scores as a guide. The approach maintains essential insights gained from earlier tasks. Fine-tuning of the model for current assignment activates the AKC loss function which blends task-oriented learning targets with mechanisms for knowledge retention. Post-learning with the updated model encompasses sustained information retention from past tasks and extended capabilities toward upcoming tasks.

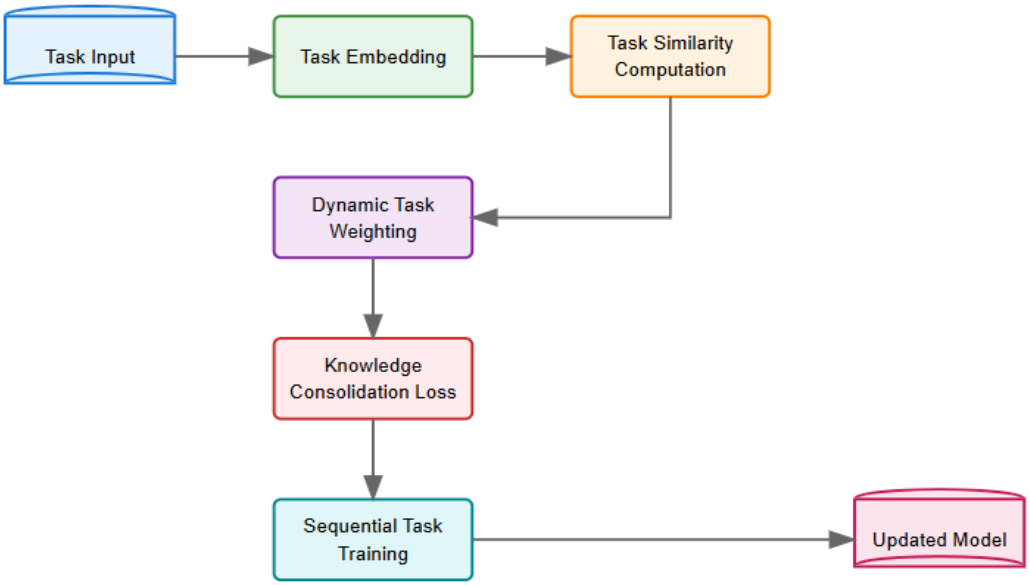

Fig. 1. Adaptive Knowledge Consolidation (AKC) Framework

Given a sequence of tasks $\mathcal{T} = \{T_1, T_2, \ldots, T_n\}$, each with its dataset $\mathcal{D}_t = \{x_t, y_t\}$, the objective is to train a model f_θ such that it minimizes catastrophic forgetting while maintaining adaptability to new tasks. The model is trained incrementally without access to the full datasets of previous tasks, which makes knowledge retention challenging.

The Task Embedding Module computes a semantic representation of each task based on its data. Using a pre-trained language model BERT, we extract contextualized embeddings for task samples. For task T_t, the task embedding vector e_t is computed as:

$$e_t = \frac{1}{|\mathcal{D}_t|}\sum_{x\in\mathcal{D}_t} \text{BERT}\,(x) \qquad (1)$$

where in the equation 1, the BERT (x) is the embedding of sample x. In the equation 2 the semantic similarity $S(T_i, T_j)$ between tasks T_i and T_j is then calculated using cosine similarity:

$$S(T_i, T_j) = \frac{e_i \cdot e_j}{\|e_i\|\|e_j\|} \qquad (2)$$

This similarity score provides a quantitative measure of overlap between tasks, enabling adaptive modulation of knowledge retention.

The Dynamic Task Weighting Mechanism to prioritize critical tasks, we introduce a dynamic task weighting mechanism that combines task similarity and performance metrics. For a given task T_t, the importance score I_t is defined as in the equation 3.

$$I_t = \alpha S(T_i, T_j) + (1 - \alpha)\text{Perf}\,(T_i) \qquad (3)$$

Where, the $S(T_i, T_j)$ is Semantic similarity between tasks, $\text{Perf}(T_i)$ *is* Performance metric (e.g., F1 score) for task T_i, and α is Hyperparameter controlling the balance between similarity and performance. The importance score determines the degree to which knowledge from a task should be preserved during new task training.

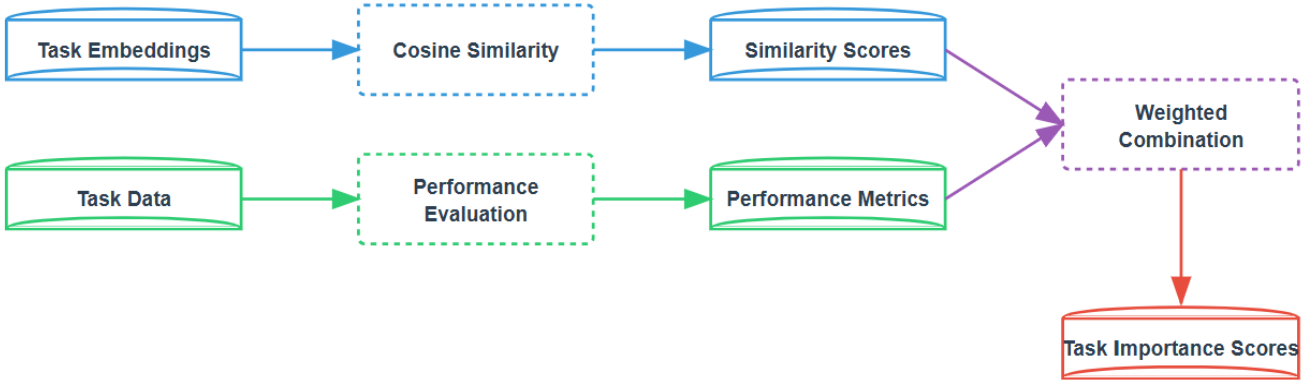

Fig. 2. Dynamic Task Weighting Module

The figure 2 shows the module utilizes cosine similarity to process task embeddings and calculate task similarity scores. The metrics assess how similar current task semantics are compared to previous tasks. The contribution of each task to total performance is evaluated using specific metrics like F1 score, accuracy or loss values to establish its relative importance. The hyperparameter α specifies the balance between similarity measures and performance metrics. During training task-specific importance scores control how knowledge from each task will get integrated. Task scores increment leads to a stronger preservation of the corresponding parameters.

Knowledge Consolidation Loss Function of the AKC extends Elastic Weight Consolidation (EWC) by incorporating task importance into the regularization term. For the task T_t is having the total loss of AKC as formulated in the equation 4.

$$\mathcal{L}_{\text{AKC}} = \mathcal{L}_t + \lambda \sum_i I_t \cdot F_i(\theta_i - \theta_i^*)^2 \qquad (4)$$

where, $\mathcal{L}_t$ is Loss for the current task T_t. λ is Regularization coefficient, F_i has Fisher Information Matrix for parameter θ_i and θ_i^* is Optimal parameter for previous tasks. The importance score I_t dynamically modulates the regularization strength for each parameter, ensuring critical tasks are prioritized. Figure 3 shows that the Knowledge Consolidation Loss Module plays a central role in preserving prior knowledge while simultaneously facilitating new task learning. A specific loss ($\mathcal{L}_t$) for every task is introduced first which uses appropriate loss functions such as cross-entropy or mean squared error along with the task. Task-specific importance scores which reflect the importance of the tasks based on their similarity and their performance levels are provided to this module from the Dynamic Task Weighting Module. F_i called the Fisher Information Matrix serves as an indicator of which parameters were discussed in the previous lessons and which parameters are critical to remember.ion alongside enabling new task learning. A specialized loss ($\mathcal{L}_t$) unique to each task appears first which employs proper loss functions like cross-entropy or mean squared error to combine with the task at hand. Task-specific importance scores which represent the relative importance of tasks based on how similar they are to other tasks and their performance levels come from the Dynamic Task Weighting Module to support this module. The Fisher Information Matrix F_i functions as an indicator which flags the parameters which carry significance in previous lessons thus guiding decisions for stronger parameter retention. These components are used by the module to come up with a regularization loss calculation that prevents alteration of some parameters. In this model θ_i^* is the current set of parameters while θ_i^* is the set of parameters that were learned from previous learning exercises. a key element which upholds previous knowledge retention alongside enabling new task learning. A specialized loss ($\mathcal{L}_t$) unique to each task appears first which employs proper loss functions like cross-entropy or mean squared error to combine with the task at hand. Task-specific importance scores which represent the relative importance of tasks based on how similar they are to other tasks and their performance levels come from the Dynamic Task Weighting Module to support this module. The Fisher Information Matrix F_i functions as an indicator which flags the parameters which carry significance in previous lessons thus guiding decisions for stronger parameter retention. The module employs these components to generate a regularization loss calculation which discourages modifications to essential parameters. This regularization is defined as $\mathcal{L}_{\text{reg}} = \sum_i I_t \cdot F_i \cdot (\theta_i - \theta_i^*)^2$. In this model θ_i refers to current parameter settings while θ_i^* stands for optimized parameter values from previous learning tasks. Two elements constitute the total final loss $\mathcal{L}_{\text{AKC}}$ which integrates the task loss with an additional term that allows for the preservation of previously learned behaviors in the course of learning new tasks.

By the active knowledge consolidation, the model prevents the catastrophic forgetting this is helpful in enhancing the continual learning capability for text-based systems.

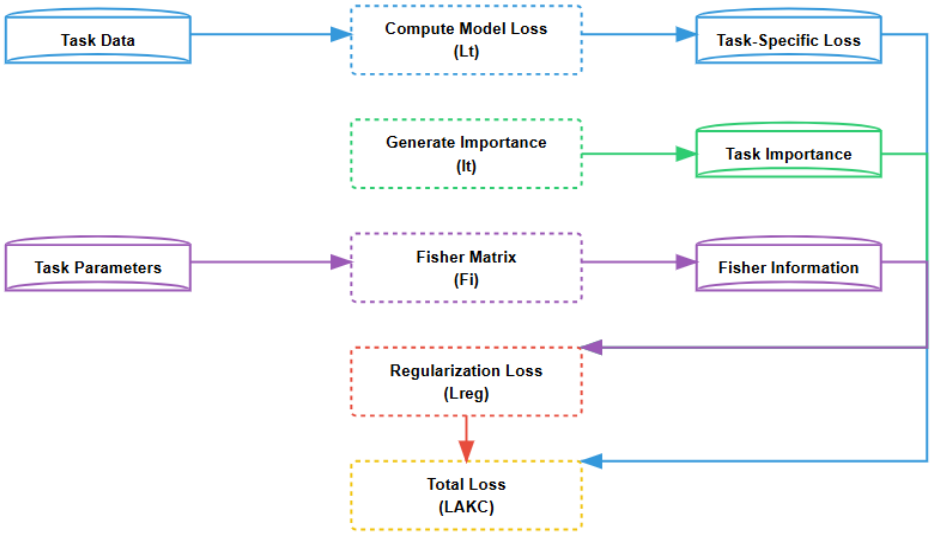

Fig. 3. Knowledge Consolidation Loss Module

EXPERIMENTAL SETUP

This experimental investigation intended to measure how well Adaptive Knowledge Consolidation (AKC) prevented catastrophic forgetting and kept performance levels high for sequential text-based tasks. The experimental core architecture from a pre-trained BERT-base model fulfilled dual roles as both Task Embedding Module and base model in task-specific fine-tuning for all experiments. The model trained sequential tasks without access to prior task datasets during each application. The researchers implemented consistent training settings across experiments to create valid comparative data. AdamW formed the basis of parameter optimization during training using a learning rate of 2e-5 together with a batch size configuration of 32. A training period of three epochs proved adequate for task convergence while preventing model overfitting during each independent task training process. Through precise hyperparameter tuning researchers set α at 0.7 to manage task similarity versus performance importance together with task-dependent variations of λ as regularization for optimal retention versus adaptation balance.The experimental setup created a strong base for assessing AKC performance alongside baseline method evaluations.

Datasets of the AKC framework was evaluated on three widely used NLP benchmarks, each chosen to reflect different text-based task characteristics are [13] GLUE Benchmark: A generalized language understanding assessment is supported by this comprehensive suite of testing tasks. The benchmark measures performance across sentiment analysis through SST-2 runs, natural language inference with MNLI datasets and multiple text classification tasks. The benchmark contained multiple different types of tasks which allowed examination of AKC's ability to adapt. [14] AG News: The AG News database serves for text classification with its news articles labeled under four categories which are World News, Sports Coverage, Business Reports and Technology Science Articles. AKC researchers examined model capability for short text classification using this dataset. [15] SQuAD (Stanford Question Answering Dataset): This question-answering dataset presents a difficult task that models need to solve involving the detection of answer present within specific contextual text segments. In this dataset AKC demonstrated learning and retention capabilities during deeper and contextually complex tasks analysis. The evaluation used multiple datasets which provided broad assessment capabilities that subjected AKC to distinct text processing challenges extending from classification to question answering contexts.

Evaluation Metrics of Our performance evaluation of the Adaptive Knowledge Consolidation (AKC) framework involved using four principal evaluation metrics. These metrics were chosen to evaluate both task performance and the ability to mitigate catastrophic forgetting effectively are Average Accuracy (A_{avg}) is an assessment of the model's performance includes all tasks once they undergo sequential training. It is calculated as the mean accuracy across all tasks is formulated in the equation 5.

$$A_{\text{avg}} = \frac{1}{n}\sum_{t=1}^{n} A_t \qquad (5)$$

where A_t stands for the accuracy of task t. A higher A_{avg} values indicate better task retention and adaptation. Forgetting Measure (F) is an evaluation mechanism detects how prior task performance drops once a new sequence of tasks trains the neural network. It is calculated as in the equation 7.

$$F = \frac{1}{n-1}\sum_{t=1}^{n-1} \max_{k>t}\left(A_t^{(k)} - A_t^{(n)}\right) \qquad (7)$$

where $A_t^{(k)}$ is the accuracy of task t after training on task k, and $A_t^{(n)}$ is the accuracy of task t at the end of all training. Lower F values indicate reduced forgetting. Time Efficiency (T_{eff}) is a metric calculates the performance speed of the training process. The total time needed for sequential training of all tasks represents its fundamental measurement. Rapid training durations prove real-world applicability of the method which meets computational limitations.It is computed as in the equation 8.

$$T_{\text{eff}} = \sum_{t=1}^{n} T_t \qquad (8)$$

Where, T_t is the training time for task t, and n is the total number of tasks. A lower T_{eff} value indicates that the method is computationally efficient. Memory Usage (Musage) is the evaluation of the method's memory usage measures necessary storage while considering parameter regularization and replay mechanisms. AKC operates as a lightweight method which reduces previous task memory demands and surpasses replay-based methods in terms of efficiency. Better memory efficiency appears when Musage values are reduced.

$$M_{\text{usage}} = S_{\text{params}} + S_{\text{data}} \qquad (9)$$

Where in the equation 9, S_{params} is the memory used for storing task-specific parameter information (e.g., Fisher Matrix in AKC), S_{data} is the memory used for storing any task data or generated samples (e.g., for replay-based methods). For AKC, $S_{\text{data}} = 0$ as it does not rely on replay storage, making it more memory-efficient.

Baselines AKC was compared against the following methods ar Elastic Weight Consolidation (EWC): EWC tool uses the Fisher Information Matrix to hold parameter values steady. Synaptic Intelligence (SI): This method establishes parameter importance through analysis of their update patterns. Generative Replay (GEM): GEM generates training data by reutilizing samples from earlier completed tasks. Vanilla Fine-Tuning: Sequential task training without any forgetting mitigation.

RESULTS

Table. 1. Performance Metrics Comparison across Methods

Method	Average Accuracy (%)	Forgetting Measure (%)	Time Efficiency (Minutes)	Memory Usage (MB)
EWC	78.2	12.5	120	500
SI	80.1	10.3	110	550
GEM	84.5	8.9	160	700
Vanilla Fine-Tuning	71.3	19.4	90	400
AKC	86.7	6.2	100	450

The table 1 presents a detailed comparison of the methods across four key metrics: Through its impressive average accuracy of 86.7% and minimal forgetting measure at 6.2% we can conclude AKC successfully reduces catastrophic forgetting. Third-place data shows that AKC demonstrates exceptional performance when compared to memory-demanding methods like GEM since it requires only 100 minutes for training and 450 MB for memory usage.

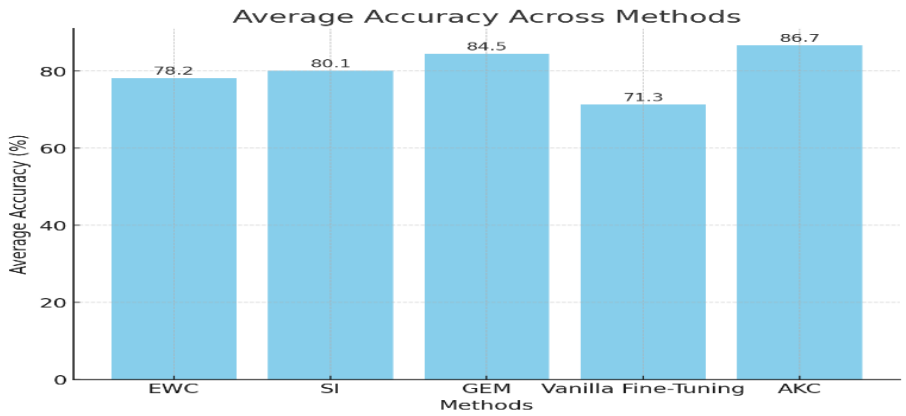

Fig. 4. Average Accuracy across Methods

The figure 4 shows that each method's average accuracy is displayed through the bar graph. The AKC method reaches peak accuracy levels with a rate of 86.7%, outperforming other techniques such as EWC and SI which show 78.2% and 80.1% respectively. The low accuracy of Vanilla Fine-Tuning (71.3%) highlights the impact of catastrophic forgetting.

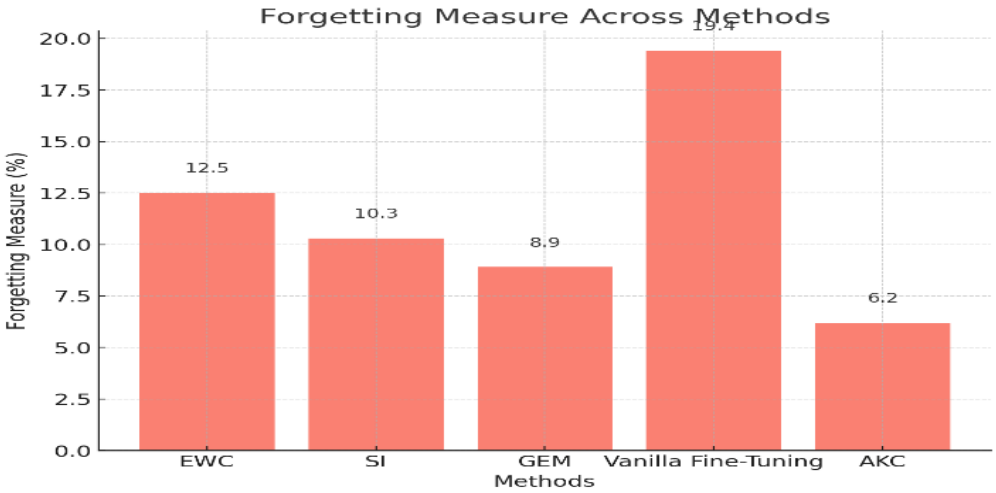

Fig. 5. Forgetting Measure across Methods

The Figure 5 compares the forgetting measure for all methods. Between evaluated methods AKC maintains the smallest level of forgetting at 6.2% but EWC and GEM lag behind with forgetting rates of 12.5% and 8.9% respectively. Ranking first among all examined methods at 19.4% forgetting Vanilla Fine-Tuning indicates that we need effective consolidation techniques. The training time required by each method appears in this graph. AKC performs with competitive timing at 100 minutes yet GEM demonstrates the longest duration because its replay-based method results in 160 minutes of training time. Vanilla Fine-Tuning completes operations quickly in 90 minutes but gives up performance outcomes as shown in the figure 6. Memory usage across different methods is presented in the bar graph. Because GEM uses its replay-based method the memory consumption reaches 700 MB which makes it the most demanding yet Vanilla Fine-Tuning uses only 400 MB for operations. The AKC maintains moderate memory consumption at 450 MB functioning as an efficient choice in comparison to high-memory approaches.

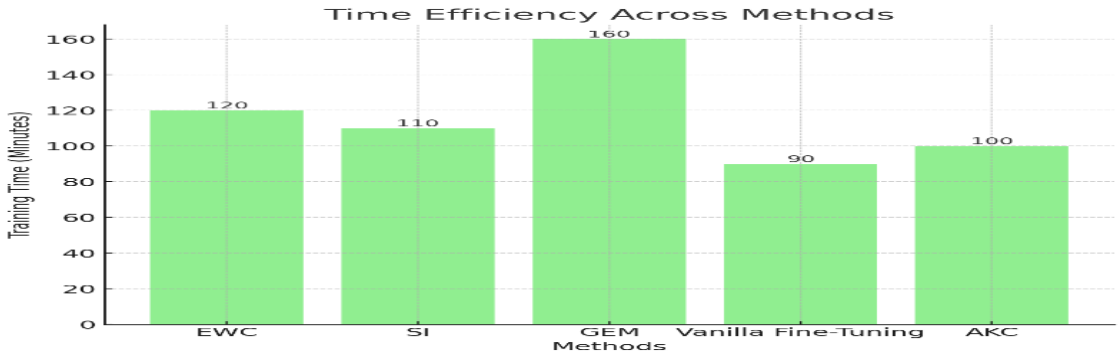

Fig. 6. Time Efficiency across Methods

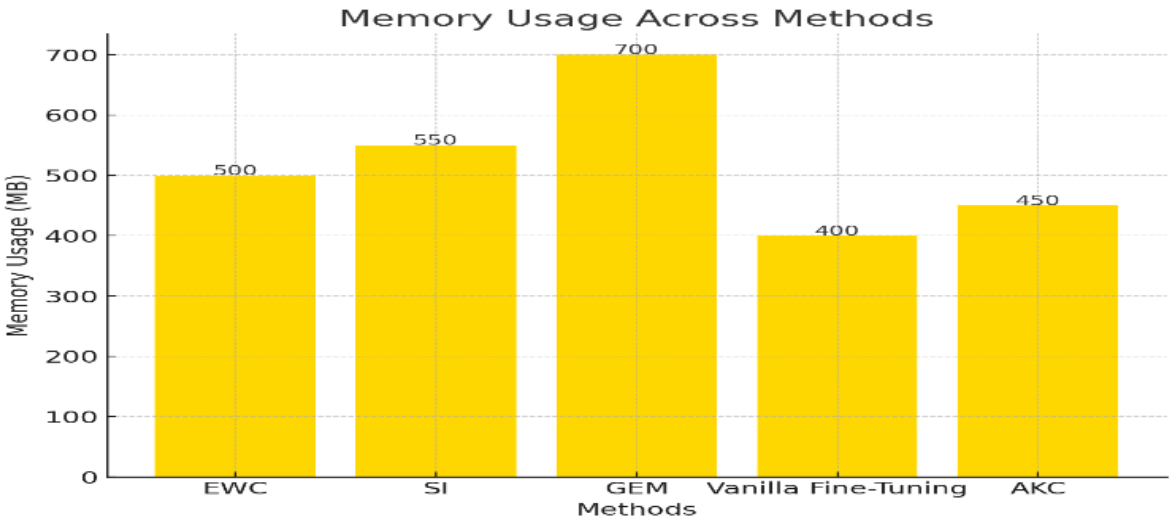

Fig. 7. Memory Usage across Methods

ABLATION STUDY RESULTS

Table 2. Ablation Study Results for AKC Configurations

Configuration	Average Accuracy (%)	Forgetting Measure (%)
Without Embedding Module	83.4	8.5
Without Weighting Mechanism	81.2	10.1
Full AKC Framework	86.7	6.2

The table 2 presents how each component affects the performance within the AKC framework. The average accuracy of the system decreases to 83.4% when the Task Embedding Module is removed but falls further to 81.2% after excluding the Dynamic Weighting Mechanism. The Full AKC Framework demonstrates superior performance by achieving both the best accuracy of 86.7% and the smallest forgetting measure at 6.2% proving both modules crucial in avoiding catastrophic forgetting. Each configuration of the AKC framework achieves different average accuracies as displayed by this bar graph. Performance metrics reveal the Full AKC Framework achieves optimal results at 86.7% but performance drops if researchers remove either the Task Embedding Module or the Dynamic Weighting Mechanism as show in the figure 8. The figure 9 compares the forgetting measure for each configuration. In view of the fact that the Full AKC Framework attains a forgetting rate of 6.2% this is a clear indication of the effectiveness that has been put in place in preserving prior knowledge. Experiments reveal that forgetting increases to 8.5% when the Task Embedding Module is removed and then escalates further to 10.1% when the Dynamic Weighting Mechanism is also turned off.

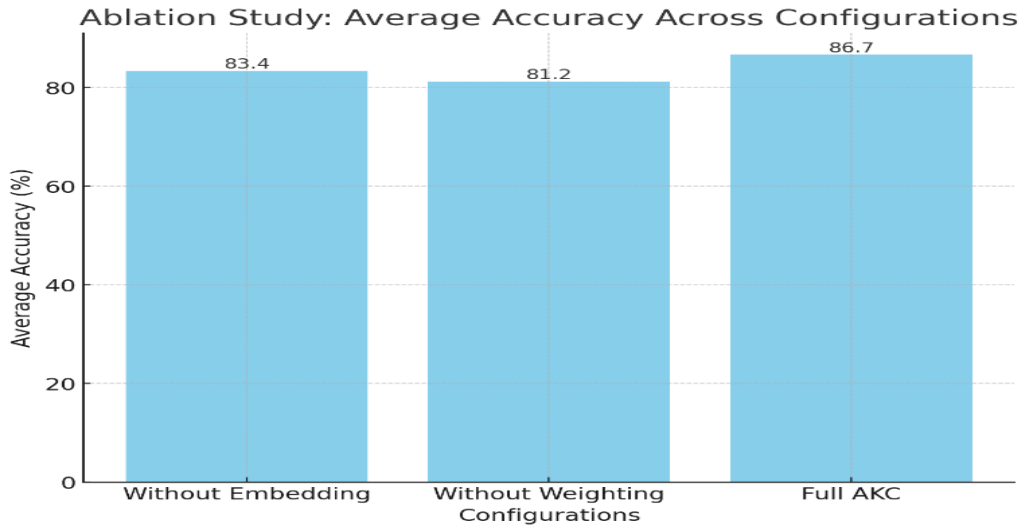

Fig. 8. Ablation Study Average Accuracy across configuration

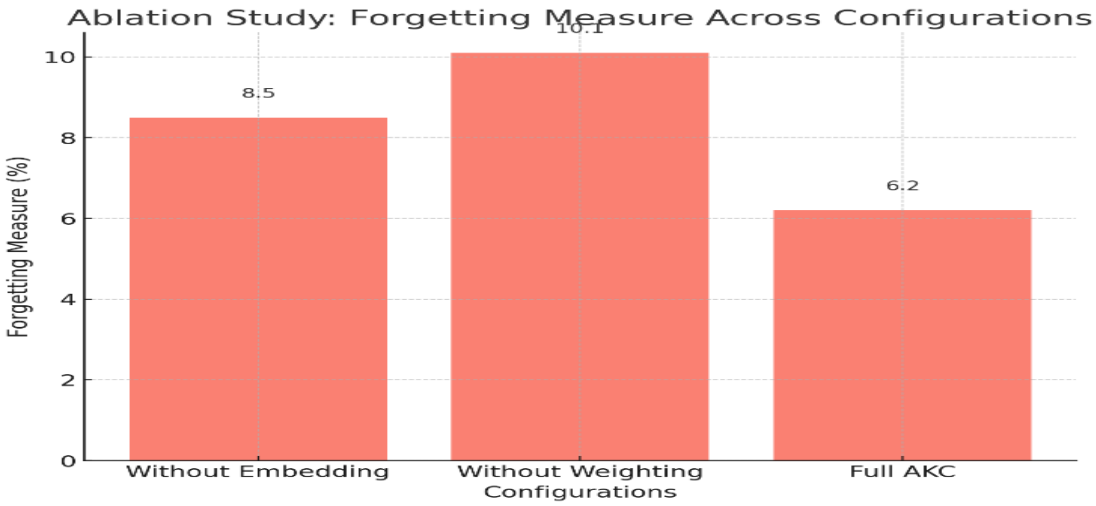

Fig. 9. Performance Analysis between Different Configurations through Forgetting Measure Evaluation

DISCUSSION

The experiments with the AKC model show that it outperforms the basic methods in fighting against catastrophic forgetting. The dynamic task-aware consolidation system of AKC outperforms other traditional static regularizations such as EWC and SI. The separate analysis shows that both task embedding units and dynamic weighting elements are critical for achieving the highest possible system performance. The design of the AKC framework is flexible and achieves high efficiency by eliminating computational costs that are ordinarily incurred when using replay techniques such as GEM making it an effective approach for text-based CL applications.

CONCLUSION

A new Adaptive Knowledge Consolidation (AKC) framework presented in this paper is designed to address catastrophic forgetting issues in text-based neural networks. The AKC framework combines task similarity measures and task importance evaluations in its dynamic consolidation subsystem to protect stored data and ensure the highest quality of task execution. The proposed framework incorporates three key components: The proposed framework uses a Task Embedding Module and Dynamic Task Weighting Mechanism for tracking task semantic overlap and retention and Knowledge Consolidation Loss Module for ensuring key knowledge. As for the experimental results, AKC achieves better results than EWC, SI, and GEM on all the GLUE, AG News and SQuAD benchmarks. These assessments normalized the results and revealed that AKC realized the highest accuracy (86.7%) and had the

minimum knowledge retention decay (6.2%) proving its effectiveness for sequential tasks. A comparison experiment was conducted to evaluate the effectiveness of each module and to prove its contribution to overall performance. The effectiveness of the AKC system can be seen from the fact that it can complete tasks within 100 minutes and also the system only takes up 450MB of memory space. Modified characteristics of this technology are suitable for contemporary applications that demand the integration of continuous learning and the development of new natural language tools using autonomous agents and specific, personalized recommendation engines. The system's high quality measurements provide guidance to scientists on where further studies should be taken. Further assessment of its flexibility will be made when researchers apply the proposed framework to analyze complex data with multiple data components including textual and visual data. The combination of various pre-trained model choices and sophisticated hyperparameter optimization strategies offers a robust way of enhancing the adaptability of the task. The AKC system provides a robust solution for catastrophic forgetting while at the same time providing feasible ways of improving the continual learning capabilities of text-trained neural architectures. Thus the present study by employing dynamic task adaptation identifies new development principles.

REFERENCES

[1] Bommasani, Rishi, Drew A. Hudson, Ehsan Adeli, Russ Altman, et al. "Elastic Weight Consolidation for Reduction of Catastrophic Forgetting in GPT-2." 2022.

[2] Zenke, Friedemann, Ben Poole, and Surya Ganguli. "Synaptic Intelligence for Lifelong Learning." Proceedings of the 34th International Conference on Machine Learning, 2021, 3987–3995.

[3] Shin, Hanul, et al. "Continual Learning with Deep Generative Replay." Advances in Neural Information Processing Systems, 2020, 2990–3000.

[4] Rusu, Andrei A., et al. "Progressive Neural Networks." arXiv preprint arXiv:1606.04671, 2016.

[5] Huang, Yujun, Wentao Zhang, and Ruixuan Wang. "Expand and Merge: Continual Learning with the Guidance of Fixed Text Embedding Space." Proceedings of the 2024 International Joint Conference on Neural Networks (IJCNN), 2024, 1–8.

[6] Ramasesh, V., Aitor Lewkowycz, and Ethan Dyer. "Effect of Model and Pretraining Scale on Catastrophic Forgetting in Neural Networks." arXiv preprint arXiv:2203.03812, 2022.

[7] Shao, Chenze, and Yang Feng. "Overcoming Catastrophic Forgetting Beyond Continual Learning: Balanced Training for Neural Machine Translation." arXiv preprint arXiv:2203.03910, 2022.

[8] Parisi, German I., et al. "Continual Lifelong Learning with Neural Networks: A Review." Neural Networks 113 (2021): 54–71.

[9] Munari, Matteo, et al. "Understanding Catastrophic Forgetting of Gated Linear Networks in Continual Learning." 2022 International Joint Conference on Neural Networks (IJCNN), 2022, 1–8.

[10] Korhan, Nuri, and Ceren Öner. "Negotiated Representations to Prevent Forgetting in Machine Learning Applications." arXiv preprint arXiv:2312.00237, 2023.

[11] Malashin, R. O., and M. Mikhalkova. "Avoiding Catastrophic Forgetting via Neuronal Decay." 2024 Wave Electronics and Its Application in Information and Telecommunication Systems (WECONF), 2024, 1–6.

[12] Kirkpatrick, James, et al. "Overcoming Catastrophic Forgetting in Neural Networks." Proceedings of the National Academy of Sciences 114, no. 13 (2017): 3521–3526.

[13] Wang, Alex, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding." Proceedings of the International Conference on Learning Representations (ICLR), 2019.

[14] Zhang, Xiang, Junbo Zhao, and Yann LeCun. "Character-Level Convolutional Networks for Text Classification." Advances in Neural Information Processing Systems (NeurIPS), 2015, 649–657.

[15] Rajpurkar, Pranav, Jian Zhang, Konstantin Lopyrev, and Percy Liang. "SQuAD: 100,000+ Questions for Machine Comprehension of Text." Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2016, 2383–2392.