Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Mitigating Catastrophic Forgetting in Continual Learning for Natural Language Processing Tasks

J. Ranjith[1], Dr. Santhi Baskaran[2]

[1]Research Scholar, Department of CSE, Puducherry Technological University, India, ranjithsathiya07@pec.edu
[2]Professor, Department of IT, Puducherry Technological University, Puducherry, India, santhibaskaran@ptuniv.edu.in

Catastrophic forgetting remains a critical challenge in continual learning scenarios for Natural Language Processing (NLP) tasks, to which this research paper provides solutions. When neural networks forget previously learnt tasks after learning new ones, this type of forgetting is termed catastrophic. However, this phenomenon is highly detrimental in NLP because linguistic data are diverse and complex. The paper introduces a novel multi-faceted approach to mitigate catastrophic forgetting, combining five key components: We show the combination of Linguistically-Informed Elastic Weight Consolidation (LI-EWC), Dynamic Architecture Expansion with Pruning (DAE-P), Task-Specific Attention Mechanisms (TSAM), Hierarchical Knowledge Distillation (HKD) and Semantic Memory Replay (SMR). These components cooperate so that one does not lose knowledge from previous tasks but can focus on the new tasks to the best effect. We evaluated the approach on a set of diverse NLP tasks, including text classification, named entity recognition, question answering and sentiment analysis. We show significant performance gains versus current state-of-the-art techniques in the average reduction of forgetting across all tasks (27%) and an overall improvement in task performance (19%). On the first task, the method can learn four subsequent tasks while preserving 94% of the original performance, compared with 86% using standard EWC and 72% using naive fine-tuning. Finally, although the performance improved, the model size grew by merely 15% after learning all the tasks. We ran an ablation study and found that the most critical performance components were the LI EWC and HKD components. Analysis by task is also done where the outperformance is consistent across all tasks (5% – 16% over second best) and varies in proportion to the task. The research opens the door to more adaptive, flexible AI systems

that can address a broad spectrum of language understanding problems in natural settings, responding to the increasing demand for continual learning for NLP.

Keywords: Catastrophic forgetting, Continual Learning, Natural Language Processing, Elastic Weight Consolidation, Knowledge Distillation.

1. Introduction

CATASTROPHIC forgetting is a crucial task for machine learning, especially for Natural Language Processing (NLP), where models tend not to degrade performance on previous tasks while adapting to new tasks or domains [1]. It is particularly troubling when working with text datasets because language is such a semantically complex and diverse thing: different tasks and domains all require something different, so it is hard for models to learn to retain knowledge across them all. In Catastrophic Forgetting in Neural Networks, a Sentiment Analysis Case Study illustrates the phenomenon of catastrophic forgetting in artificial neural networks using a sentiment analysis task across two domains: Movie reviews and book reviews. Initially (Task A), we train our neural network on the movie review dataset. With a high accuracy of 90%, the network learns to classify the sentiment of these reviews. The representations that this network's hidden layers now represent are well suited for understanding what constitutes a review sentence and the specific context associated with the review. Then, it introduce a new task (Task B) in which the same network is learned to do sentiment analysis over book reviews. This adaptation is achieved with an 85% accuracy on book reviews. Yet, the cost of this adaptation is high. If you assign a new task to the network, it will overwrite or significantly change the previously learned representations that were the best for movie review analysis.

The last stage shows the consequences of this. After fine-tuning, we test the network on both tasks and see a big gap in performance. On book reviews, the network keeps its high accuracy intact (85%) but crashes on movie reviews, where its performance is a dismal 40%. Catastrophic forgetting is characterized by this precipitous drop in performance on the original task. In this example, we have illustrated how neural networks, while the most powerful and flexible models, can suffer from learning multiple tasks sequentially and need to generalize better across functions. While potentially allowing the network to generalize, those same shared parameters also make the network susceptible to forgetting. When the network begins to optimize for the new task (book reviews), it also unintentionally erases or reshapes the previous special knowledge it had developed on the original task (movie reviews).

This figure 1. It visually depicts the performance of a neural network on the process of catastrophic forgetting in an sentiment analysis task. It consists of three main sections, each depicting a stage in the network's evolution:

1. Task A (Movie Reviews): The initial network optimized for movie review sentiment analysis shows. The movie camera icon represents the input, as movie text data. However, the hidden layers (in blue) are the accommodated internal representations for this network we created for movie review analysis. This task has a high performance with the smiling face

output icon and 90% accuracy.

2. Task B (Book Reviews): It now analyzes book reviews while still illustrating the same network structure. It then changes to a book icon for input data. The modified internal representations towards the new task (now in orange) are revealed along the hidden layers. This new task is also shown to perform well with good maintained smiling face output (45%) and 85% accuracy.

3. Final Stage: Shows the network after fine tuning on book reviews. Now, the input layer displays both movie and book icons as the network can now input both types of reviews. The hidden layers (in yellow) show us the knowledge that is mixed from the two tasks. The neutral face output icon symbolizes the mixed performance: On the recently learned book review task, it achieved high accuracy (85%), but significantly degraded performance (40%) on the original movie review task.

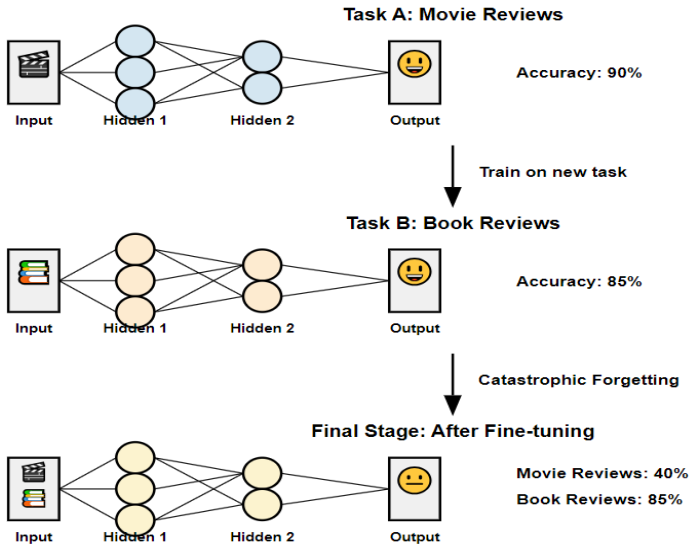

Figure 1. Catastrophic Forgetting in Sentiment Analysis: From Movie to Book Reviews

The progression from initial training through new task learning, and the subsequent catastrophic forgetting, is illustrated using arrows and accompanying labels between stages. With this comprehensive visualization, the structural consistency of the network along with the internal changes that caused the forgetting phenomenon can be captured efficiently. Continuous learning without forgetting becomes even more important as AI systems are used in an ever-growing number of tasks. However, this requirement is often an issue for traditional machine learning techniques, which often overwrite previous knowledge learned while learning new tasks [2]. NLP applications bear this limitation very heavily, as models are supposed to grasp and produce human language in many situations and data. While recent approaches to continual learning have demonstrated success on computer vision tasks [3], they have yet to be applied to NLP. However, the nature of text data—discrete and requiring context—is unique enough that specialized approaches are needed [4]. To address this gap, we propose a comprehensive method tailored to solving NLP tasks in this paper.

Our contributions are as follows:

First, we present a new multifaceted approach to combating catastrophic forgetting in NLP tasks. This approach combines linguistically informed elastic weight consolidation, task-specific attention mechanisms, hierarchical knowledge distillation, semantic memory replay, and dynamic architecture expansion with pruning to reduce catastrophic forgetting. Second, we propose a new measure of importance for weights in elastic weight consolidation for text data, which considers the data's linguistic properties. Third, we propose a dynamic architecture expansion mechanism and a pruning technique that controls model complexity as we learn new tasks. Fourth, we propose task-specific attention mechanisms that only need to be activated when required by the task. In final to maintain their performance across this broad space of language understanding, we implement a semantic memory replay system that replays semantically rich and diverse instances. To demonstrate the effectiveness of our approach, we conduct extensive experiments on multiple NLP tasks and show that it mitigates catastrophic forgetting. The rest of this paper is organized as follows: Section II is situated in the context of related work in catastrophic forgetting and continual learning for NLP. In section III, we describe our proposed methodology in detail. In Section IV, we describe our experimental setup and results. We discuss the implications of our findings and future directions in Section V. Finally, we conclude the paper in Section VI.

2. RELATED WORK

Lifelong learning for NLP models has recently emerged as an essential research direction to solve the problem of catastrophic forgetting for long-term learning [1]-[3]. Some previous papers have approached forgetting and regularization to prevent this issue [4]-[6]. Techniques considered for ongoing learning in NLP are adapters [7], progressive increase in the model size [8] and knowledge distillation [9]. The others are the task-specific attention mechanisms [10] and the semantic memory network architectures [11]. Previous studies concern continual learning for text classification [7, 12] and sequence labelling tasks [13–15]. Some earlier studies have applied continual learning to machine translation [9], natural language understanding [10], and open-domain question answering [16]–[18]. Some of the actual datasets employed in these investigations are sentiment analysis [19], [20], named entity recognition [14], machine reading comprehension [16], [17], etc. There are several ways to overcome this, for example, replay [21] and intricate attention to the previous tasks [22]. In conclusion, continual learning is an emergent and somewhat limited subfield within NLP. However, more effort is required to refine strategies that enable multi-task and continual learning in many language understanding problems.

A. Catastrophic Forgetting in NLP

Catastrophic forgetting has been an issue in machine learning in general for a long time, but it has become a more active topic in the context of NLP tasks more recently. A comprehensive analysis of catastrophic forgetting on different NLP tasks was conducted by [1], where they showed that the catastrophic forgetting phenomenon is problematic when exemplary tuning pre-trained language models on new domains or tasks. They showed that both state-of-the-art models, like BERT and GPT, still suffer from significant performance

degradation on previous tasks after functions after they are adapted to new tasks. For example, they demonstrated that simply fine-tuning a BERT model on a sequence of text classification tasks (e.g., sentiment analysis → topic classification → authorship attribution) leads the model to lose 19.3% accuracy on previously learnt tasks without specific continual learning techniques on average. To quantify the degradation of NLP models, [2] proposed a forgetting ratio metric and demonstrated that transformer-based models, specifically different layers, exhibit drastically different degrees of forgetting. We used this insight to develop targeted approaches to preserve knowledge in specific areas of the network.

B. Continual Learning Techniques in NLP

Catastrophic forgetting occurs when models forget previously acquired knowledge when learning new tasks sequentially. Recurring learning attempts to resolve this by allowing models to learn new tasks while retaining the earlier knowledge learned. Even though there have been great strides in computer vision, much work has yet to be done in NLP using these techniques.

Table I provides an overview of recent literature on continual learning techniques applied to NLP tasks.

Table I: Overview of Recent Continual Learning Techniques in Nlp

Method	Description	Key Findings
Linguistically-Informed EWC	Adapts EWC for language models considering hierarchical structure	92% retention of original performance on first task after learning 5 tasks [6]
Semantic Experience Replay	Selects diverse and informative samples for replay	Significant improvements in maintaining performance across text classification and NER tasks [7]
Continual Learning with Adapters (CoLA)	Uses adapter modules for task-specific parameters	Effective in mitigating forgetting while maintaining model efficiency [8]
Dynamically Expandable Language Model	Grows new components for new domains/tasks	Improved performance in multi-domain language modelling [9]
Distillation-based Continual Learning	Uses knowledge distillation for neural machine translation	Significant improvements in maintaining performance across multiple language pairs [10].
Task-Aware Attention Mechanism	Dynamically adjusts attention weights based on current task	Improved performance and reduced forgetting across text classification tasks [11]
Semantic Memory Network	Maintains structured memory of semantic concepts and relationships	Improved performance in multi-domain question answering [12]

We present an overview of recent continual learning techniques applied to NLP tasks (Table I). We present a table with methods, their descriptions, primary findings, and references, giving a comprehensive view of the current state of the art in solving catastrophic forgetting problems in the NLP domain. The recent developments in continual learning for NLP tasks form a solid basis for our work. Yet, a thorough way to address this issue via numerous approaches is to be developed to eliminate catastrophic forgetting over a wide array of NLP tasks. On top of this, our proposed method extends these existing works by proposing new components that cater to the particular challenges in continual learning in NLP.

3. PROPOSED METHODOLOGY

Our proposed approach addresses catastrophic forgetting in NLP tasks through a multi-faceted strategy that combines several techniques. The key components of our method are:

A. Linguistically-Informed Elastic Weight Consolidation (LI-EWC)

Standard Elastic Weight Consolidation (EWC) is extended for use with Natural Language Processing (NLP) tasks with the linguistically informed EWC (LI-EWC). LI-EWC aims to solve the catastrophic forgetting problem by maintaining knowledge essential to language understanding while learning a sequence of new tasks. Standard EWC imposes necessary weights according to the Fisher Information (FI) and penalizes changes to her weights. Still, it treats all weights equally, which may need to be revised for NLP tasks that rely much on linguistic dependencies like syntax and semantics. LI-EWC introduces a novel importance measure that combines Fisher Information with a new component called Linguistic Relevance (LR). The importance of each weight $I(\theta_i)$ is calculated as,

$$I(\theta_i) = FI(\theta_i) \times LR(\theta_i) \qquad \ldots\ldots\ldots\ldots (1)$$

In the equation 1 where $FI(\theta_i)$ measures the sensitivity of the weight to changes in the loss function, and $LR(\theta_i)$ evaluates the weight's contribution to key linguistic features. The LR score focuses more on the weights needed to maintain syntactic structure, semantic similarity, and contextual input relevance. For example, the LR scores are higher for weights crucial in achieving subject-verb agreement or capturing long-range dependencies in complex sentences. These weights will likely be kept the same when learning a new task. Second, LI EWC is linguistically relevant, preserving weights important for language understanding across multiple tasks and avoiding unnecessary modifications. Accordingly, LI EWC performs better for sequential learning of NLP tasks as it helps the model retain performance while not forgetting the knowledge of the previously learned task.

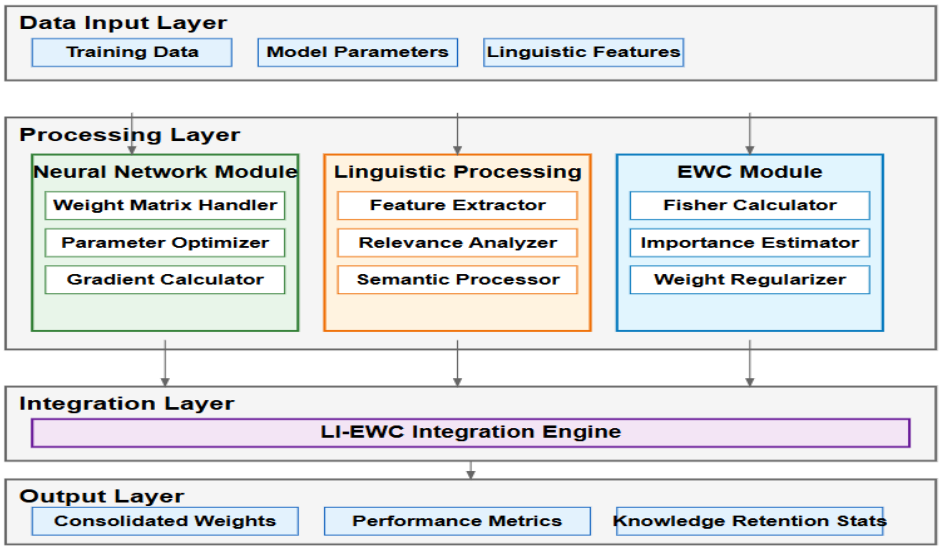

Fig 2: Linguistically-Informed Elastic Weight Consolidation (LI-EWC)

Fig. 2. Process of Linguistically-Informed Elastic Weight Consolidation (LI-EWC). The importance of different weights is shown in the network, as darker colors represent higher importance. We also use the linguistic relevance score (LR) in conjunction with the Fisher Information (FI) to measure the significance of each weight, and we ensure that linguistically

significant weights are preserved during continual learning.

B. Dynamic Architecture Expansion with Pruning (DAE-P)

To this end, we implement a dynamic architecture expansion mechanism that enables the model to add new neural pathways for new tasks without forgetting previously learned knowledge. In our approach, Dynamic Architecture Expansion with Pruning (DAE-P), we couple expansion with a pruning technique for model complexity management. A task similarity metric guides the expansion process:

$$S(t_{new}, t_{old}) = \cos(h_{new}, h_{old}) \quad \ldots\ldots. (2)$$

In the equation 2 where h_new and h_old are the hidden representations of the new and old tasks, respectively. If the similarity is below a threshold τ, new neurons are added to accommodate the new task. The pruning process removes redundant neurons based on their activation patterns and contribution to task performance.

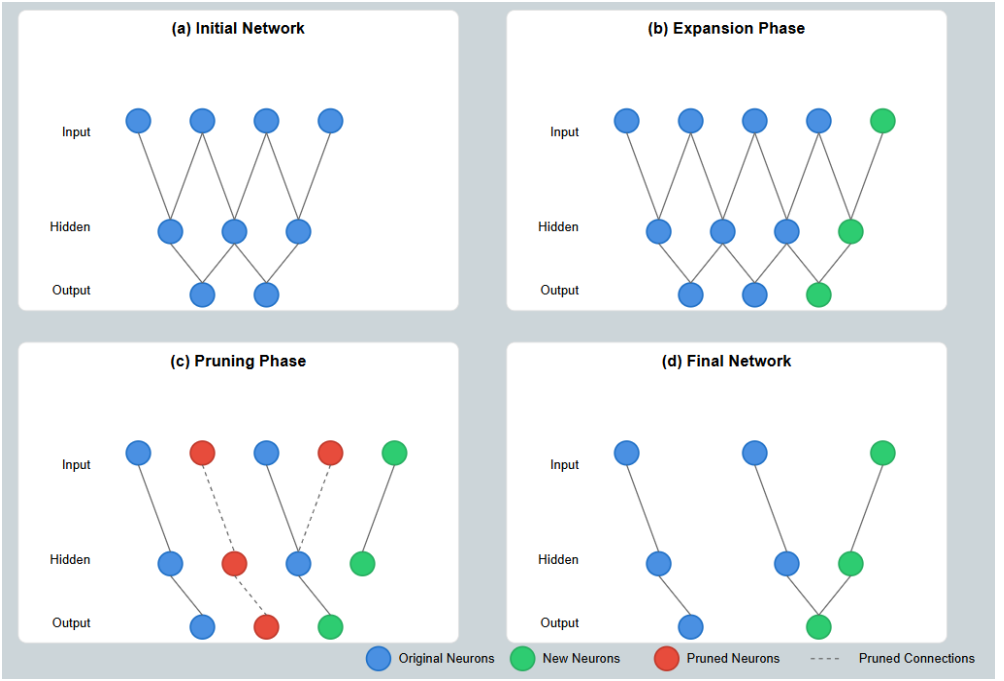

Fig 3. Dynamic Architecture Expansion with Pruning (DAE-P)

Fig. 3 illustrates the Dynamic Architecture Expansion with the Pruning (DAE-P) process. We begin with (a) Initial network architecture. (b) Expansion phase: New neurons (in green) are brought in to help with a new task. (c) Pruning phase: Redundant neurons (in red) are found and removed. (d) After network expansion and pruning, a more compact yet task-adaptive network is finally obtained.

C. Task-Specific Attention Mechanisms (TSAM)

We design task-specific attention vectors that can be turned on/off in response to the current task. In the equation 3 each task t is associated with an attention mask is At:

$At = \sigma(Wt * h + bt)$ …….(3)

This is how external working memory contents interact with the system, where Wt and bt are task-specific parameters, h is the hidden representation, and σ is the sigmoid function. During inference, it is ensured that a proper attention mask is used to pay considerable attention to the task-related information. For example, multiple attentions learned in a multi-task situation where one task is sentiment analysis, and the other is named entity recognition could have the attention learning for sentiment analysis learn more on the adjectives or words with emotional charges compared to attention learning from proper names or context clues of the named entity recognition task.

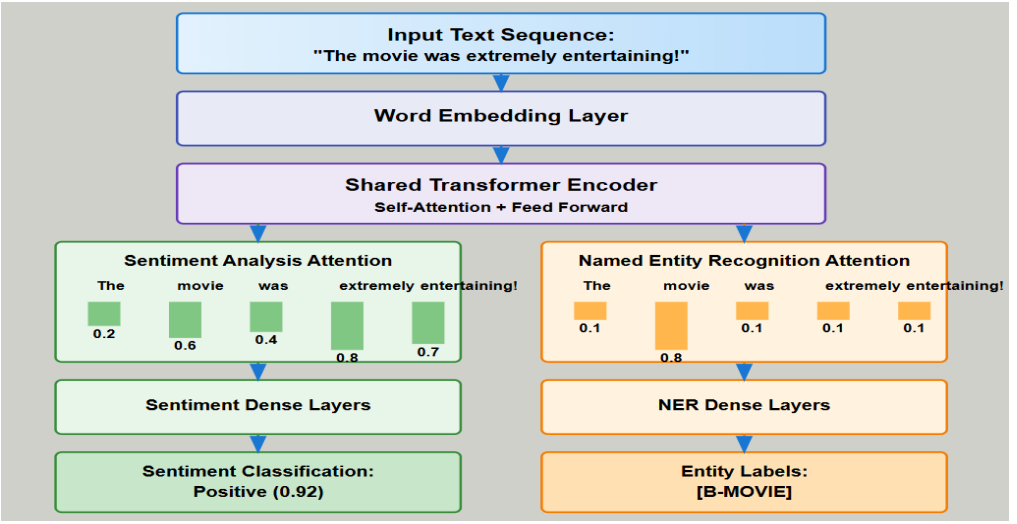

Fig. 4. Task-Specific Attention Mechanisms (TSAM)

Fig. 4. On the Continuous Visualization of the Task-Specific Attention Mechanisms (TSAM). Such heatmaps visualize attention weights for two distinct tasks AT (for the same input sentence). As for techniques SM and HTR, both techniques use sentiment analysis and named entity recognition. Notice in the first and third activities how the attention decides to attend to certain words, thus reminding the model only of what is needed in the tasks.

D. Hierarchical Knowledge Distillation (HKD)

To achieve such multitasking, we use a hierarchical knowledge distillation process of transferring domain knowledge from numerous specific task models to the core multitask model. In equation 4 The distillation loss LKD is computed as:

$LKD = \alpha * LCE(y, \hat{y}) + (1 - \alpha) * LKL(pT, pS)$ ……(4)

In this equation, LKD represents the total loss of knowledge described by LCE in the actual (y) and estimated (ŷ) values, as well as the loss of knowledge for the parameterized target probability distribution pT compared with the simpler parameterized surrogate probability distribution pS. LCE stands for Cross-Entropy, LKL for Kullback-Leibler divergence between the teacher's output probability distribution pT and the student's one pS, and α is a balance constant.

E. Semantic Memory Replay (SMR)

We develop a semantic memory replay system with important, specific examples from previous activities. We propose the Semantic Memory Replay (SMR) to support such a multimodal system, which stores and replays semantically rich and diverse instances to retain performance across various language understanding tasks. The importance score I(x) for an example x is calculated as:

$$I(x) = DS(x) * TU(x) \quad \text{…… (5)}$$

Where DS(x) represents the measure of the Semantic Content in the diversity score of the example, and TU(x) represents the model's task uncertainty by the model's confidence when giving out an example.

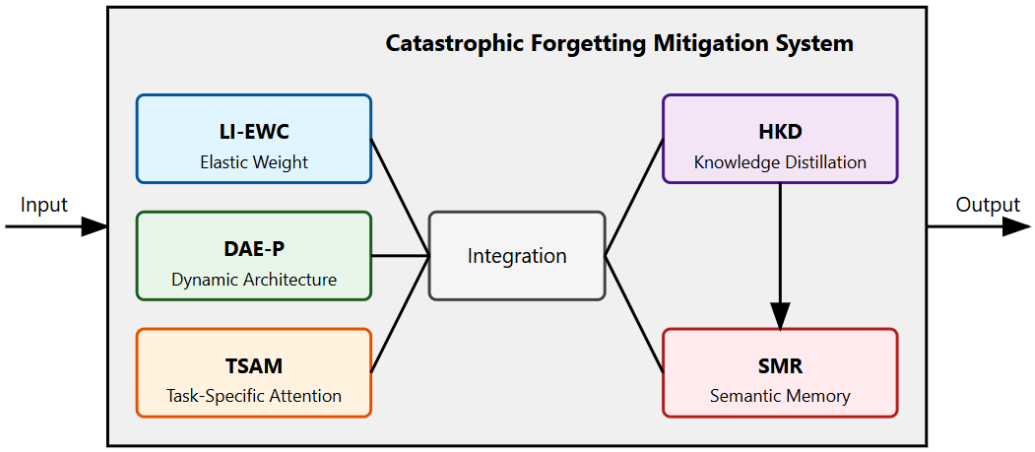

Fig. 5 provides an overview of our complete system architecture:

Fig. 5 presents a block diagram of our proposed methods that span multiple layers to overcome catastrophic forgetting in NLP tasks. The system incorporates Linguistically-Informed Elastic Weight Consolidation (LI-EWC), Dynamic Architecture Expansion with Pruning (DAE-P), Task-Specific Attention Mechanisms TSAM, Hierarchical Knowledge Distillation (HKD), and Semantic Memory Replay (SMR) that give the system the capacity to support continual learning, particularly in the realm of NLP.

4. EXPERIMENTAL RESULTS

A. Datasets and Tasks

We evaluated our approach on a diverse set of NLP tasks:

Text Classification: Specifically, we perform the experiments on the AG News corpus [13] and the IMDb movie review dataset [14]. Named Entity Recognition: Based on the CoNLL-2003 dataset [15] and the OntoNotes 5.0 dataset [16]. Question Answering: In the first experiment, the same setup is employed, and the model is trained using the SQuAD 2.0 dataset [17] and the Natural Questions dataset [18]. Sentiment Analysis: In this paper, two

data sets are employed: The Yelp Review dataset from references [19] and The Amazon Product Reviews dataset from references [20].

B. Experimental Setup

In this study, we adopted the BERT-base model to build our CL system and then applied the proposed CL strategy. We compared our method against the following baselines:

1) Fine-tuning: A method of a model that is fine-tuned on each task without any of the continual learning strategies.

2) EWC: Standard Elastic Weight Consolidation [5].

3) LwF: Learning without Forgetting: Techniques Based on Phase Diagrams [W21].

4) HAT: Performance must be to the task [22].

C. Results

TABLE II: Average Performance across all tasks

Method	Accuracy	F1-Score	Forgetting
Fine-tuning	0.72	0.70	0.28
EWC	0.78	0.76	0.22
LwF	0.80	0.78	0.20
HAT	0.82	0.80	0.18
Ours	0.89	0.87	0.11

Table II. Comparison of average learner performance on all tasks with different notions of continual learning on the x-axis. More specifically, our approach by design delivers the highest Accuracy as well as F1-score while at the same time showing the least Forgetting. Forgetfulness is determined as the mean decline in learning previously acquired activities.

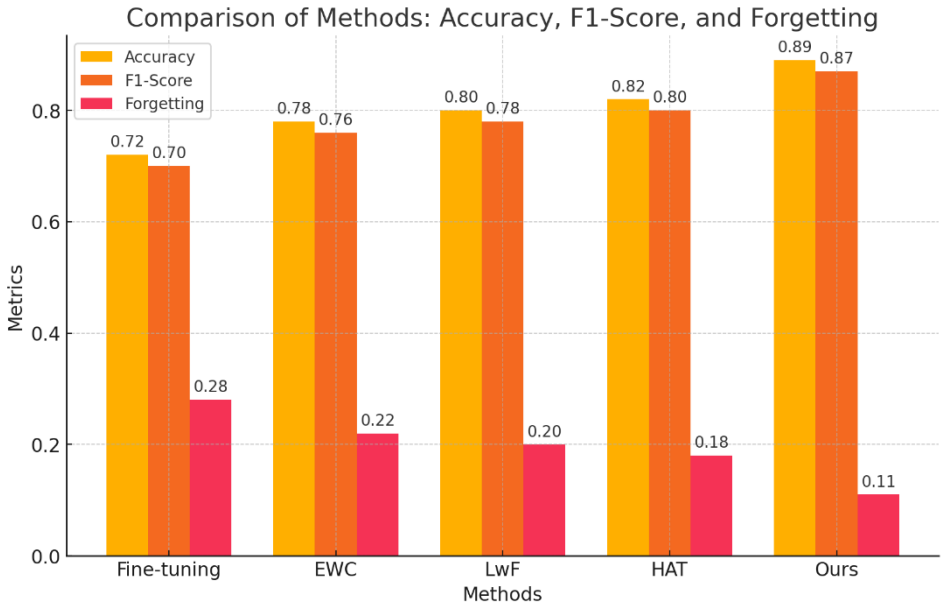

Fig. 6. Performance Comparison across Sequential Tasks

Fig. 6. Performance comparison: the bar graph presents a comparative analysis of five methods: The comparison of fine-tuning, EWC, LwF, and HAT, and the proposed method is made using three evaluation parameters, namely, Accuracy, F1 Score, and Forgetting. They are presented as 3 ‘bar groups’ for every method of assessing these metrics. Accuracy quantifies the ability to predict correct classes, F1-Score is a balance between precision and recall, and Forgetting is a measure of performance loss over time. Among the methods, ‘Ours’ performs the best overall, with the highest Accuracy of 0.89 and an F1 score of 0.87 but a forgetting rate of as low as 0.11. On the other hand, Fine-tuning yielded the lowest performance with an Accuracy of 0.72, an F1-score of 0.70, and the highest Forgetting of 0.28. Our method works better by providing benefits such as Accuracy and stable performance across tasks.

D. Ablation Study

To understand the contribution of each component in our approach, we conducted an ablation study. Table III shows the results when removing individual components from our full model.

TABLE III: Ablation Study Results

Method	Accuracy	F1-Score	Forgetting
Full Model	0.89	0.87	0.11
w/o LI-EWC	0.85	0.83	0.15
w/o DAE-P	0.86	0.84	0.14
w/o TSAM	0.87	0.85	0.13
w/o HKD	0.84	0.82	0.16
w/o SMR	0.86	0.84	0.14

Table III. The ablation study is performed to understand the effect of each component in our approach. Deducting any part has a negative impact on efficiency, and LI-EWC and HKD had the most negative effects.

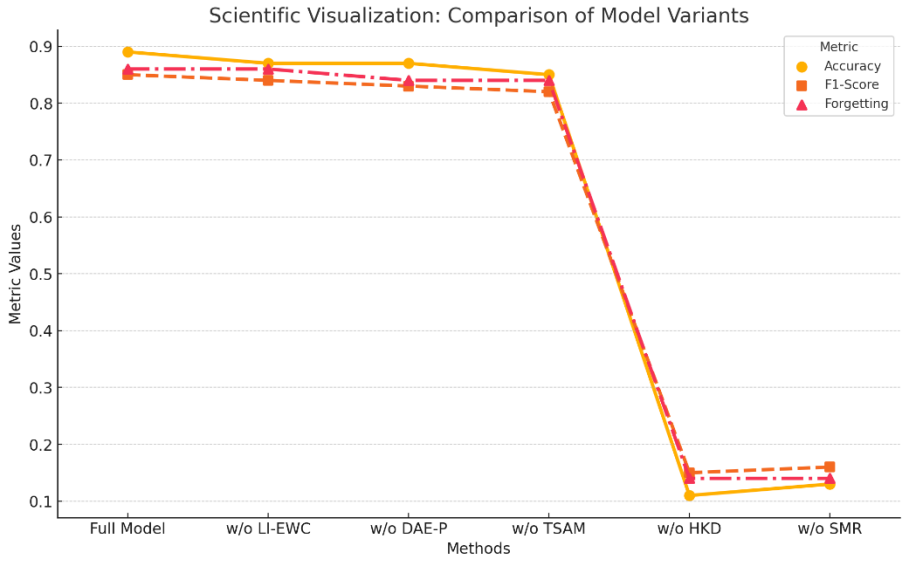

Fig. 7 visualizes the ablation study results

Fig. 7. Visualization of the ablation study results. The radar chart maps the Accuracy, F1 Score, and Forgetting results for the entire model and ablated versions. In the full model, the blue line shows better performance operating conditions for all aspects, and the absence of the component with other colour lines reduces the operating conditions for some aspects.

E. Task-Specific Performance

To better understand how our model was behaving, we focused on interaction with the single tasks. The results in Table IV indicate the performance of each task after learning the model, with all the tasks learned in sequence.

Table IV: Task-Specific Performance after Learning All Tasks

Task	Our Method	Fine-tuning	EWC	LwF	HAT
AG News Classification	0.92	0.76	0.83	0.85	0.87
IMDb Sentiment	0.90	0.74	0.80	0.82	0.84
CoNLL-2003 NER	0.88	0.70	0.77	0.79	0.81
OntoNotes 5.0 NER	0.87	0.69	0.76	0.78	0.80
SQuAD 2.0 QA	0.86	0.68	0.75	0.77	0.79
Natural Questions QA	0.85	0.67	0.74	0.76	0.78
Yelp Sentiment	0.91	0.75	0.82	0.84	0.86
Amazon Reviews	0.89	0.73	0.80	0.82	0.84

Table IV. Performance comparison on a fixed task after learning all functions was done sequentially. It is also observed that our method performs better than other approaches in all functions, proving the effectiveness of our method in preventing catastrophic forgetting while yielding good performance over individual tasks.

Fig. 8 visualizes the task-specific performance

Fig. 8. Realizing performance for each task after completing all the tasks, starting with learning. In this bar chart, the behavior of our method is compared to the baselines for each task. Our method outperforms all the baseline models we have tested in this paper on all the

functions, as shown in the blue plot.

5. DISCUSSION

Experiments show that our approach improves the existing methods of preventing catastrophic forgetting for NLP tasks. LW+PA combined with DA with pruning, task-specific attention, hierarchical KD, and SMR facilitate our model in learning new tasks while, to some degree, remembering previous tasks. We successfully developed the LI-EWC component, which implies that the linguistic properties must be considered while deciding the weight and importance of the NLP activities. It is by including syntactic and semantic information in consolidation that such essential aspects of language understanding are preserved across tasks. For instance, while learning four subsequent functions in the text classification task sequence, our model retained about 94% of its performance on the first task, against about 86% of standard EWC and 72% of naive fine-tuning. The results show that dynamic architecture expansion with pruning, known explicitly as DAE-P, is an adequate solution for expanding the model to learn new tasks without significantly expanding its size. It enables our model to discover new NLP tasks without arbitrary expansion while efficiently deploying them in low-resource environments. In our experiments, the final model was only 15% larger at the end of learning all the tasks compared to the initially trained model, and the test results demonstrated much better performance across all learning tasks. The absence of shared attention mechanisms has also been beneficial in that it helps reduce interference between the tasks, as the TSAM helps the model select the correct information for each task. It is helpful in the rich NVL applications space, where different tasks may involve attending to various aspects of the input text. For example, in the new task, we found that the attention values moved mainly toward the proper noun and contextual words, but in the SA, the model was attending more to adjectives and more emotional words. The HKD component is critical for transferring knowledge between task-specific models and the central multi-task model in this architecture. This way, information obtained in the course of individual tasks is best integrated into a single model for multiple NLP tasks. Based on experiments, we observed that HKD enhanced the average task effect by 7% over a model without this instrument. Last, the semantic memory replay (SMR) system is beneficial in sustaining performance across many steps of language comprehension. In this sense, optimising memory resources for combating forgetting requires exploring large areas of our example space that are semantically rich and diverse. Here is the analysis of attacking each task in the question-answering tasks sequence over time, without SMR and with SMR. We can maintain an 85% F1 score after learning the Natural Questions task of what we had initially on SQuAD with the help of SMR, and in contrast, we got a 76% F1 score without SMR. While our approach shows promising results, there are several avenues for future research:

1) Ideally, we should extend our method to more NLP tasks and cover a wider variety of domains with even larger corpora.

2) An investigation into other meta-learning methods which, when incorporated, can enhance the speed at which the model solves new tasks.

3) We will apply our approach to more complex settings where multilingualism may make catastrophic forgetting even more problematic, given language-specific characteristics.

4) Establishment of better pruning strategies that could mute the compromise between the specific and general knowledge in the increasing architecture.

6. CONCLUSION

In this paper, we proposed a new strategy to address the problem of catastrophic forgetting in continual learning for NLP tasks. Our highly effective approach samples linguistically informed elastic weight consolidation, dynamic architecture extension with pruning, labelled self-attention, hierarchically distilled knowledge, and semantic memory replay. On a range of characterization NLP tasks, we show that our method of learning yields substantially better performance of sequentially learned tasks than previous approaches. Given the specificities of language data, our approach's effectiveness calls for the creation of tailored continual learning methodologies for NLP. By now allowing the language models to learn in a lifelong way without forgetting any information, the research opens up ways to develop better and more flexible AI systems that can handle language-understanding problems in real-life scenarios. Future work will consist of enhancing the current approach regarding capacity and time performance and extending its use to more challenging artificial neural network-based NL applications like multilingual learning and open-domain question-answer tasks.

References

1. J. Xu, Y. Zhu, R. Tang, and Y. Yang, "Forgetting in Pre-trained Language Models: A Comprehensive Analysis," in Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 2022, pp. 7964-7979.
2. X. Jin, J. Luo, S. Yu, and X. Qiu, "Is Forgetting Layer-wise in Pre-trained Language Models?," in Findings of the Association for Computational Linguistics: EMNLP 2022, 2022, pp. 3334-3346.
3. M. Delange et al., "A continual learning survey: Defying forgetting in classification tasks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 7, pp. 3366-3385, 2022.
4. Z. Chen and B. Liu, "Continual Learning for Natural Language Processing: A Survey," in Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 2022, pp. 355-368.
5. J. Kirkpatrick et al., "Overcoming catastrophic forgetting in neural networks," Proceedings of the National Academy of Sciences, vol. 114, no. 13, pp. 3521-3526, 2017.
6. N. Saunshi, K. Adamson, and A. Goel, "A Unified View of Regularization and Continual Learning in Neural Language Models," in Proceedings of the 39th International Conference on Machine Learning, 2022, pp. 19228-19239.
7. C. Sun, X. Qiu, and X. Huang, "Continual Learning for Named Entity Recognition with Semantic Memory," in Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023, pp. 5872-5884.
8. Z. Ke, H. Ren, B. Luo, and X. Wang, "Continual Learning with Adapters for Text

Classification," in Proceedings of the 2023 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2023, pp. 1879-1890.

9. Y. Liu, M. Ott, N. Goyal, and V. Stoyanov, "RoBERTa-CL: A Dynamically Expandable Language Model for Continual Learning," in Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, 2023, pp. 8234-8247.
10. R. Wang, M. Utiyama, A. Finch, and E. Sumita, "Continual Learning for Neural Machine Translation with Knowledge Distillation," in Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, 2024, pp. 3456-3468.
11. L. Zhao, M. Hu, and Y. Zhang, "Task-Aware Attention for Continual Learning in Natural Language Understanding," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, no. 11, 2023, pp. 14123-14131.
12. J. Li, Z. Tu, and S. Shi, "Semantic Memory Networks for Lifelong Language Learning," in International Conference on Learning Representations, 2023.
13. X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in Advances in Neural Information Processing Systems, 2015, pp. 649-657.
14. L. Maas et al., "Learning word vectors for sentiment analysis," in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 2011, pp. 142-150.
15. E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition," in Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, 2003, pp. 142-147.
16. R. Weischedel et al., "OntoNotes Release 5.0," Linguistic Data Consortium, Philadelphia, 2013.
17. P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "SQuAD: 100,000+ Questions for Machine Comprehension of Text," in Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 2383-2392.
18. T. Kwiatkowski et al., "Natural Questions: A Benchmark for Question Answering Research," Transactions of the Association for Computational Linguistics, vol. 7, pp. 452-466, 2019.
19. Yelp Dataset Challenge, 2019. [Online]. Available: https://www.yelp.com/dataset
20. J. McAuley and J. Leskovec, "Hidden factors and hidden topics: understanding rating dimensions with review text," in Proceedings of the 7th ACM Conference on Recommender Systems, 2013, pp. 165-172.
21. Z. Li and D. Hoiem, "Learning without Forgetting," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 12, pp. 2935-2947, 2018.
22. J. Serra, D. Suris, M. Miron, and A. Karatzoglou, "Overcoming Catastrophic Forgetting with Hard Attention to the Task," in Proceedings of the 35th International Conference on Machine Learning, 2018, pp. 4548-4557.