Continual Learning using Transformers

Pranav Balaji
CSIS Department
BITS Pilani, Hyderabad Campus
Hyderabad, India
f20190040@hyderabad.bits-pilani.ac.in

Sathvik Bhaskarpandit
CSIS Department
BITS Pilani Hyderabad Campus
Hyderabad, India
f20191200@hyderabad.bits-pilani.ac.in

T V Chandra Vamsi
CSIS Department
BITS Pilani, Hyderabad Campus
Hyderabad, India
f20190033@hyderabad.bits-pilani.ac.in

***Abstract*—In the domain of Natural Language Processing (NLP), it is unheard of for models to be trained on multiple tasks sequentially. This is because after the first task, there is a significant dip in performance on the first task while training for the second task. This is known as catastrophic forgetting. Continual learning aims to combat this problem by retaining knowledge of previous tasks while being able to adapt to any new task. We utilize a simple regularization based method along with transformers which can adapt to almost any NLP task while being fast and memory efficient.**

***Index Terms*—neural networks, natural language processing, catastrophic forgetting, continual learning, elastic weight consolidation**

I. Introduction

Textual data is ubiquitous and the amount of such data at hand increases rapidly. Several complex natural language processing (NLP), natural language understanding (NLU) and natural language generation (NLG) models have been proposed to deal with textual data. Yet, many of these models are designed to be trained for a specific purpose or problem, for example, sentiment analysis, paraphrase generation, sentiment analysis, etc. While humans and animals intrinsically possess the ability to retain knowledge from various sources and for a long period of time, computational systems do not. When several tasks are trained independently and sequentially, catastrophic forgetting [1] is observed in neural networks. Catastrophic forgetting is the tendency of neural networks to completely and abruptly forget previously learned information upon learning new information. This is a common problem existent in deep learning as weights obtained after training for a certain task change drastically when subsequently trained for another task.

In light of catastrophic forgetting, several techniques have been proposed to alleviate this major problem. Nevertheless, a separate study has emerged for lifelong learning or continual learning: that is, the ability of a deep neural network to learn consecutive tasks without forgetting how to perform on previously trained tasks. There has been significant research on continual learning in the field of computer vision, with NLP rather behind. Additionally, almost all of these approaches have been proposed to deal with streams of fundamentally indifferent tasks. However, in the field of NLP, it is sometimes desirable to have a model that can perform several fundamentally different tasks without catastrophic forgetting.

Transfer learning can be used to solve several fundamentally different tasks. In computer vision, pre-training is typically done via supervised learning on a large labeled data set like ImageNet. Modern techniques for transfer learning in NLP often pre-train using unsupervised learning on unlabeled data. This approach has recently been used to obtain state-of-the-art results in many of the most common NLP tests and benchmarks. Nonetheless, transfer learning cannot effectively subsidize the effect of catastrophic forgetting. As the number of tasks become large, the experience from the pretrained task, may not perform well on subsequent tasks (negative transfer).

[2] have proposed an interesting approach where each NLP task is treated as a text to text problem. We utilize a model that adopts this idea and make use of Elastic Weight Consolidation [3] to limit catastrophic forgetting, while simultaneously able to adapt to various types of NLP tasks. Our model has an added advantage that it is computationally inexpensive and memory efficient

Our main contributions to this paper is as follows:

- We utilize a text to text model that can be learned on various fundamentally different NLP tasks
- We demonstrate that our text to text approach when combined with a simple regularization based continual learning approach is adequate to alleviate the problem of catastrophic forgetting
- We compare our model performance to other proposed models and measure the amount of positive transfer and catastrophic forgetting

II. Related Work

A. Lateral Transfer

Progressive Neural Networks [4] avoid catastrophic forgetting by adding support for lateral transfer, allowing useful features to be extracted for new tasks. Forgetting is also prevented by adding a new column of neural network for each task that is being solved. However, as the number of tasks increases, the number of parameters as well as columns. The paper demonstrates positive transfer in Reinforcement Learning (RL) domains by using RL agents within a continual learning framework.

B. Regularization-Based Methods

1) Elastic weight consolidation: (EWC) [3] ensures that catastrophic forgetting does not take place by selectively

decreasing the plasticity of weights (important parameters are constrained to their old values). EWC regularizes the model parameter at every step, enabling the model to find a good fit for both tasks.
Suppose that the parameters are set such that the performance is optimized for a task A. While learning task B, EWC retains the task A performance by constraining the parameters for task A to stay in a region of low error for task A. The model is implemented such that the constraint gives us a quadratic penalty, similar to a spring stretching from its old parameters, hence the name. The paper demonstrates continual learning with EWC in a supervised learning context and reinforcement learning context.

2) Information Disentanglement Based Regularization: A continual learning model that uses information disentanglement based regularization on text classification [5]. Text hidden spaces are disentangled into representations common to tasks, and these representations are further regularized to narrow down the knowledge that needs to be generalized.

3) Episodic Memory: Gradient Episodic Memory (GEM) [6], a model for continual learning, uses episodic memory to minimize catastrophic forgetting. The episodic memory stores some representative examples for each task. The model's ability to learn is measured with backward transfer (BWT) and forward transfer (FWT). BWT is positive if improvement in performance of previous tasks are observed after training on new tasks. Large negative BWT indicates that catastrophic forgetting is taking place. FWT is positive if improvement on new tasks is observed after training on previous tasks.

4) Improved memory-based parameter adaptation: A lifelong learning model with episodic memory is used for sparse experience replay (sampling from examples at uniform intervals of time and performing gradient updates) and local adaptation [7] to protect it from catastrophic forgetting.

5) Synaptic Framework: Forgetting can be alleviated using a synaptic framework [8] for neural networks, where each synapse estimate their importance for solving past tasks.

6) Dynamically Expandable Networks: Dynamically Expandable Networks (DEN) [9], partially retains its old network, and takes advantage of similarities between various tasks. It increases its capacity to learn new tasks and prevents catastrophic forgetting effectively.

7) LAMOL: LAnguage MOdeling for Lifelong Language Learning (LAMOL) [10], automatically generates sample training examples of previous tasks without the use of extra memory or model capacity. Hence, it simultaneously learns as well as generate examples, preventing catastrophic forgetting. The LAMOL model can perform five very different language tasks sequentially.

C. Transfer Learning

Text-to-Text Transfer Transformer (T5) [2], uses a transformer architecture that converts all text-based language problems into a text-to-text format and allowing us to use the same model for a wide variety of tasks.

III. Methodology

A. Data Pre-processing

Textual data from various datasets as mentioned in section IV-A is used to train our model. No stemming or lemmatization is done to preserve grammatical syntax and semantics. Each input sequence is padded with a ¡pad¿ token to the required length. The end of sentence token is ¡/s¿ and unknown token is ¡unk¿. Depending on the exact NLP task required to be performed, we add a prefix at the beginning of an input sequence. The sequences are then fed into a tokenizer. These tokenized sequences are used as input to the transformer.

B. Training

Assume a stream of tasks $\{T_A, T_B, ...\}$, where the number and type of tasks may be unknown. Directly training the LM on these tasks sequentially results in catastrophic forgetting.

The first task T_A is trained as usual. After T_A has finished training, the next task T_B is trained on, but with an extra regularization (EWC) term. The overall loss for training T_B is given as:

$$\mathcal{L}(\theta) = \mathcal{L}(\theta_B) + \frac{\lambda}{2}\sum_i F_i(\theta_i - \theta^*_{A,i})^2 \quad (1)$$

where,
$\mathcal{L}(\theta_B)$ = loss for training T_B only,
θ_A = optimal parameters for T_A,
F_i represents the element at (i, i) in the Fisher information matrix
λ = hyperparameter that signifies how important the old task T_A is compared to the new one T_B

When a third task T_C arrives, T_A and T_B together can be treated as the old task, and F is computed for them jointly.

IV. Experimental Setup

A. Datasets

1) SQuAD (Stanford Question Answering Dataset): [11] A large reading comprehension dataset consisting of question-answer pairs from Wikipedia articles. The dataset contains over 100,000 questions, and the answer to each question is present in the corresponding passage of the Wikipedia article. The input given to our model is a question, along with its context, and is fed into the model.

2) MNLI (Multi-Genre Natural Language Inference): This dataset is a collection of 433k sentence pairs with information about entailment of the pair of sentences. Each pair has a label associated with it, where, '0' represents entailment, '1' represents neutral, '2' represents contradiction.

3) PAWS (Paraphrase Adversaries from Word Scrambling): [12] This dataset consists of 108,463 human labelled and 656k noisily labelled pairs generated from Wikipedia pages and Quora. Each pair has two sentences, and each pair has a label associated with it, where, '0' represents different_meaning, '1' represents paraphrase. We removed the instances where the label = '0', and reversed the instances (the target sentences became input sentences). We name our transformed dataset as eqPAWS.

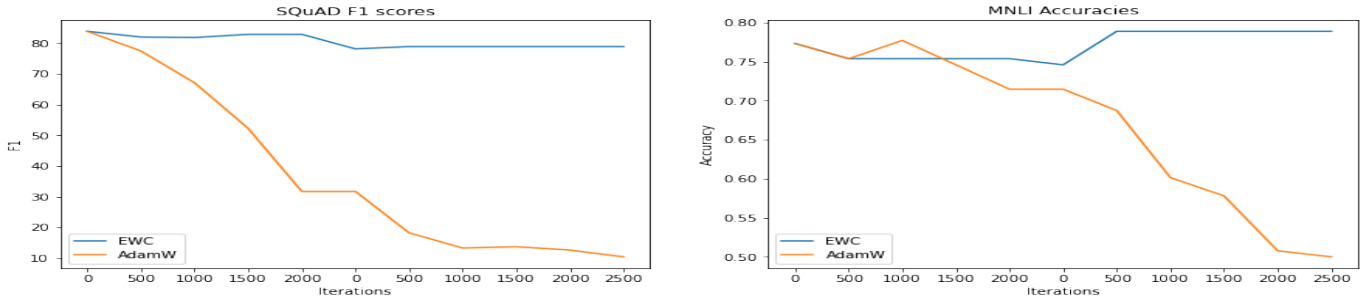

Fig. 1. SQuAD F1 Scores and MNLI Accuracy

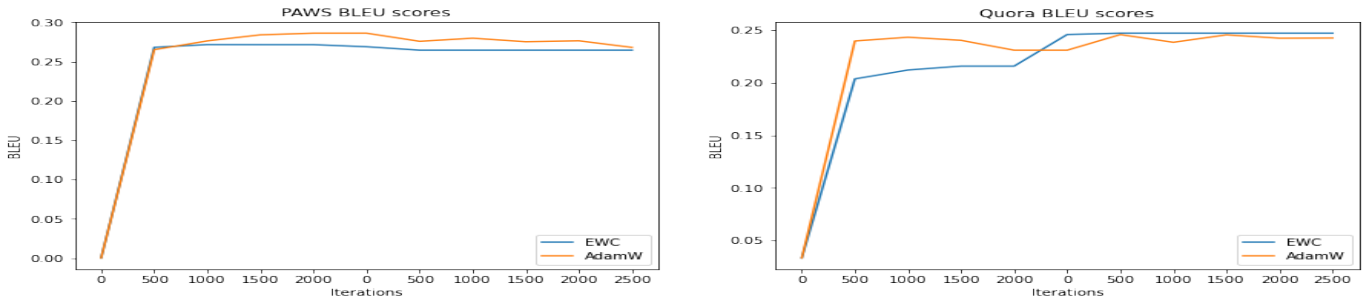

Fig. 2. PAWS and Quora BLEU Scores

	SQuAD F1	SQuAD Exact Match (%)	MNLI - matched Accuracy (%)	MNLI - mismatched Accuracy (%)	eqPAWS BLEU	eqQuora BLEU
T5-Small original	84.12	74.95	80.40	81.94	0.00	0.04
After standard fine-tuning	12.83	0.10	49.47	49.58	0.27	0.25
After EWC training	81.48	70.87	78.59	78.88	0.27	0.24

4) Quora Question Pairs: The Quora dataset consists of question pairs, where the task is to determine whether the questions are paraphrases of each other. Each pair has a label associated with it, where, 'false' represents different meanings and 'true' represents paraphrases. We use the same transformation as the PAWS dataset, and name our dataset as eqQuora.

B. Training workflow

We use the pre-trained model T5-Small [2] to conduct our experiments. Out of all the tasks it was trained on, we consider the "to be remembered tasks" to be Question-Answering and Natural Language Inference. The model is fine-tuned for Paraphrase Generation using the eq-PAWS dataset and later the eq-Quora dataset. During the fine-tuning process, the model is evaluated on all of these tasks every 500 iterations. Fine-tuning stops when the BLEU score for the currently training task does not deviate by 10^{-3}

C. Loss function

We first fine-tune the T5-Small model for paraphrase generation using cross-entropy as the loss function. Afterwards, we re-initialize all the parameters and fine-tune them for paraphrase generation using the modified loss function described in Equation 1.

D. Hyperparameters

An initial learning rate of 3×10^{-4} along with the AdamW optimizer [13] scheduled linearly was found give the best results. Early stopping was used to stop training when validation loss saturated. Convergence was observed to be achieved relatively fast.

E. Evaluation

While fine tuning, metrics for each task are calculated on 128 randomly sampled instances from the corresponding datasets. After fine tuning, the metrics for each task are calculated on the entire dataset. The metrics and dataset for each task are:

1) Question-Answering: Metrics: F1 score and Exact Match. Dataset: SQuAD validation.

2) Natural Language Inference: Metrics: Accuracy. Dataset: MNLI validation-matched and mismatched. (Note that evaluating on the validation-mismatched is omitted during the fine-tuning process)

3) Paraphrase Generation: Metrics: BLEU score. Dataset: eq-PAWS while training on eq-PAWS, then eq-Quora while training on eq-Quora.

V. Results and Discussion

T5-Small was originally trained on SquAD, MNLI and a couple of other datasets (excluding PAWS and Quora). It was

originally trained to identify a pair of paraphrases (labelled as 0 or 1) instead of generating them. From our results, it is evident that the model performs well on SQuAD and MNLI, but poorly on paraphrase generation (eqPAWS and eqQuora datasets). This is to be expected, because it wasn't a task it was trained on.

We then fine-tune the model for paraphrase generation using the eqPAWS and eqQuora datasets. We evaluate the model on all the mentioned tasks although performance on paraphrase generation has gone up, the performances on SQuAD and MNLI have decreased drastically. Formally, our model has exhibited catastrophic forgetting. It is unable to retain knowledge of past tasks when exposed to new ones.

We then re-initialize all the parameters of the model, and train it on the same new task, paraphrase generation, using EWC loss. On evaluation, we observe that the performance on old tasks are much better than with standard fine-tuning, while also being able to learn new task reasonably well. This shows that the new model is much less prone to catastrophic forgetting.

VI. Conclusion and Future Work

From our experiments, it can be inferred that using EWC in a sufficiently large model like T5 which has numerous parameters (many of which are not as important to a task as others) makes fine-tuning possible while retaining previous knowledge.
Additionally, it is computationally inexpensive, memory efficient and can adapt to fundamentally different NLP tasks. In the near future we look to increase the number of NLP tasks as well as observe performance as we do so.

References

[1] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of learning and motivation*, vol. 24, pp. 109–165, Elsevier, 1989.

[2] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *arXiv preprint arXiv:1910.10683*, 2019.

[3] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[4] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[5] Y. Huang, Y. Zhang, J. Chen, X. Wang, and D. Yang, "Continual learning for text classification with information disentanglement based regularization," *arXiv preprint arXiv:2104.05489*, 2021.

[6] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," *Advances in neural information processing systems*, vol. 30, pp. 6467–6476, 2017.

[7] C. d. M. d'Autume, S. Ruder, L. Kong, and D. Yogatama, "Episodic memory in lifelong language learning," *arXiv preprint arXiv:1906.01076*, 2019.

[8] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *International Conference on Machine Learning*, pp. 3987–3995, PMLR, 2017.

[9] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, "Lifelong learning with dynamically expandable networks," *arXiv preprint arXiv:1708.01547*, 2017.

[10] F.-K. Sun, C.-H. Ho, and H.-Y. Lee, "Lamol: Language modeling for lifelong language learning," *arXiv preprint arXiv:1909.03329*, 2019.

[11] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," *arXiv preprint arXiv:1606.05250*, 2016.

[12] Y. Zhang, J. Baldridge, and L. He, "Paws: Paraphrase adversaries from word scrambling," *arXiv preprint arXiv:1904.01130*, 2019.

[13] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2019.