Inter-Batch Cross-Attention: See More to Forget Less

Anonymous ACL submission

Abstract

Our paper presents a simple training strategy to help prevent catastrophic forgetting in continual learners, named Inter-Batch Cross-Attention (IBCA). We discover that adding an IBCA module at the input level can significantly increase the model's continual learning performance, with minimum memory and performance overhead. Our method makes minimum changes to existing transformer-based model architectures and can be used in parallel with other continual learning strategies. We demonstrate its effectiveness on class-incremental classification tasks on the 20 Newsgroups dataset.

1 Introduction

The ability of an artificial intelligence system to continuously adapt to new tasks and new data has been a main focus in the field of continual learning (CL), which is also known as lifelong learning. Unlike traditional AI models which are trained to fit a static dataset, continual learners are more suitable for real-life applications, where new data and tasks are dynamically allocated to the system continuously. However, such systems usually suffer from catastrophic forgetting, where the model's performance on previously seen tasks drops significantly as new tasks and classes are introduced continually. This phenomenon can be attributed to the major difference in the memory mechanism between human intelligence and machine intelligence. The human brain has a long-term memory for retrieval and a short-term working memory that interacts with active tasks. In contrast, most existing language models are trained on segmented data, with a limited context horizon. Neural science has shown that better working memory is related to better long-term memory. This correlation is also observed in LLMs. When models are trained with a longer context, more context memorization can lead to better temporal consistency in the generated results.

Inspired by these observations, we propose a simple but effective training scheme that has a major discrepancy over traditional model training. We enable inter-batch interactions within a batch of training samples, by introducing an Inter-Batch Cross-Attention. The intuition behind this design is to introduce sample-wise context to the model, which will assist the learning of more general, and thus less shift-prone features.

With our proposed training scheme, we see a 2% improved performance on the 20 Newsgroups class-incremental learning benchmark without and without experience replay. This is achieved without any additional continual learning strategies tailored for this task. Our experiences and analysis indicate that this more human-like training scheme potentially closes the gap between artificial intelligence and human intelligence in terms of continual learning performance.

2 Related Work

Class incremental learning (CIL) has been widely studied in continual learning literature. It is considered one of the most challenging tasks in a continual learning setting which requires models to retain and integrate knowledge across incrementally introduced classes. Techniques spanning several categories such as regularization (Kirkpatrick et al., 2016; Gok et al., 2023; Li and Hoiem, 2016; Kirkpatrick et al., 2016; Mi et al., 2020), knowledge distillation (Li and Hoiem, 2016; Hui et al., 2021), memory mechanisms (Chaudhry et al., 2018; Sprechmann et al., 2018; Wang et al., 2021; Shao et al., 2023; Hu et al., 2021; Madotto et al., 2020), experience replay(Sun et al., 2019; Song et al., 2023), data augmentation (Wang et al., 2024; Ke et al., 2022), and dynamic networks (Ke et al., 2021), have been adopted to resolve catastrophic

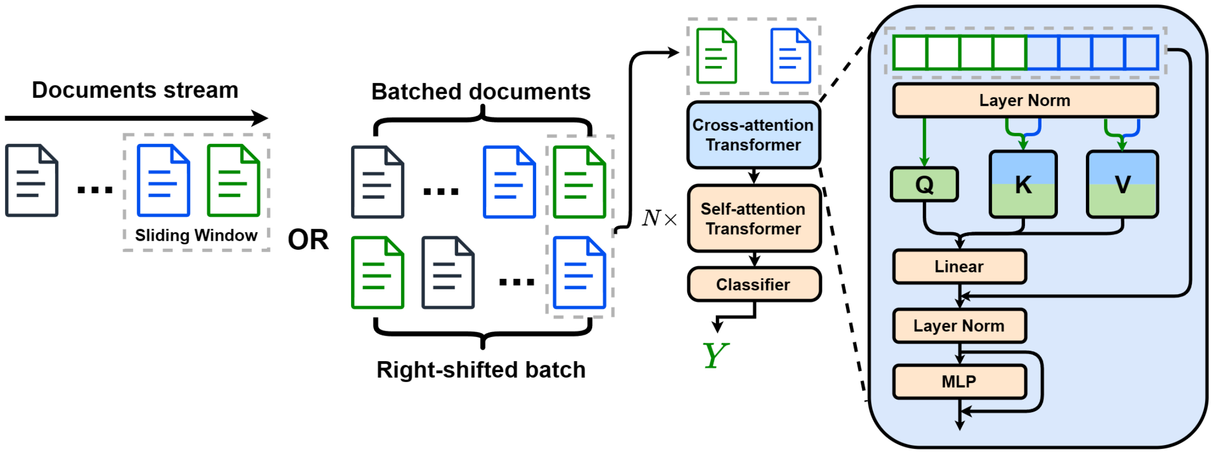

Figure 1: Pipeline and the cross-attention mechanism used in our method.

forgetting in language classification tasks.

However, in this work, we focus on the aspect of training. We believe that a more human-like training scheme might give us some insight into the reason behind catastrophic forgetting. (Lake and Baroni, 2023) discovers that by providing a longer context consisting of multiple samples, a language model can generalize more like a human in a composition task. Inspired by this work, we introduce IBCA, which enables a similar training scheme on a wide range of transformer-based batch-training pipelines.

3 Background

Class Incremental Learning: Class Incremental Learning (Kim et al., 2022) is a learning paradigm where a model is exposed to a sequence of classes $C_1, C_2, \ldots, C_n$ within a single task T. The objective is for the model to learn these classes sequentially, such that after learning all classes $C_1, C_2, \ldots, C_n$, it can correctly classify examples from all classes $\bigcup_{i=1}^{n} C_i$.

Formally, a model M undergoing class incremental learning operates in three phases. Initially, given a dataset D_1 consisting of classes C_1, the model M is trained to classify instances from C_1. In the incremental learning phase, for each subsequent set of classes C_i with dataset D_i, the model M is updated to classify instances from $\bigcup_{j=1}^{i} C_j$, ensuring minimal accuracy loss for previously learned classes $\bigcup_{j=1}^{i-1} C_j$. Finally, the model's performance is evaluated on a test set that includes examples from all classes $\bigcup_{j=1}^{i} C_j$ after each increment C_i.

4 Methodology

4.1 Model Architecture

Batch-wise Context Expansion: We introduce additional information for each training sample by expanding the context of each sample with the previous in the same batch. This can be achieved by concatenating neighboring data in a data stream, or concatenating the right-shifted copy of the batch with the original batch (See Figure 1). We set a 30% probability of the context being substituted with a zero context in the training phase since this setting produces the best results in our ablation study (Section 5.4). While in the testing phase, the context is introduced the same way as training, except no zero context is introduced.

Batch-wise Cross Attention: To process the concatenated input, we use a cross-attention transformer layer (See Figur 1 right). Given the target sequence and the concatenated context sequence, we calculate the query matrix from the target sequence (the sequence for which we predict the class) and the key and value matrices from the concatenated sequence which contains the additional context and the target sequence concatenation. This mechanism is only carried out in the first layer and its output is passed to the successive self-attention transformer layers.

5 Experiments and Results

5.1 Experimental Settings

The experiments were conducted on a system running with two Tesla T4 GPUs. For training, we used 40 epochs with a learning rate of 1×10^{-4}. The batch size was set to 32 samples per batch, the embedding dimension was set to 512, and the maximum sequence length was also set to 512. The number of attention heads were 4, while the number of layers were 3. The attention dropout rate and layer dropout rate were both set to 0.5 and 0.2 respectively and the layer normalization epsilon was set to 1×10^{-5}.

5.1.1 Dataset

The 20 Newsgroup dataset was used, comprising 20 classes with approximately 1000 documents per class. For class-incremental learning settings, 10 tasks were created with 2 classes assigned per task.

5.2 Evaluation Metrics

The evaluation metrics included **Average Accuracy (AA)**, which is the average accuracy of all tasks at the end of the last task, and **Average Forgetting (AF)**, which is the average forgetting ratio of all tasks at the end of the last task.

5.3 Main Results

Model Name	**20 News**		Speed (s)	Buffer
	AA ↑	**AF ↓**		
Baselines (Non-continual learning)				
Full	55.93	-	7509	-
IBCA (ours)	**57.49**	-	9613	-
Class incremental learning without replay				
None	15.04	77.76	10068	-
IBCA (ours)	**15.24**	77.74	12376	-
Class incremental learning with replay				
Replay	45.80	0.3454	13475	1000
IBCA (ours)	**46.93**	0.32	17379	1000

Table 1: Performance comparison with the baselines. Speed measures the time required for training by a model. Buffer Size represents the number of document samples replayed in total.

Table 1 shows IBCA's performance against the baselines in a non-continual-learning setup, class-incremental learning with replay, and class-incremental learning without replay. A full baseline model is a conventional approach trained simultaneously on all 20 classes. The IBCA model outperforms the traditional baseline model with a performance improvement, as reflected by a higher accuracy (57.49% compared to 55.93%). In the CIL without replay setup, IBCA reports a 2-3% decrease in average forgetting and a 2% increase in the average accuracy, with a negligible additional amount of computation complexity.

5.4 Ablation Results

Model Setup	**20 News**		Speed (s)	Buffer
	AA ↑	**AF ↓**		
Number of context samples				
0 sample	0.430	0.349	17588	1000
1 sample	0.456	0.348	17087	1000
2 samples	0.401	0.284	18517	1000
Training empty context probability				
0	0.456	0.348	17150	1000
0.3	0.469	0.322	17379	1000
0.7	0.432	0.375	17091	1000
Testing context				
Empty	0.467	0.308	17279	1000
Test natch	0.469	0.322	17379	1000
Saved Train	0.459	0.331	17814	1000

Table 2: Ablation study

Table 2 shows the ablation studies performed on the model. We present ablation studies across 3 different setups: 1. number of context samples provided during training. 2. probability which controls which sample is provided with an empty context instead of a context sample, and 3. context samples provided during testing It is important to note that the best setting was carried forward to the next ablation setup in the table.
For the number of context samples, 0 sample corresponds to a randomly generated tensor given as context to the training sample whereas 1 sample and 2 samples denote the number of previous context samples concatenated to the training sample. The setup with 1 context sample performed the best with an accuracy of 45.57%. The next component is replacing the context with an empty context to study if the improved performance is from the additional architecture or the information from the context. 0 probability corresponds to no replacement, whereas 0.3 and 0.7 correspond to 30% and 70% probabilities of the context sample being a zero tensor for a training example. Out of these the 0.3 context probability performed the best and gave an accuracy of 46.93%.

Finally, the testing context provided to the model was examined with empty context, test batch context, and saved train batch sample context. In empty context, each testing sample receives a zero tensor as a context whereas in the test batch and saved train batch, each testing sample is appended with its neighboring sample as context or a saved sample during training as context respectively. This led to the test batch performing the best with a 46.93%

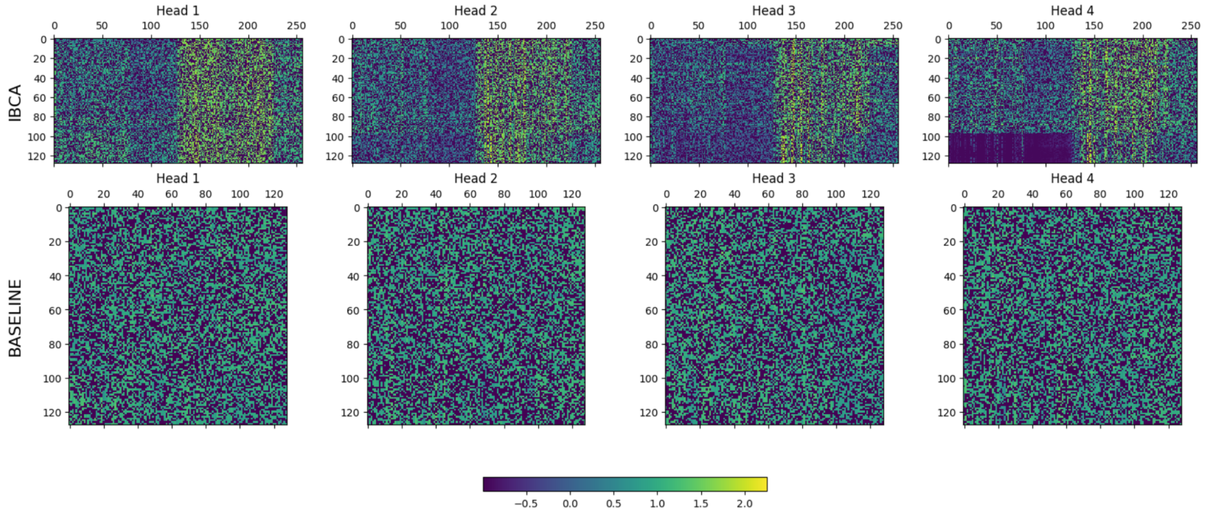

Figure 2: Row 1 depicts the Attention Map across the 4 heads generated from the IBCA Model. Row 2 depicts the Attention Map across the 4 heads generated from the Baseline Model

accuracy and 32.20% average forgetting.

The best performance and efficiency trade-off has the setting of 1 context sample, 0.3 zero ratio, and the inter-batch samples from the text batch. This is also the setting we use for the main experiments.

5.5 Analytic Results

Figure 2 provides a comparative analysis between the attention maps generated from the baseline model and the IBCA model. Both the models have the same hyperparameters. As proposed in our methodology, the target sample in IBCA is concatenated with additional context, effectively doubling its sequence length. The attention maps are obtained from the first layer for the respective models; it is interesting to note that this attention map is from a test sample which is misclassified by the baseline model but correctly classified by the IBCA model. Comparing the attention maps across each of the individual heads, we can a relatively consistent attention in the plots generated from the baseline mode, whereas the attention map from the IBCA model exhibits a completely different behavior. The attention map from the target samples has higher values resulting in a brighter shade, whereas the attention from the additional context is lower in value but with distinct patterns. This indicates that the context information is indeed passed to the next stage of target image processing. Furthermore, it can be observed that IBCA at the first layer, can distinguish apart the padding tokens in the samples, while the baseline model fails to do so. This indicates the IBCA can capture a general feature better than the baseline model. Finally, we can conclude that IBCA assists in the main training objective, by providing additional guidance on forming high-level features at the initial levels of a multi-level transformer based architecture.

6 Conclusion

We present our novel, yet simple, Inter-Batch Cross-Attention (IBCA) technique to tackle the enduring problem of catastrophic forgetting problem. With small computing overheads, it presents a viable approach to mitigate the catastrophic forgetting problem, even though it might not outperform the most advanced approaches' performance benchmarks. Our technology offers a lightweight, portable, and flexible way to support CL efforts. In a world where "Compute is king". Our experimental findings support the effectiveness of IBCA in knowledge preservation over time, showing an average 2% improvement in accuracy and a minimum 2% decrease in forgetting rate. IBCA offers a solution by balancing resource savings with performance improvement. IBCA provides a valuable tool paving the way for future research to further optimize and expand its application in diverse AI systems.

7 Limitations

In this study, we only explored the class-incremental learning setup, but our method can be easily applied to domain and task incremental settings. Our models are trained with limited computing resources, thus the performance presented might not reflect full convergence.

More experiments and compatibility with other state-of-the-art continual learning strategies can be explored. Future research with access to larger computational resources could investigate the scalability and efficiency of handling larger datasets or pretrained models. In its current form, this method is not sufficient to be used in real-life applications that can achieve significant good results. However, we believe our method can be an inspiration for future works to focus on a more human-like learning experience, rather than treating catastrophic forgetting as an engineering problem. We believe a more general framework like the one we present in the paper is of greater long-term impact on the development of human-level artificial general intelligence.

References

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. 2018. Efficient lifelong learning with a-gem. *ArXiv*, abs/1812.00420.

Elif Ceren Gok, Murat Onur Yildirim, Mert Kilickaya, and Joaquin Vanschoren. 2023. Adaptive regularization for class-incremental learning. *ArXiv*, abs/2303.13113.

Wenpeng Hu, Qi Qin, Mengyu Wang, Jinwen Ma, and Bing Liu. 2021. Continual learning by using information of each class holistically. In *AAAI Conference on Artificial Intelligence*.

Yanfei Hui, Jianzong Wang, Ning Cheng, Fengying Yu, Tianbo Wu, and Jing Xiao. 2021. Joint intent detection and slot filling based on continual learning model.

Zixuan Ke, Haowei Lin, Yijia Shao, Hu Xu, Lei Shu, and Bin Liu. 2022. Continual training of language models for few-shot learning. *ArXiv*, abs/2210.05549.

Zixuan Ke, Bing Liu, Nianzu Ma, Hu Xu, and Lei Shu. 2021. Achieving forgetting prevention and knowledge transfer in continual learning. *ArXiv*, abs/2112.02706.

Gyuhak Kim, Changnan Xiao, Tatsuya Konishi, Zixuan Ke, and Bin Liu. 2022. A theoretical study on solving continual learning. *ArXiv*, abs/2211.02633.

James Kirkpatrick, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2016. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114:3521 – 3526.

Brenden M. Lake and Marco Baroni. 2023. Human-like systematic generalization through a meta-learning neural network. *Nature*, 623(7985):115–121.

Zhizhong Li and Derek Hoiem. 2016. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:2935–2947.

Andrea Madotto, Zhaojiang Lin, Zhenpeng Zhou, Seungwhan Moon, Paul A. Crook, Bing Liu, Zhou Yu, Eunjoon Cho, and Zhiguang Wang. 2020. Continual learning in task-oriented dialogue systems. In *Conference on Empirical Methods in Natural Language Processing*.

Fei Mi, Liangwei Chen, Mengjie Zhao, Minlie Huang, and Boi Faltings. 2020. Continual learning for natural language generation in task-oriented dialog systems. pages 3461–3474.

Yijia Shao, Yiduo Guo, Dongyan Zhao, and Bin Liu. 2023. Class-incremental learning based on label generation. In *Annual Meeting of the Association for Computational Linguistics*.

Yifan Song, Peiyi Wang, Dawei Zhu, Tianyu Liu, Zhifang Sui, and Sujian Li. 2023. Repcl: Exploring effective representation for continual text classification.

Pablo Sprechmann, Siddhant M. Jayakumar, Jack W. Rae, Alexander Pritzel, Adrià Puigdomènech Badia, Benigno Uria, Oriol Vinyals, Demis Hassabis, Razvan Pascanu, and Charles Blundell. 2018. Memory-based parameter adaptation. *ArXiv*, abs/1802.10542.

Fan-Keng Sun, Cheng-Hao Ho, and Hung yi Lee. 2019. Lamol: Language modeling for lifelong language learning. In *International Conference on Learning Representations*.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer G. Dy, and Tomas Pfister. 2021. Learning to prompt for continual learning. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 139–149.

Zihan Wang, Jiayu Xiao, Mengxiang Li, Zhongjiang He, Yongxiang Li, Chao Wang, and Shuangyong Song. 2024. Towards robustness and diversity: Continual learning in dialog generation with text-mixup and batch nuclear-norm maximization.