

On Robust Incremental Learning over Many Multilingual Steps

Karan Praharaj and Irina Matveeva

Reveal

Chicago, IL

{kpraharaj, imatveeva}@revealddata.com

Abstract

Recent work in incremental learning has introduced diverse approaches to tackle catastrophic forgetting from data augmentation to optimized training regimes. However, most of them focus on very few training steps. We propose a method for robust incremental learning over dozens of fine-tuning steps using data from a variety of languages. We show that a combination of data-augmentation and an optimized training regime allows us to continue improving the model even for as many as fifty training steps. Crucially, our augmentation strategy does not require retaining access to previous training data and is suitable in scenarios with privacy constraints.

1 Introduction

Incremental learning is a common scenario for practical applications of deep language models. In such applications, training data is expected to arrive in batches rather than all at once, and so incremental perturbations to the model are preferred over retraining the model from scratch every time new training data becomes available for efficiency of time and computational resources. When multilingual models are deployed in applications, they are expected to deliver good performance over data across multiple languages and domains. This is why it is desirable that the model keeps acquiring new knowledge from incoming training data in different languages, while preserving its ability on languages that were trained in the past. The model should ideally keep improving over time, or at the very least not deteriorate its performance on certain languages through the incremental learning lifecycle.

It is known that incremental fine-tuning with data in different languages leads to catastrophic forgetting (French, 1999; McCloskey and Cohen, 1989) of languages that were fine-tuned in the past (Liu et al., 2021b; Vu et al., 2022). This means that

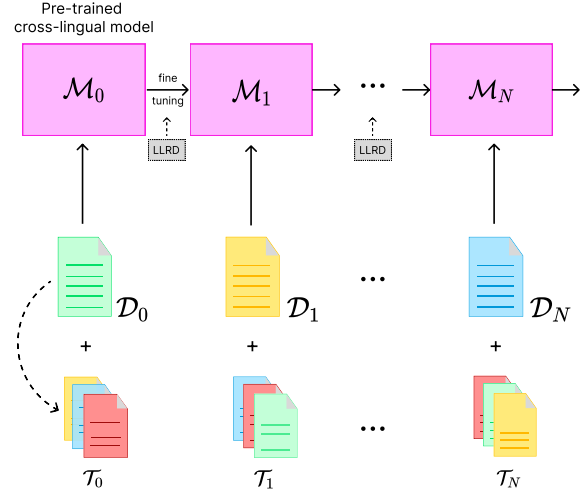


Figure 1: Translation augmented sequential fine-tuning approach with LLRD-enabled training. We begin with a pre-trained multilingual model \mathcal{M}_0 and fine-tune it over multiple stages to obtain $(\mathcal{M}_i$ where $i = 0 \dots N$). At each fine-tuning stage, we train the model \mathcal{M}_i over examples from the training set that is available at that stage (\mathcal{D}_i) which is in language \mathcal{L}_j . At each step, we sample a small random subset from \mathcal{D}_i and translate that sample into languages $(\mathcal{L} \setminus \mathcal{L}_j)$ to create a set of translated examples \mathcal{T}_i . Each step of training includes LLRD as a hyperparameter.

the performance on previously fine-tuned tasks or languages decreases after training on a new task or language. Multiple strategies have been proposed to mitigate catastrophic forgetting. Data-focused strategies such as augmentation and episodic memories (Hayes et al., 2019; Chaudhry et al., 2019b; Lopez-Paz and Ranzato, 2017), entail maintaining a cache of a subset of examples from previous training data, which are mixed with new examples from the current training data. The network is subsequently fine-tuned over this mixture as a whole, in order to help the model "refresh" its "memory" of prior information so that it can leverage previous experience to transfer knowledge to future tasks.

Closely related to our current work is the work

by M’hamdi et al. (2022); Ozler et al. (2020) of understanding the effect of incrementally fine-tuning models with multi-lingual data. They suggest that joint fine-tuning is the best way to mitigate the tendency of cross-lingual language models to erase previously acquired knowledge. In other words, their results show that joint fine-tuning should be used instead of incremental fine-tuning, if possible.

Optimization focused strategies such as Mirzadeh et al. (2020); Kirkpatrick et al. (2017) focus on the training regime, and show that techniques such as dropout, large learning rates with decay and shrinking the batch size can create training regimes that result in more stable models.

Translation augmentation has been shown to be an effective technique for improving performance as well. Wang et al. (2018); Fadaee et al. (2017); Liu et al. (2021a) and Xia et al. (2019) use various types of translation augmentation strategies and show substantial improvements in performance. Encouraged by these gains, we incorporate translation as our data augmentation strategy.

In our analysis, we consider an additional constraint that affects our choice of data augmentation strategies. This constraint is that the data that has already been used for training cannot be accessed again in a future time step. We know that privacy is an important consideration for continuously deployed models in corporate applications and similar scenarios and privacy protocols often limit access of each tranche of additional fine-tuning data only to the current training time step. Under such constraints, joint fine-tuning or maintaining a cache like Chaudhry et al. (2019a); Lopez-Paz and Ranzato (2017) is infeasible. Thus, we use translation augmentation as a way to improve cross-lingual generalization over a large number of fine-tuning steps without storing previous data.

In this paper we present a novel translation-augmented sequential fine-tuning approach that mixes in translated data at each step of sequential fine-tuning and makes use of a special training regime. Our approach shows minimization of the effects of catastrophic forgetting, and the interference between languages. The results show that for incremental learning over dozens of training steps, the baseline approaches result in catastrophic forgetting. We see that it may take multiple steps to reach this point, but the performance eventually collapses.

The main contribution of our work is combin-

ing data augmentation with adjustments in training regime and evaluating this approach over a sequence of 50 incremental fine-tuning steps. The training regime makes sure that incremental fine-tuning of models using translation augmentation is robust without the access to previous data. We show that our model delivers a good performance as it surpasses the baseline across multiple evaluation metrics. To the best of our knowledge, this is the first work to provide a multi-stage cross-lingual analysis of incremental learning over a large number of fine-tuning steps with recurrence of languages.

2 Related Work

Current work fits into the area of incremental learning in cross-lingual settings. M’hamdi et al. (2022) is the closest work to our research. The authors compare several cross-lingual incremental learning methods and provide evaluation measures for model quality after each sequential fine-tuning step. They show that combining the data from all languages and fine-tuning the model jointly is more beneficial than sequential fine-tuning on each language individually. We use some of their evaluation protocols but we have different constraints: we do not keep the data from previous sequential fine-tuning steps and we do not control the sequence of languages. In addition, they considered only six hops of incremental fine-tuning whereas we are interested in dozens of steps. Ozler et al. (2020) do not perform a cross-lingual analysis, but study a scenario closely related to our work. Their findings fall in line with those of M’hamdi et al. (2022) as they show that combining data from different domains into one training set for fine-tuning performs better than fine-tuning each domain separately. However, this type of joint fine-tuning is ruled out for our scenario where we assume that access to previous training data is not available, and so we focus on sequential fine-tuning exclusively.

Mirzadeh et al. (2020) study the impact of various training regimes on forgetting mitigation. Their study focuses on learning rates, batch size, regularization method. This work, like ours, shows that applying a learning rate decay plays a significant role in reducing catastrophic forgetting. However, it is important to point out that our type of decay is different from theirs. Mirzadeh et al. (2020) start with a high initial learning rate for the first task to obtain a wide and stable minima. Then, for each

subsequent task, slightly decrease the learning rate, while simultaneously reducing the batch size, as recommended by [Smith et al. \(2017\)](#). On the other hand, we apply our decay rate across the transformer model’s layer stack so that the deviations from the current optimum get progressively smaller as one moves down the layers and we do this at each step of incremental fine-tuning.

Memory-based approaches such as [Chaudhry et al. \(2019b\)](#); [Lopez-Paz and Ranzato \(2017\)](#) have been explored to mitigate forgetting. Such methods make use of an *episodic memory* or a cache which stores a subset of data from previous tasks. These examples are then used for training along with the current examples in the current optimization step. Similarly, [Xu et al. \(2021\)](#) suggest a gradual fine-tuning approach, wherein models are eased towards the target domain by increasing the concentration of in-domain data at every fine-tuning stage. This work builds on the findings from [Bengio et al. \(2009\)](#), who show that a multi-stage curriculum strategy of learning easier examples first, and gradually increasing the difficulty level leads to better generalization and faster convergence. While we cannot maintain a cache of this sort because of our constraints, we take inspiration from this line of research and generate “easier examples” using translation in languages that are expected to appear in our data.

Sequential fine-tuning of languages has not been extensively studied for long sequences. [Liu et al. \(2021b\)](#) and [Garcia et al. \(2021\)](#) go up to two stages, whereas [M’hamdi et al. \(2022\)](#) go upto six stages. We provide an analysis of a much longer fine-tuning sequence with fifty stages. We are also the first to present an analysis of sequences with repetition of languages.

3 Method

We propose a translation augmented sequential fine-tuning approach for incremental learning in a cross-lingual setting. Our approach addresses the scenario in which a pre-trained model is incrementally fine-tuned over dozens of steps without access to previously seen training data. There is a set of languages $\mathcal{L} = \mathcal{L}_0, \dots, \mathcal{L}_K$ that can appear during the incremental fine-tuning steps and we assume that in each step the data comes from only one language. We exploit the benefits of data augmentation, as well as specialized optimization techniques.

We begin with a pre-trained multilingual model

\mathcal{M}_0 which will be fine-tuned over multiple stages to create incremental versions \mathcal{M}_i where $i = 0 \dots N$. The training data in each incremental fine-tuning step is \mathcal{D}_i and is in a randomly selected language \mathcal{L}_j , where $0 \leq j \leq K$. At each step, we sample a small random subset \mathcal{T}_i from \mathcal{D}_i and translate that subset to all languages from \mathcal{L} except \mathcal{L}_j , to create multiple additional subsets of training data \mathcal{T}_i . We denote the augmented training set as \mathcal{D}_i^T , where

$$\mathcal{D}_i^T = \mathcal{D}_i + \mathcal{T}_i$$

Figure 1 provides a graphical representation of our approach.

3.1 Fine-tuning regime with LLRD

Motivated by [Yosinski et al. \(2014\)](#), we apply a layer-wise learning rate decay (or LLRD, denoted by ζ) to the model parameters depending on their position in the layer stack of the model, based on the discriminative fine-tuning method proposed by [Howard and Ruder \(2018\)](#). Layer-wise Learning Rate Decay (LLRD) is a method that applies higher learning rates for top layers and lower learning rates for bottom layers. The goal is to modify the lower layers that encode more general information less than the top layers that are more specific to the pre-training task. This is accomplished by setting the learning rate of the top layer and using a multiplicative decay rate to decrease the learning rate layer-by-layer from top to bottom. We split the parameters θ into $\{\theta^1, \dots, \theta^L\}$ where θ^l contains the parameters of the l^{th} layer of the pre-trained model. The parameters are updated as follows:

$$\theta_t^l = \theta_{t-1}^l - \eta^l \cdot \nabla_{\theta} J(\theta)$$

where η^l represents the learning rate of the l^{th} layer. We set the learning rate of the top layer to η^l and use

$$\eta^{k-1} = \zeta \cdot \eta^k$$

4 Experiments

4.1 Data

We use the Multilingual Amazon Reviews corpus (MARC) ([Keung et al., 2020](#)). This dataset is a large-scale collection of product reviews from 6 different languages and from 31 different categories. We construct our training sets by extracting reviews for ten common categories: *apparel*, *automotive*,

beauty, drugstore, grocery, home, kitchen, musical instruments, sports, wireless. The number of reviews for each language-category combination are not equal, but to ensure consistency of training examples at each training step, we create two unique training sets of size 100 and 150 for each language-category combination. For our experiments, we use reviews from all 6 languages provided in the dataset (Chinese, English, French, German, Japanese and Spanish). We drop the 3-star reviews and bifurcate the rest into two class labels: positive (4-star and 5-star) and negative (1-star and 2-star) sentiment.¹

Each incremental training set \mathcal{D}_i contains 100 reviews from a particular language-category combination, for example, *de-grocery*. To ensure class balancing, we sample an equal number of positive and negative records for each training set.

We use the original test-splits for each of the 60 language-category combinations of the MARC data as our test set.

4.2 Translation augmentation

The translations were generated using the Google Translate API. In the current work we sample a fraction of 0.1 of the training examples. For example, if the training data is \mathcal{D}_i has 100 records, the translation augmented data \mathcal{D}_i^T will have 150 records.

4.3 Constructing the sequence

We tested our approach on a large number of incremental fine-tuning steps using data from various language-category combinations. To do that we created 3 random sequences $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$ with 50 training sets $\mathcal{D}_1\text{-}\mathcal{D}_{50}$ each. We term each incremental fine-tuning step as a *hop*. Multiple hops comprise a *sequence*.

Each training set \mathcal{D}_i contains data from a particular language-category combination. To construct these 3 randomized sequences we used the following approach. We first generated all possible language-category combinations, and then sampled one combination at a time for each hop. The only constraint placed on the sampling is that it cannot choose a combination that has already appeared in the sequence. However, the same language with a different category and the same category with

a different language still can occur. E.g., if *de-grocery* features once in the sequence, it cannot be repeated, even though *de-sports* or *en-grocery* are possible options. The plots in Fig. 2 and 3 show the language-category combination for each \mathcal{D}_i in all three sequences.

4.4 Model and training

We use multilingual BERT (cased) as our base model. We run a hyperparameter optimization over a relatively small search space containing values that were most effective in our preliminary experiments. Two types of settings were used for training:

- Default settings:
 - Epochs: 5
 - Learning rate: $2e - 5$
 - LLRD: 1.0
- LLRD-enabled settings:
 - Epochs: 5
 - Learning rate: $2e - 5$
 - LLRD: 0.75, 0.85²

The best checkpoint from any given stage is chosen for subsequent fine-tuning over the next language dataset. At the first stage, we use the pre-trained mBERT checkpoints released by (Devlin et al., 2019).³ All experiments have been run on a single machine with a 6-core NVIDIA Tesla K80 GPU.

Train data ↓	$\zeta = 0.38$	$\zeta = 0.5$	$\zeta = 0.75$	$\zeta = 0.85$	$\zeta = 0.95$	$\zeta = 1.0$
de-100	0.70	0.73	0.76	0.75	0.75	0.69
en-100	0.65	0.64	0.73	0.69	0.65	0.66
fr-100	0.71	0.72	0.73	0.73	0.72	0.69
jp-100	0.62	0.63	0.65	0.67	0.63	0.33
zh-100	0.46	0.57	0.69	0.65	0.59	0.57
es-100	0.70	0.73	0.76	0.75	0.69	0.63

Table 1: Comparison of different LLRD settings (ζ). We observe that $\zeta = 0.75$ and $\zeta = 0.85$ deliver the most consistent performance.

¹We ensure that both final labels contain an equal number of examples of their constituent star-ratings. E.g., the negative sentiment class will contain an equal number of reviews from 1- and 2-star reviews.

²Our preliminary experiments showed that LLRD values of $\zeta = 0.75$ and $\zeta = 0.85$ are the most suitable candidates for our scenario. We show in brief the results of other values for comparison in Table 1. The models are trained over the *sports* datasets of all six languages and compare the scores averaged over the 60 evaluation sets. $\zeta = 0.75$ and $\zeta = 0.85$ deliver the most consistent performance.

³github.com/google-research/bert/blob/master/multilingual.md

4.5 Experimental setup

We show the results with the following four variations. We start with the default incremental fine-tuning approach and add modifications such as translation augmentation, LLRD and the combination of translation and LLRD.

- *Sequential fine-tuning* (SEQFT): Data is \mathcal{D}_i , default training settings.
- *Sequential fine-tuning with LLRD* (SEQFT-LLRD): Data is \mathcal{D}_i , Trained with LLRD-enabled settings.
- *Translation augmented sequential fine-tuning* (SEQFT-TRANS): Data is \mathcal{D}_i^T , default training settings.
- *Translation augmented sequential fine-tuning using LLRD* (SEQFT-TRANS-LLRD): This is our approach. Data is \mathcal{D}_i^T , Trained with LLRD-enabled settings.

4.6 Evaluation Metrics

We evaluate our proposed approach against the baseline models on overall F_1 scores over the following metrics:

- **Average hop-wise F_1 :** The F_1 scores over each of the 60 test sets are averaged for every single fine-tuning hop.
- **Overall F_1 :** The averages of hop-wise F_1 scores for all stages are averaged to give the overall performance.
- **Forgetting (F):** The average forgetting across languages at the end of sequential fine-tuning. This evaluation metric measures the decrease in performance on each of the languages between the peak F_1 score and the F_1 score after final training step of the sequence. We evaluate forgetting by language (F-lang) as well as by category (F-categ).
- **In-language, in-domain performance (IL/ID):** These are the average scores on all the test sets corresponding to the last fine-tuned language-category combination. For example, if the current stage of fine-tuning uses Chinese *zh-grocery* data, then the in-language performance is the F_1 over the *zh-grocery* test set.

- **Out-of-language, in-domain performance (OL/ID):** These are the average scores on all the test sets corresponding to languages that were *not* seen in the previous stage of fine-tuning but are of the same domain. For example, if the current stage of fine-tuning uses *zh-grocery* data, the test sets used to calculate OL/ID performance are English (*en-grocery*), French (*fr-grocery*), German (*de-grocery*), Japanese (*jp-grocery*) and Spanish (*es-grocery*).
- **In-language, out-of-domain performance (IL/OD):** The average scores on test sets of the same language as training but corresponding to the domains that were *not* used during training. For example, if the current stage of fine-tuning uses *zh-grocery* data, the performance on the Chinese test sets of all domains other than *grocery* are averaged at each fine-tuning stage.
- **Out-of-language, out-of-domain performance (OL/OD):** The average scores on the test sets corresponding to all language-category combinations except the one that was used during training.

5 Results

We present below a comparison of our approach SEQFT-TRANS-LLRD with different variations of sequential fine-tuning in our results.

5.1 SEQFT (baseline)

Our proposed approach SEQFT-TRANS-LLRD outperforms SEQFT decisively. We see that it is able to dramatically improve the overall F_1 performance and reduce forgetting on both forgetting metrics by one order of magnitude. This is evident from Fig. 2 and Table 2. We see in the plots for average F_1 (Fig. 2) that for each of the three sequences, the default approach SEQFT results in catastrophic forgetting. It can happen at different hops. In sequence 1, at the 2nd hop, in sequence 2, at the 17th hop and in sequence 3, at the 23rd hop. But eventually, the F_1 drops and never recovers. This highlights the importance of studying sequential fine-tuning over a large number of steps to be able to observe these effects. After the model performance collapses, we observed that the model classifies almost every example as negative. It is not clear from our results in these three different sequences if a particular

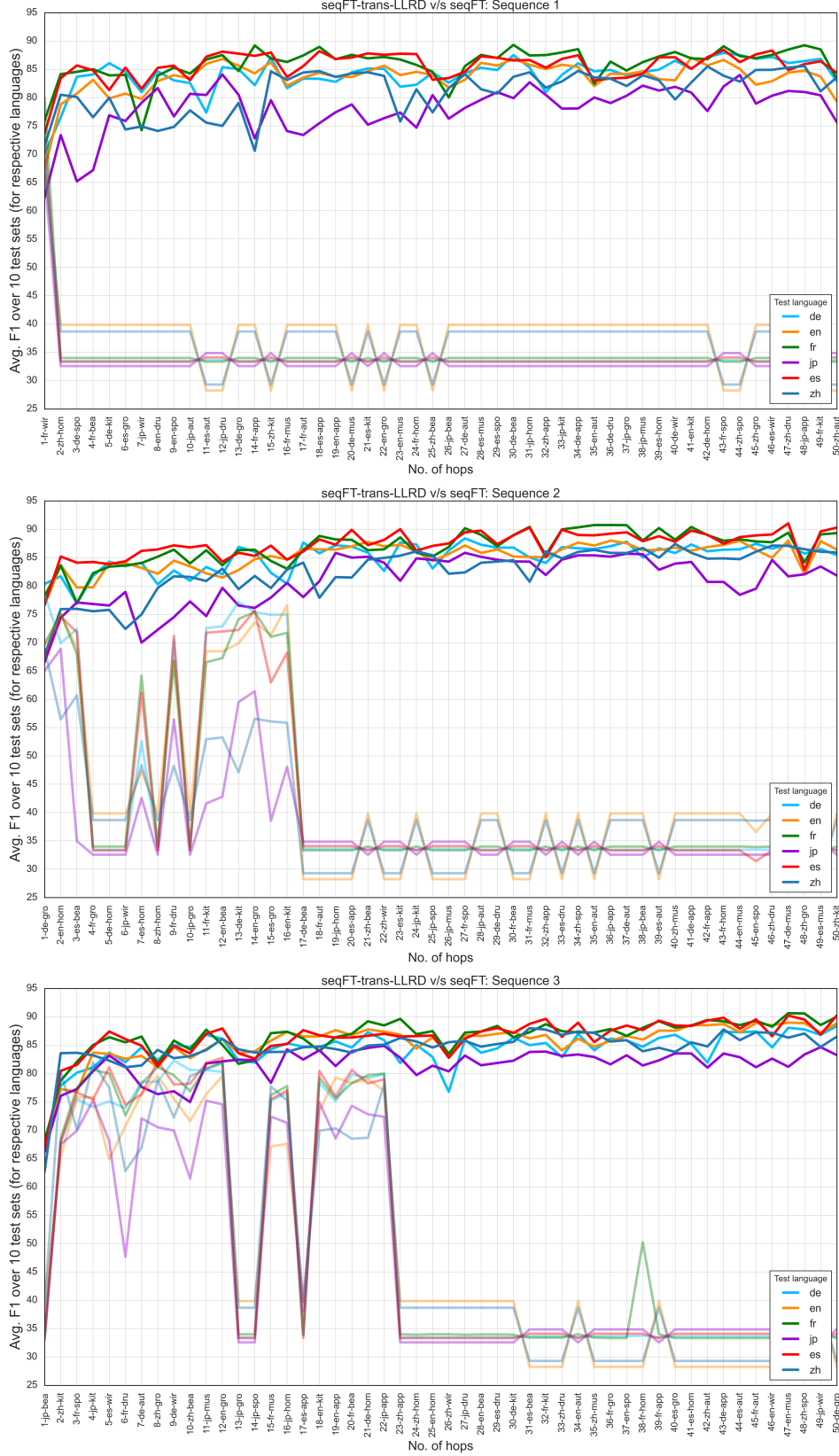


Figure 2: SEQFT vs SEQFT-TRANS-LLRD: We show the plots of hop-wise F_1 scores for each randomized sequence of 50 hops each. Each plot has the details for one sequence. We show the F_1 for each language separately in color-coded lines. The translucent lines show the results for our baseline of no data augmentation and default fine-tuning settings. The regular lines show the results of our approach. The x-axis shows the language-category combination in each training set D_i .

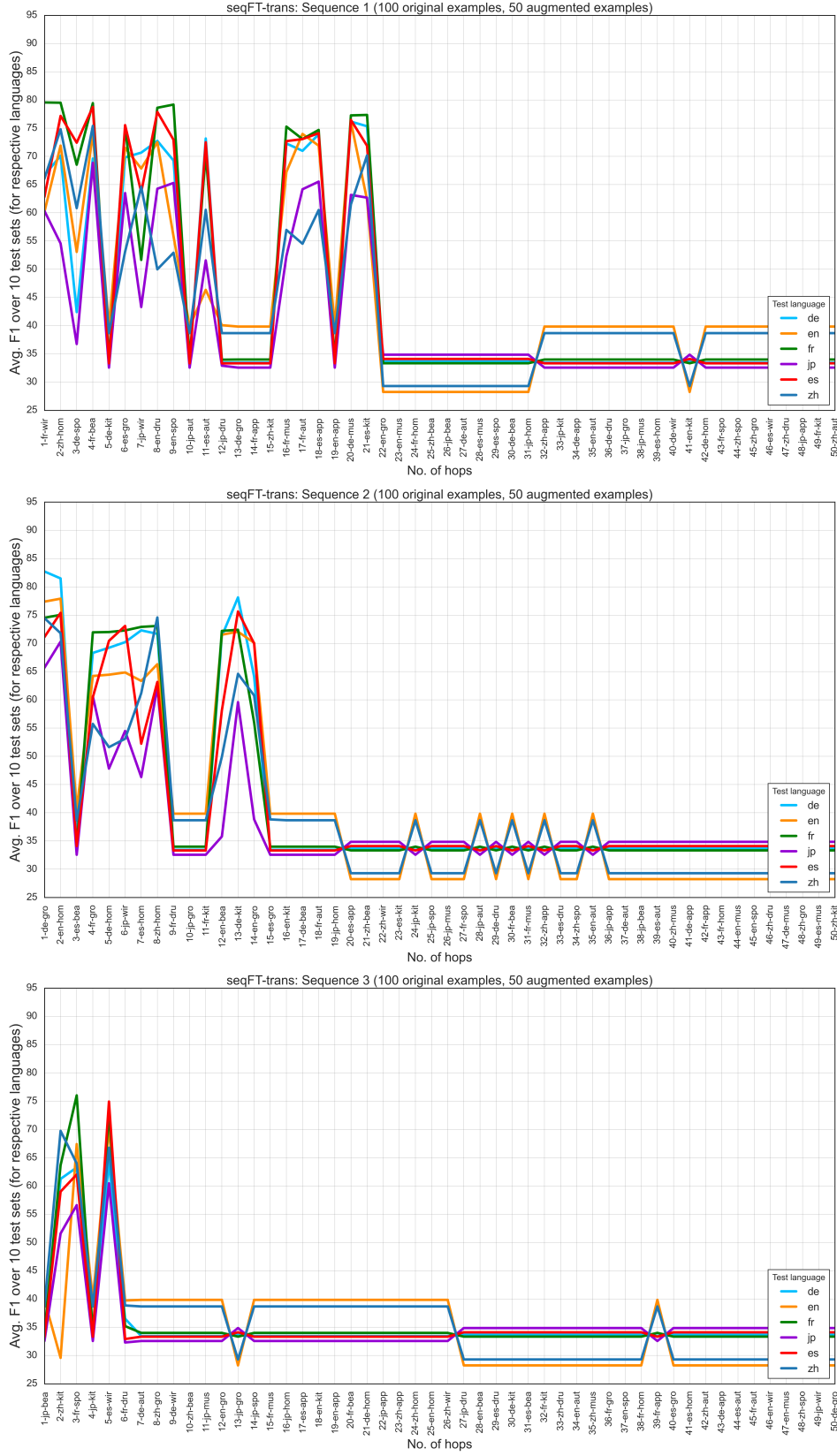


Figure 3: SEQFT-TRANS: We show the plots of hop-wise F_1 scores for each randomized sequence of 50 hops each. Each plot has the details for one sequence. We show the F_1 for each language separately in color-coded lines. The x-axis shows the language-category combination in each training set D_i .

Method	Size	Seq	Overall F_1	IL/ID	OL/OD	IL/OD	OL/ID	F-lang	F-categ
SEQFT	100	1	35.36	35.96	35.35	35.11	35.34	39.32	39.32
SEQFT-LLRD	100		85.44	85.70	85.44	85.94	85.24	2.73	2.66
SEQFT-TRANS	100+50		43.54	43.18	43.55	43.91	42.74	40.50	39.21
SEQFT-TRANS-LLRD	100+50		83.05	84.86	83.02	84.31	82.93	1.54	1.53
SEQFT	100	2	40.71	40.86	40.71	42.39	40.46	38.70	35.83
SEQFT-LLRD	100		85.32	86.39	85.30	86.03	85.09	1.37	1.33
SEQFT-TRANS	100+50		39.94	39.21	39.95	40.55	39.74	43.78	43.37
SEQFT-TRANS-LLRD	100+50		84.67	86.45	84.64	86.00	84.62	2.04	1.75
SEQFT	100	3	48.74	49.73	48.72	49.58	48.65	48.37	48.10
SEQFT-LLRD	100		84.49	86.04	84.46	85.44	83.99	1.31	1.24
SEQFT-TRANS	100+50		35.51	35.18	35.52	36.15	35.39	37.04	36.02
SEQFT-TRANS-LLRD	100+50		84.74	86.02	84.71	85.74	84.42	0.65	0.82

Table 2: Summary of results of our approach in comparison to the baselines.

language or category or combination triggers this collapse in performance. This is something we intend to explore in future work. Even in the initial hops before the collapse in performance, SEQFT under-performs our approach. From Table 2, we see that our approach outperforms this baseline on all metrics. There is at least a 36 point improvement on the overall F_1 score between SEQFT and our approach.

In subsections 5.2 and 5.3, we study the effects of the translation augmentation and specialized training regime separately to understand their contributions in isolation.

5.2 SEQFT-TRANS (baseline)

We observe that translation augmentation on its own performs very similarly to the baseline SEQFT. It outperforms the baseline only on one sequence in terms of overall F_1 . The overall F_1 for SEQFT-TRANS is significantly lower compared to our approach. The plots look similar to SEQFT, but we still provide them in Fig. 3. Augmentation seems to delay catastrophic forgetting until 6, 15 and 22 hops. However, both approaches eventually result in catastrophic forgetting. Thus, the performance at the end of the sequence is extremely low. This is reflected in the high values of the F-lang and F-categ metrics in Table 2.

5.3 SEQFT-LLRD

In contrast to SEQFT-TRANS, using only the specialized training regime SEQFT-LLRD shows a strong performance. In fact, it appears that the main advantage of our approach stems from the optimized training regime with LLRD since SEQFT-LLRD and SEQFT-TRANS-LLRD have comparable performance on many evaluation metrics. For se-

quence 3, our full approach SEQFT-TRANS-LLRD shows a slightly higher overall F_1 performance as compared to SEQFT-LLRD. But on the other two sequences SEQFT-LLRD has a higher F_1 score. However, in terms of the forgetting metrics, our approaches outperforms SEQFT-LLRD on two out of three sequences. Also, for sequence 3, the out-of-domain performance with our full approach is higher.

In summary, our approach outperforms the baseline (SEQFT) by a wide margin. Since we use multiple language and category combinations, we show results on metrics based on similarity of the train and test data with respect to language and category. Our observations are consistent across all evaluation metrics. The main performance boost for our approach comes from including LLRD in the training regime. However, our combination of LLRD and translation augmentation slightly outperforms SEQFT-LLRD in terms of both forgetting metrics.

6 Conclusion

We introduce a sequential fine-tuning approach wherein the language data for fine-tuning is augmented by a subset of translated examples. Our augmentation strategy emulates episodic memory and decreases the reliance on a cache of stored examples from previous stages. We also advocate the use of layer-wise learning rate decay and illustrate its effectiveness in mitigating forgetting. With our results, we show that the proposed approach can outperform joint fine-tuning based methods, in spite of not having access to the complete set of examples from all languages. Crucially, it achieves robust and consistent performance over multiple

cross-lingual fine-tuning stages. The trajectories of performances over different languages suggest that the model can continue learning over new data (or languages) for even more stages in the sequence without undergoing a significant reduction in performance. Furthermore, our approach surpasses all baselines when evaluated on in-domain, out-of-domain, in-language and out-of-language performance, showing that the model has a strong generalization ability. All-in-all, we hope our work provides encouragement to the community to pursue similar recipes that facilitate long-term continual learning.

7 Limitations

One of the primary limitations of our work is that the analysis has only been provided for a single random sequence with only six languages. Diversification of this study with more languages, more random sequences and an even higher number of fine-tuning stages is a strong avenue for future work that we intend to pursue. Additionally, we would also like to extend this study to other cross-lingual tasks to see if the findings are similar.

Another limitation is a lack of experimentation with adapter-based methods. In the future, we would also like to experiment with varying proportions of translated examples with respect to the original training size.

We would also like to extend our work to include a more in-depth study of the underlying linguistic factors that underpin cross-lingual transfer or forgetting. A study of this kind would ideally include, but not be limited to, analyses based on word order, scripts, morphology and syntax.

Acknowledgements

The authors of this work would like to express their gratitude to Dinesh Karamchandani, for help with setting up the experimentation framework, and Dan Roth for feedback on our approach.

References

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. [Curriculum learning](#). In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09*, page 41–48, New York, NY, USA. Association for Computing Machinery.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K. Dokania,

Philip H. S. Torr, and Marc’ Aurelio Ranzato. 2019a. [On Tiny Episodic Memories in Continual Learning](#). Number: arXiv:1902.10486 arXiv:1902.10486 [cs, stat].

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet Kumar Dokania, Philip H. S. Torr, and Marc’ Aurelio Ranzato. 2019b. [Continual learning with tiny episodic memories](#). *CoRR*, abs/1902.10486.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2017. [Data augmentation for low-resource neural machine translation](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics.

Robert French. 1999. [Catastrophic forgetting in connectionist networks](#). *Trends in cognitive sciences*, 3:128–135.

Xavier Garcia, Noah Constant, Ankur Parikh, and Orhan Firat. 2021. [Towards Continual Learning for Multilingual Machine Translation via Vocabulary Substitution](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1184–1192, Online. Association for Computational Linguistics.

Tyler L. Hayes, Kushal Kafle, Robik Shrestha, Manoj Acharya, and Christopher Kanan. 2019. [Remind your neural network to prevent catastrophic forgetting](#).

Jeremy Howard and Sebastian Ruder. 2018. [Universal language model fine-tuning for text classification](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. 2020. [The multilingual Amazon reviews corpus](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4563–4568, Online. Association for Computational Linguistics.

James Kirkpatrick, Razvan Pascanu, Neil 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.

2017. [Overcoming catastrophic forgetting in neural networks](#). *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.
- Qi Liu, Matt Kusner, and Phil Blunsom. 2021a. [Counterfactual data augmentation for neural machine translation](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 187–197, Online. Association for Computational Linguistics.
- Zihan Liu, Genta Indra Winata, Andrea Madotto, and Pascale Fung. 2021b. [Preserving Cross-Linguality of Pre-trained Models via Continual Learning](#). In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 64–71, Online. Association for Computational Linguistics.
- David Lopez-Paz and Marc’ Aurelio Ranzato. 2017. Gradient Episodic Memory for Continual Learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- Michael McCloskey and Neil J. Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. *The Psychology of Learning and Motivation*, 24:104–169.
- Meryem M’hamdi, Xiang Ren, and Jonathan May. 2022. [Cross-lingual Lifelong Learning](#). Number: arXiv:2205.11152 arXiv:2205.11152 [cs].
- Seyed Iman Mirzadeh, Mehrdad Farajtabar, Razvan Pascanu, and Hassan Ghasemzadeh. 2020. [Understanding the role of training regimes in continual learning](#).
- Kadir Bulut Ozler, Kate Kenski, Steve Rains, Yotam Shmargad, Kevin Coe, and Steven Bethard. 2020. [Fine-tuning for multi-domain and multi-label uncivil language detection](#). In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 28–33, Online. Association for Computational Linguistics.
- Samuel L. Smith, Pieter-Jan Kindermans, Chris Ying, and Quoc V. Le. 2017. [Don’t decay the learning rate, increase the batch size](#).
- Tu Vu, Aditya Barua, Brian Lester, Daniel Cer, Mohit Iyyer, and Noah Constant. 2022. [Overcoming catastrophic forgetting in zero-shot cross-lingual generation](#).
- Xinyi Wang, Hieu Pham, Zihang Dai, and Graham Neubig. 2018. [Switchout: an efficient data augmentation algorithm for neural machine translation](#).
- Mengzhou Xia, Xiang Kong, Antonios Anastasopoulos, and Graham Neubig. 2019. [Generalized data augmentation for low-resource translation](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5786–5796, Florence, Italy. Association for Computational Linguistics.
- Haoran Xu, Seth Ebner, Mahsa Yarmohammadi, Aaron Steven White, Benjamin Van Durme, and Kenton Murray. 2021. [Gradual fine-tuning for low-resource domain adaptation](#). In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 214–221, Kyiv, Ukraine. Association for Computational Linguistics.
- Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.