Extended Abstract

Motivation Continual learning in language models remains a challenge due to catastrophic forgetting, where learning new tasks erases previously learned knowledge. This issue is especially problematic in multilingual translation, where languages often share lexical features that exacerbate parameter interference. We explore RL methods to enable shared-parameter models to learn new translation tasks without retraining from scratch or losing prior capabilities. RL is appropriate for this scenario because it allows task-specific optimization while incorporating learning constraints, such as reward shaping or policy regularization, to retain prior behaviors.

Method We evaluate four methods for continual learning in translation from German and Dutch to English: sequential finetuning (SFT), experience replay (ER), reinforcement distillation (RD), and proximal policy optimization (PPO), all integrated with LoRA adapters for efficient finetuning. Each method reflects a different tradeoff between retention and adaptation, using techniques such as memory buffers, teacher-student alignment, or reward-based optimization. This is a novel approach because our combination of ER, LoRA, and other RL-based training methods have not been used, to the best of our knowledge, in continual multilingual translation tasks across semantically and lexically similar langues.

Implementation We use `SmolLM-360M-Instruct` HuggingFaceTB (2024) as our base model, finetuned on 35000 German and Dutch parallel examples from WMT. All models were trained with PEFT using LoRA adapters, and reinforcement-based methods (RDL and PPO + SFT) incorporated supervised warmup phases to improve stability when transitioning to Dutch. Evaluation was conducted using BLEU and METEOR on held-out test sets for both languages.

Results Sequential finetuning showed clear signs of catastrophic forgetting, with large drops in German performance after Dutch training. Reinforcement distillation preserved German accuracy best but hindered Dutch adaptation, while PPO improved semantic fidelity in some cases but suffered from reward instability. Experience replay offered a moderate balance, though its effectiveness was highly sensitive to the replay ratio and size of the dataset. With a ratio of $\rho = 0.9$ and a Dutch dataset size of $\phi = 30000$, ER achieved a 99% and 75% BLEU performance compared to our upper bound model.

Discussion Our experiments highlight distinct strengths and limitations across RL strategies: RDL strongly preserves prior knowledge but resists learning new tasks; PPO shows promise in preserving meaning but is unstable without careful warmup and reward shaping. ER offers a simpler tradeoff mechanism, but its adaptation capability depends heavily on the ratio of the different training data types. These findings suggest that no single approach strictly dominates in all scenarios.

Conclusion We present a proof of concept for using RL-based strategies to mitigate catastrophic forgetting in low-resource multilingual translation. While no approach achieved perfect retention and adaptation, each offered valuable insights into tradeoffs between stability and flexibility. Future work could explore more scalable models, denser reward signals, and continual task sequences to improve performance in lifelong learning scenarios.

RL Methods for Mitigating Catastrophic Forgetting in Continual SLM Translation

Abhijit Devalapura
Department of Computer Science
Stanford University
abhydev5@stanford.edu

Riley Carlson
Department of Computer Science
Stanford University
rileydc@stanford.edu

Abstract

Machine learning models often struggle to incorporate new tasks without retraining from scratch, especially in settings where previously-learned capabilities must be retained. This challenge, known as catastrophic forgetting, arises when learning on new data overwrites useful prior knowledge. In this work, we explore RL-based approaches for mitigating catastrophic forgetting in continual task learning, using multilingual translation as the proof of concept task. We experiment with a small language model (SmolLM) and evaluate four strategies: sequential finetuning, experience replay, reinforcement distillation, and proximal policy optimization (PPO). Each method represents a different mechanism for balancing retention of past knowledge with the acquisition of new skills. Our findings show that reinforcement distillation provides strong retention but limits adaptation, experience replay provides a data-sensitive balanced strategy, while PPO captures semantic shifts best but remains unstable. This study presents a proof of concept for using RL to support continual learning in shared-parameter models with implications for more general multitask and lifelong learning settings.

1 Introduction

It still remains an open problem in the field of natural language processing (NLP) to finetune a language model to translate new languages without catastrophic forgetting: the situation of learning new information at the direct expense of forgetting previously learned tasks. In this project, we address the underexplored problem of catastrophic forgetting in continual translation—specifically, when adapting a small language model (SLM) to in sequence translate two structurally similar languages, German and Dutch, to English in a low-resource setting.

In this project, we focus on formulating a novel problem: how to continually adapt a small language model (SLM) to translate multiple, semantically similar, low-resource languages using reinforcement learning (RL). Unlike prior work that assumes access to all task data at once or uses task-specific parameter modules, we explore how RL can enable a shared-parameter model to incorporate new languages sequentially while retaining prior knowledge. To our knowledge, this is the first work to explicitly frame low-resource multilingual continual machine translation as an RL problem. We propose and evaluate a suite of RL-based methods as an alternative to standard supervised finetuning.

Both German and Dutch belong to the West Germanic language branch of the Germanic language family, and so they share substantial syntactic and lexical similarities, both in grammar and in practice. These similarities, while beneficial for transfer learning, also increase the risk of parameter interference when a model is trained on one language and then finetuned on the other. In such cases, newly learned parameters optimized for Dutch, for example, may overwrite those optimized for German, leading to a substantial drop in performance on the earlier task. This challenge underscores

the need for continual learning strategies that allow a single model to retain prior knowledge while acquiring new capabilities.

A straightforward approach to multilingual continual learning is to finetune the model first on German, then on Dutch. However, this often results in catastrophic forgetting. A common remedy is to use task-specific Parameter Efficient FineTuning (PEFT) adapters, such as separate LoRA (Low-Rank Adaptation) modules for German and Dutch Hu et al. (2021). While effective, this approach lacks shared representations and becomes inefficient as the number of tasks grows. Another alternative is to aggregate data from all languages and finetune on the combined dataset. While this joint training yields strong performance, it assumes access to all language data simultaneously, ignores the incremental nature of real-world data collection, and requires retraining from scratch for each new task—making it computationally expensive and impractical at scale. These limitations motivate our goal: to develop methods that approach joint training performance while leveraging prior models in a sequential setting. In other words, our work serves as a proof of concept for integrating new translation tasks into existing models without retraining from scratch or requiring access to all past data.

This project is systematically applies RL methods to mitigate catastrophic forgetting in multilingual translation between semantically similar languages. While RL has been explored in NMT for quality improvements on single tasks (e.g., using BLEU-based rewards), it has rarely been used in the context of sequential multilingual adaptation. Moreover, most continual learning efforts in RL focus on control tasks, not on sequence generation.

Our hypothesis is that reinforcement learning offers a promising direction for continual translation because of its capacity to optimize for task-specific rewards while implicitly regularizing the model's behavior to preserve prior knowledge. We treat this as a sequential decision-making problem, where RL-based algorithms like reinforcement distillation and PPO Schulman et al. (2017) can in theory guide the model's policy toward optimal translation behaviors under limited feedback. Through reward signals (e.g., BLEU) and policy constraints (e.g., via KL divergence or clipping), we aim to achieve a balance between adaptation and retention.

We evaluate four strategies: (1) sequential finetuning (as a baseline), (2) experience replay (ER), (3) reinforcement distillation with a supervised warmup phase, and (4) proximal policy optimization (PPO) with a supervised warmup phase.

2 Related Work

Catastrophic forgetting, the tendency of models to overwrite prior knowledge when trained on new tasks, has long been recognized in continual learning research Kirkpatrick et al. (2017). Two primary categories of solutions have emerged: regularization-based and data-driven.

Regularization-based methods, such as Elastic Weight Consolidation (EWC), constrain parameter updates by penalizing deviation from previously important weights, as defined by the Fisher information matrix. While effective in supervised image classification and classical RL tasks, such methods are rarely applied in sequence generation, with only limited studies exploring their use in tasks like translation de Masson d'Autume et al. (2019)[1].

Data-driven approaches, such as Experience Replay (ER), attempt to retain performance by replaying examples from earlier tasks during training a new one. ER has shown success in vision and reinforcement learning domains Rolnick et al. (2019), and limited success in NLP tasks like dialogue modeling. However, its adaptation to continual translation of closely related languages—where lexical and grammatical overlaps exacerbate interference—is not well explored. Our use of ER to replay German examples while learning Dutch-English translation is, to our knowledge, a novel application.

In neural machine translation (NMT), RL has been primarily used to improve single-task generation quality. Norouzi et al. (2017) introduced Reward Augmented Maximum Likelihood (RAML) to optimize BLEU-based objectives. Subsequent work applied policy gradient methods (e.g., REINFORCE, actor-critic) to improve sequence-level metrics Bahdanau et al. (2017); Ranzato et al. (2016).

[1]We experimented with EWC combined with PPO, but this corrupted our model because of exploding weights despite using clipping.

However, these efforts focused solely on enhancing translation quality, not on retaining prior language capabilities or addressing catastrophic forgetting.

Continual reinforcement learning has been explored in vision and control settings, often via parameter isolation or replay Schwarz et al. (2018). Yet, applications to language generation remain extremely limited. We extend these ideas by incorporating reinforcement distillation and PPO into continual NMT, a setting that presents new challenges due to semantic similarity between tasks and sparse reward signals such as BLEU.

Finally, PEFT methods, such as LoRA Hu et al. (2021), offer a computationally viable way to adapt large models with minimal parameter updates. LoRA is increasingly used in NLP but has not been extensively tested in continual learning settings. We integrate LoRA into all of our methods to ensure consistency and efficiency under low-resource constraints.

In summary, while components of our approach—ER, LoRA, and RL-based training—have each been studied independently, our combination of these in the context of continual multilingual translation with semantically similar languages is novel, and we believe this offers valuable insights into mitigating catastrophic forgetting in practical, lower-resource scenarios.

3 Methods & Experimental Setup

Here we will provide a formal description of the methods we employed to attempt to show a mitigation of the catastrophic forgetting phenomenon in the setting of low-resource continual machine translation. We begin with curating our data, then doing sequential finetuning (SFT) to establish a lower bound, finetuning on an aggregated dataset to establish an upper bound, experimenting with dataset size and replay ratio on experience replay (ER), reinforcement learning-based knowledge distillation (RD), and proximal policy optimization (PPO) with an SFT warmup phase, all integrated with PEFT via LoRA. In all of the continual learning methods, the base model was the model finetuned on solely German; Dutch data was added into the training pipeline afterwards. All methods share a common setup unless otherwise noted. We use the `SmolLM-360M-Instruct` HuggingFaceTB (2024) as a base, tokenized with a maximum length of 512 tokens. All training is done using PEFT with LoRA adapters ($r = 16, \alpha = 16$, dropout $= 0.05$), optimized with AdamW ($\eta = 5E - 5$, batch size $= 8$).

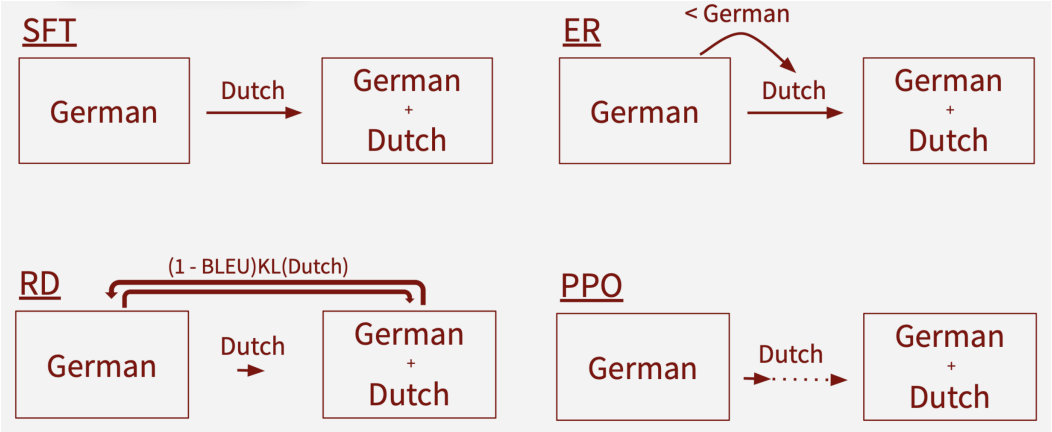


Figure 1: Model and Data relationships for SFT, ER, RD with SFT Warmup, and PPO with SFT between the input model that only supports German and the next iteration of the model that supports both German and Dutch

To better illustrate the differences in how each method approaches continual translation, we provide an overview of their training pipelines in Figure 1. Each method represents a different strategy for managing the relationship between the previous task (German to English) and the new task (Dutch to English) either by sharing parameters across tasks (as in sequential finetuning), replaying past data (ER), constraining updates via model-based alignment (reinforcement distillation), or shaping optimization through reward signals (PPO). This distinction reflects our broader hypothesis: that different forms of memory explicit via data, implicit through gradients, or policy-aligned will yield varying balances between retention and adaptation.

3.1 Datasets

Language	Input	Gold English Translation
German (DE)	"Diese Geschichte ist jetzt besonders signifikant."	“The latter story is now particularly significant.”
Dutch (NL)	“Nu moeten er daden volgen.”	"Actions must follow."

Table 1: Sample entries from Dutch and German parallel corpora within WMT Bojar et al. (2014)

We collected our parallel corpora on German (DE)-English (EN) pairs and Dutch (NL)-EN pairs from WMT in Bojar et al. (2014). Let's denote the DE-EN training set as $\mathcal{D}_{DE}$ and the analogous NL-EN training set as $\mathcal{D}_{NL}$. We formatted each entry $x^{(i)} \in \mathcal{D}_l$, where $l \in \{DE, NL\}$ as the following:

$$x^{(i)} = \texttt{\{“id”: } \; x_{\text{id}}^{(i)}\texttt{, “source”: } \; l\texttt{, “input”: } \; x_i\texttt{, “output”: } \; y_i\texttt{\}},$$

where $x_{\text{id}}^{(i)} = i \in [|\mathcal{D}_{DE}|] = [|\mathcal{D}_{NL}|] = [35000]$ contains our example identification ID, x_i is our input phrase in our source language l, and y_i is our reference translation in EN. Examples of translations for each of the languages is seen in Table 1. Each language in our parallel corpora contained over 35000 examples, and so we partitioned each dataset into a 90% training, 5% validation, and 5% test set. Further, each experiment uses the same prompt when training on the data:

$$\texttt{system} : \text{Translate from } l \text{ to English} \quad \texttt{user} : x_i \quad \texttt{assistant} : y_i.$$

for all $l \in \{\text{Dutch, German}\}$

3.2 Baselines

We have two baselines: a lower bound and an upper bound.

3.2.1 Lower Baseline

We began with a two-stage sequential finetuning (SFT) approach. Starting with our base model of `SmolLM-360M-Instruct` HuggingFaceTB (2024) (Base), we finetuned on $\mathcal{D}_{DE}$ and then on $\mathcal{D}_{NL}$ to form our lower-bound baseline. To construct our input, we used `AutoTokenizer` from Base with a maximum sequence length of 512 tokens.

We enabled PEFT; this means that LoRA adapters with rank $r = 16$, scaling factor $\alpha = 16$, and dropout rate of 0.05 were injected into the transformer attention and feedforward layers. Taking matrices $A \in \mathbb{R}^{d\times r}, B \in \mathbb{R}^{r\times d}$, where $W_0 \in \mathbb{R}^{d\times d}$ are the original frozen feedforward weights, we see that the effective weight we use during finetuning is:

$$W_{new} = W_0 + \Delta W,$$

where $\Delta W = AB$ is our low-rank weight update.

Our training objective here was:

$$\mathcal{L}_{\text{CE}}(\theta) = -\frac{1}{|\mathcal{D}_l|} \sum_{i=1}^{|\mathcal{D}_l|} \sum_{t\in\text{target}} \log p(y_{i,t}|x_{i,<t};\theta),$$

where our training proceeded in two stages indicated by

$$\theta = \theta^* = \arg\min_\theta \mathcal{L}_{\text{DE}}(\theta) \quad \text{and} \quad \theta = \theta^{**} = \arg\min_\theta \mathcal{L}_{\text{NL}}(\theta).$$

In the above weight formulation, note that $\mathcal{L}_{\text{DE}}$ and $\mathcal{L}_{\text{NL}}$ denotes the cross-entropy losses on DE-EN and NL-EN datasets, respectively. We used AdamW Paszke et al. (2019) with a learning rate of $\eta = 5 \times 10^{-5}$ and a batch size of 8. We trained on $\mathcal{D}_{DE}$ for 3 epochs and then on $\mathcal{D}_{NL}$ for 5 epochs.

For the continual methods, we use the German-only finetuned model as the input model.

3.2.2 Upper Bound

To construct an upper bound, we aggregated and shuffled our data to form $\mathcal{D}_{UB} = \mathcal{D}_{DE} \cup \mathcal{D}_{NL}$ finetuned on it using the obvious analogue from doing SFT. We trained for 3 epochs with the same learning rate and LoRA parameters as in the lower bound.

3.3 Experience Replay

Experience Replay (ER) aims to mitigate the effects of catastrophic forgetting by maintaining a memory buffer of examples from the previous training situation. In our case, we took our model that was finetuned on $\mathcal{D}_{DE}$ and trained it to translate NL-EN using a memory buffer of examples from $\mathcal{D}_{DE}$ along with the abbreviated NL-EN training set $\mathcal{D}_{NL}^{\phi} = \mathcal{D}_{NL[:\phi]}$. Specifically, we took the maximum number of replay samples as $N_{max} = 8000$, the replay ratio as ρ, and sampled

$$N_{replay} = \min\{\lfloor \rho |\mathcal{D}_{NL}^{\phi}| \rfloor, N_{max}, |\mathcal{D}_{DE}|\}$$

samples from $\mathcal{D}_{DE}$ to form $\mathcal{D}_{\text{DE-replay}}^{\phi} \subset \mathcal{D}_{DE}^{\phi}$, and aggregated to finally get:

$$\mathcal{D}_{mix}^{\phi} = \mathcal{D}_{NL}^{\phi} \cup \mathcal{D}_{\text{DE-replay}}^{\phi},$$

which we then shuffled in a random order to balance new and replayed samples every epoch. Using the same tokenization and model scheme as when constructing the lower and upper bounds, the model now minimizes the function:

$$\mathcal{L}_{\text{ER}}(\theta) = -\frac{1}{|\mathcal{D}_{\text{mix}}^{\phi}|} \sum_{(x_i, y_i) \in \mathcal{D}_{\text{mix}}^{\phi}} \sum_{t \in \text{target}} \log p(y_t | x_{<t}; \theta)$$

We began training on $\mathcal{D}_{mix}^{\phi=15000}$ with $\rho = 0.8$, but observed poor adaptation performance (as detailed in Section 4). To better balance retention and learning, we then proceeded to repeat ER with $\mathcal{D}_{mix}^{\phi=|\mathcal{D}_{DE}|}$ with $\rho = 0.9$.

3.4 Reinforcement Distillation with Supervised Warmup

We now describe a hybrid approach combining supervised learning and reinforcement learning to fine-tune a student model for the NL-EN task.

This method is motivated by the idea that a frozen teacher model, trained on the prior task, can serve as a stabilizing influence on the student model during training on a new task. By aligning the student's output distribution with that of the teacher via KL divergence and weighting this alignment by a reward signal, we bias learning toward preserving prior knowledge while still allowing for adaptation.

We begin by initializing both the teacher and student models from the German-only finetuned model. LoRA adapters trained on DE-EN data are loaded into both the teacher (θ_{teacher}) and student (θ_{student}) via PEFT. During training, the teacher model is frozen. Using the same tokenization and prompting scheme as in prior methods, we begin with a supervised warmup phase in which the student minimizes the following cross-entropy loss:

$$\mathcal{L}_{\text{CE}}(\theta_{\text{student}}) = -\frac{1}{|\mathcal{D}_{NL}^{\phi}|} \sum_{i=1}^{|\mathcal{D}_{NL}^{\phi}|} \sum_{t} \log p(y_{i,t} | x_{i,<t}; \theta_{\text{student}}). \tag{1}$$

This objective teaches the student to map Dutch inputs to correct English outputs. We use the AdamW optimizer with a learning rate of $\eta = 5 \times 10^{-5}$. The warmup phase consists of 1000 training steps using 1000 Dutch examples.

After the warmup, we perform reinforcement distillation. For each input $x^{(i)}$, the student generates a translation $\hat{y}^{(i)}$. A BLEU-based reward is then computed:

$$r^{(i)} = \text{BLEU}(\hat{y}^{(i)}, y^{(i)}), \tag{2}$$

where $y^{(i)}$ is the reference translation. The reinforcement distillation loss is then:

$$\mathcal{L}_{\text{RD}} = \frac{1}{|\mathcal{D}_{NL}^{\phi}|} \sum_{i=1}^{|\mathcal{D}_{NL}^{\phi}|} (1 - r^{(i)}) \cdot \text{KL}_i, \tag{3}$$

where KL_i is the KL-divergence between the teacher and student output distributions:

$$\mathrm{KL}_i = \sum_t p_{\text{teacher}}(t|\mathbf{x}_i) \log \frac{p_{\text{teacher}}(t|\mathbf{x}_i)}{p_{\text{student}}(t|\mathbf{x}_i)}. \tag{4}$$

We apply a temperature parameter $\tau = 2$ to soften the teacher logits before computing the KL divergence. This encourages smoother gradients and better alignment between teacher and student distributions. The reinforcement phase was trained on $\phi = 30,000$ Dutch examples.

3.5 Simple PPO Fine-tuning with Supervised Warmup

Similar to reinforcement distillation, we explore another hybrid approach that combines supervised learning with PPO Schulman et al. (2017). PPO is a policy gradient method that stabilizes training through a clipped objective, which prevents large updates that can lead to catastrophic forgetting. The initial supervised warmup anchors the model's behavior, ensuring it does not begin reinforcement learning from a random or unstable policy, which is particularly critical for sequence generation.

We begin with a supervised warmup phase using the same cross-entropy loss as in Equation 1. Afterwards, the model switches to PPO, generating translations $\hat{y}^{(i)}$ and computing BLEU-based rewards $r^{(i)}$ as defined in Equation 2. A value network $V(s_i)$ estimates the expected return, and the advantage is computed as:

$$A_i = r^{(i)} - V(s_i).$$

The PPO objective is then defined as:

$$\mathcal{L}_{\text{PPO}} = -\frac{1}{N} \sum_{i=1}^{N} \min\left(r_t A_t, \text{clip}(r_t, 1-\epsilon, 1+\epsilon) A_t\right),$$

where r_t is the ratio of student to reference policy probabilities, $\epsilon = 0.15$ is the clipping parameter, and N is the number of training examples. The value network is optimized using mean-squared error loss:

$$\mathcal{L}_{\text{value}} = \frac{1}{N} \sum_{i=1}^{N} (V(s_i) - r^{(i)})^2.$$

Although PPO typically struggles when initialized on previously unseen tasks, our supervised warmup phase mitigates this issue by grounding the model in the new task before policy optimization begins. We finetune on 30000 Dutch examples, using 1000 supervised warmup steps with 1000 examples.

3.6 Evaluation

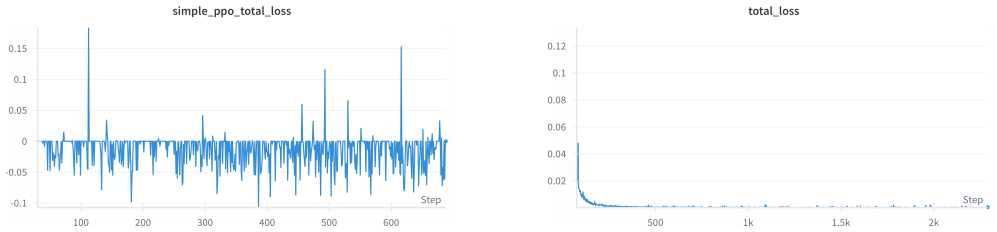
We evaluated on our validation and test splits for both languages l after each task. Using two metrics, BLEU Papineni et al. (2002) and METEOR Banerjee and Lavie (2005), we generated translations $\hat{y}^{(i)}$ for all $i \in [3000]$ for each language and for each split, and then calculated the average scores on them (comparing with $y^{(i)}$).We use BLEU because it is a standard metric for neural machine translation in which the quality of a generated sentence is measured by the precision of overlapping n-grams (up to 4-grams) between the candidate translation and one or more reference translations. It is particularly well-suited for evaluating surface-level fluency and structural similarity to the ground truth.

We complement BLEU with METEOR, which incorporates both unigram precision and recall, and includes stemming, synonymy matching, and a fragmentation penalty to better assess alignment and word order. METEOR has been shown to correlate more strongly with human judgment than BLEU in some settings, especially for morphologically-rich or low-resource languages. Together, these metrics provide a robust evaluation of both the syntactic fidelity and semantic adequacy of our model outputs in this setting.

4 Results

In this section, we will show how the models trained with training curves along with the model's translation performance on German and Dutch.

4.1 Training Results



(a) Total loss training curve for PPO

(b) Total loss training curve for RDL

Figure 2: Selected training loss curves

For brevity, we attach two training curves: one for PPO and one for RDL in Figure 2. We can see different volatilities in these training paradigms. In the RDL paradigm, the loss consistently moves toward 0. On the other hand, the training curve for PPO oscillates much more around 0. These training curves demonstrate the differing convergence behavior between the RL strategies. RD steadily reduces the loss across training steps, indicating strong alignment with the teacher. In contrast, PPO fluctuates significantly, suggesting sensitivity to reward sparsity and the importance of longer warmup or alternative reward shaping strategies.

4.2 Evaluation Results

Table 2: BLEU and METEOR scores for German → English and Dutch → English translation

Method	German BLEU	Dutch BLEU	German METEOR	Dutch METEOR
German Base	**8.44**	N/A	**34.49**	N/A
Lower Baseline	4.68	11.39	29.31	37.91
Upper Baseline	7.40	8.57	31.61	33.04
Experience Replay	7.30	6.43	32.14	29.28
Reinforcement Distillation	7.55	3.43	35.84	21.99
PPO w/ SFT Warmup	6.86	3.84	31.79	23.90

We report in Table 2 the BLEU and METEOR scores of the models' performance on German and Dutch to English translations. The German performance is compared to that of German Base; to get a performance close to it and the Dutch performance should be maximized. From the table, we can see that the lower baseline had catastrophic forgetting, having a significant decrease in German translation performance. For the continual learning methods, all of them had moderate retention of German, with Reinforcement Distillation being slightly better at the retention. Clearly, the ER learned Dutch the best by both metrics, suggesting a possible advantage through data-based methodologies over clipping strategies.

5 Discussion

We proceed here to evaluate each of our methods aimed to mitigate catastrophic forgetting along two distinct dimensions: retention and adaptation. While the former deals with the preservation of the performance on the German to English task after training on the second one, the latter is about the ability to learn the new task, specifically translating from Dutch to English. Refer to 4.2 for quantitative results.

5.1 Baseline Performance

The lower baseline demonstrates catastrophic forgetting very clearly. After finetuning on the NL-EN task, the model's BLEU and METEOR scores drop significantly on German, confirming a degrade in the performance on prior knowledge. The upper baseline, however, retains good performance on the DE-EN task and learns the NL-EN task effectively. However, the unrealistic presumption of having access to all task data in each stage of the training process is in violation with the assumptions of purpose of continual learning strategies.

5.2 Experience Replay (ER)

We first observed that using $\phi = 15000$ and $\rho = 0.8$ resulted in extremely poor learning of the NL-EN task, and so decided to repeat the experiment with a larger set of $\phi = 30000$ examples and a lower replay ratio of $\rho = 0.9$. This means we had 30000 Dutch examples interwoven with 3000 German ones. Seeing the BLEU and METEOR scores, we realize that ER clearly retains German performance better than the lower baseline. However, even with more examples and a lower replay ratio, the model struggles to effectively adapt to the new task. This suggests that a modest-to-high replay ratio biases the optimizer towards retention at the expense of adaptation. The BLEU score for Dutch drops significantly from that of the upper baseline, indicating still underfitting on the second task.

5.3 Reinforcement Distillation (RD)

Speaking only in terms of retention, RD performed the best, as it performed nearly on par with the original German-only finetuned model. This suggests that a frozen German model as a teacher gives an effective bias during Dutch training. However, as seen with ER, there are severe problems when looking at adaptation. The RD model is tightly constrained by the teacher, and so often follows its path almost exclusively. One thing to note, however, is that RD preserves semantic meaning quite decently at times. This could be because token-level metrics understate performance with regards to meaning (focusing on form and structure instead).

5.4 PPO with SFT Warmup

Clearly, this is the most unstable method tested. Doing a warmup on 500 NL examples, the BLEU-based reward function yielded extremely low BLEU and METEOR scores. BLEU is sparse and gives a brittle reward signal; this could be a reason PPO had a difficult time optimizing effectively. A denser reward such as METEOR or some learned strategy, along with a long warmup period, may help stabilize updates.

5.5 Example Output Analysis and Comparison

Although the metrics included summarize the performance of the models in comparison quite faithfully to their true performance, examination of example output sometimes yields additional observations. Take, for example, the following (DE) sentence:

> "Nach dem neuen Gesetz können Medienhäuser mit einer Strafe von bis zu 20 Millionen kenianischen Schillingen belegt werden und einzelne Journalisten mit bis zu einer Million sowie dem weiteren Risiko, ihre berufliche Anerkennung zu verlieren oder von einer offiziellen Presseakkreditierung ausgeschlossen zu werden."

This sentence is about the dangers of a new law potentially being adopted in Kenya (reference translation shown in 3) and demonstrates the some of the myriad challenges facing continual translation for languages sharing even somewhat similar lexica.

Semantically, this sentence is a little subtle, since it uses the phrase "media houses," the latter of which word has extremely close translations in German ("haus") and Dutch ("huis"). This is an example of the dangers of continual translation on closely-related (at least, lexically) languages, hence our attempt on the task. If the model is trained on translating the word "house" to "haus," its motivation to change that to "huis" when translating from Dutch (a one-letter discrepancy) may be

Table 3: German → English Example Translations

Method	Translation Output
Lower Bound	"After the new law, media companies will be liable for losses of up to 20 million Swiss francs and individual journalists will be liable for losses of up to 2 million Swiss francs and the risk of losing their professional identity or being forced to resign from their job." (BLEU: 0.10)
Upper Bound	"After the new law, media companies will be liable for losses of up to 20 million euros and will be liable for losses of up to 10 million, and journalists will be liable for losses of up to 10 million and will be liable for losses of up to 10 million." (BLEU: 0.07)
ER	"After the new law, media companies will be subject to a fine of up to 20 million Swiss francs and, of course, journalists will be subject to a fine of up to 20 million Swiss francs or, if they are already employed by a public company, up to 10 million Swiss francs." (BLEU: 0.07)
RD	"After the new law, media companies will be liable for losses of up to 20 million Swiss francs and will be liable for losses of up to 20 million Swiss francs for each journalist who is liable for losses of up to 1 million Swiss francs." (BLEU: 0.08)
PPO & SFT	"After the new law, media companies will be liable for losses of up to 20 million Swiss francs and individual journalists will be liable for losses of up to one million Swiss francs or the risk of losing their professional license." (BLEU: 0.17)
Reference	"Under the new bill, media houses can be fined up to 20 million Kenyan shillings and individual journalists up to one million with the additional risk of being "de-listed" or barred from receiving official press accreditation."

entirely dependent on the signal of the reward function, for example. Seeing the performance of our models on this sentence, we see that PPO actually performs surprisingly well, rehashing what we hinted about earlier regarding the potential strength of the PPO-based translation: meaning, not structural similarity.

6 Conclusion and Future Work

In this work, we investigated RL-based approaches to mitigate catastrophic forgetting in the setting of continual multilingual translation. Specifically, we explored how an SLM trained to translate German to English could be adapted to translate Dutch to English without losing prior knowledge of translating German. We explore sequential finetuning, experience replay, reinforcement distillation, and PPO. Our results show that it is easier to retain the German prior knowledge than learning the new task of translating Dutch. Experience replay was susceptible to the replay ratio ρ, with a higher Dutch ratio significantly improving the learning of Dutch. Reinforcement distillation offered the strongest retention of prior knowledge, but constrained adaptation. PPO demonstrated promise in preserving semantic meaning, though its instability and sensitivity to reward design limited performance.

We have shown a proof of concept for mitigating catastrophic forgetting in SLMs. Future work could extend this framework to large language models (LLMs), where increased capacity may better support learning new tasks. Additionally, exploring longer task sequences would help assess the scalability of these methods in more realistic multilingual and multitask settings. Lastly, a learned reward model could result in better reward signals to better learn the new task.

7 Team Contributions

Each member contributed equally to the project codebase, report, and poster.

- **Riley Carlson:** Implemented and ran the baseline, reinforcement distillation, and PPO scripts
- **Abhijit Devalapura** Implemented and ran experience replay, inference, and evaluation scripts

Changes from Proposal We used different languages than we first tproposed, using Dutch instead of Danish because of the better quality and amount of data compared to the proposed languages. Although we originally proposed EWC with PPO as a experiment (see 1), the model became corrupted due to the exploding gradient phenomenon despite usually-effective gradient clipping. Lastly, we added SFT warm ups to the PPO and RD experiments in order to better train for Dutch since the model had not seen Dutch before at the point of training and would produce garbled outputs initially, resulting in very sparse rewards.

References

Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017. An actor-critic algorithm for sequence prediction. In *International Conference on Learning Representations*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Association for Computational Linguistics, Ann Arbor, Michigan, 65–72. https://aclanthology.org/W05-0909

Ondřej Bojar, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, and Matt Post. 2014. Findings of the 2014 Workshop on Statistical Machine Translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Baltimore, Maryland, USA. https://aclanthology.org/W14-3302

Cyprien de Masson d'Autume, Sebastian Ruder, Lingpeng Kong, and Dani Yogatama. 2019. Episodic memory in lifelong language modeling. In *Advances in Neural Information Processing Systems*. 13143–13152.

Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

HuggingFaceTB. 2024. SmolLM-360M-Instruct. https://huggingface.co/HuggingFaceTB/SmolLM-360M-Instruct. Accessed: 2025-06-07.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Karol Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. In *Proceedings of the National Academy of Sciences*, Vol. 114. National Academy of Sciences, 3521–3526.

Mohammad Norouzi, Samy Bengio, Navdeep Jaitly, Mike Schuster, Yonghui Wu, and Jan Chorowski. 2017. Reward augmented maximum likelihood for neural structured prediction. In *Advances in Neural Information Processing Systems*. 1723–1731.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, 311–318. https://aclanthology.org/P02-1040

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative

Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 8024–8035. https://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence level training with recurrent neural networks. In *International Conference on Learning Representations*.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, Greg Wayne, and Demis Hassabis. 2019. Experience replay for continual learning. In *Advances in Neural Information Processing Systems*. 348–358.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. *arXiv preprint arXiv:1707.06347* (2017). https://arxiv.org/abs/1707.06347

Jonathan Schwarz, Jelena Luketina, Wojciech Marian Czarnecki, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. 2018. Progress & compress: A scalable framework for continual learning. In *International Conference on Machine Learning*. 4528–4537.