CAN SCALE OVERCOME PLASTICITY LOSS IN LARGE LANGUAGE MODELS?

Anonymous authors
Paper under double-blind review

ABSTRACT

The loss of plasticity – the ability of a network to learn new information after having already learned older information – is a fundamental challenge in creating artificial neural networks capable of continual learning. Although this phenomenon has been known for decades, it has mostly been studied in older, relatively small architectures and rarely in natural-language domains. To determine whether loss of plasticity remains a problem in the modern transformer-based LLM paradigm, we study plasticity loss in GPT-style Transformer models trained on a multilingual continual learning problem. Consistent with prior work, we find evidence of plasticity loss across models ranging from 5M to 314M non-embedding parameters, as measured by deterioration on a held-out Vietnamese probing task. We further find that the onset of plasticity loss follows a predictable scaling law, growing sublinearly with model size. These results suggest that larger models may delay the measurable effects of plasticity loss, but that increasing parameter count alone is likely to be insufficient to completely prevent it. We also find evidence of plasticity loss under stationary multilingual training, challenging the view that the phenomenon is exclusive to continual learning with abrupt task changes. Overall, our results suggest that even large Transformer language models trained on natural-language will eventually lose the ability to efficiently adapt to new data after sufficiently long training, in both continual and stationary settings.

1 INTRODUCTION

Neural networks trained for extended periods on changing data can gradually lose their ability to learn from new observations—a phenomenon known as *loss of plasticity* (Ash & Adams, 2020; Lyle et al., 2022; Nikishin et al., 2022; Dohare et al., 2024). In this work, we operationalize plasticity loss as a degradation in a model's ability to improve on a target distribution under a fixed training budget. This phenomenon represents a fundamental challenge for developing systems capable of continual learning, since such systems must be capable of incorporating new information without losing the ability to adapt. Importantly, loss of plasticity differs from catastrophic forgetting in that plasticity loss is about the network becoming incapable of learning new information, rather than forgetting old information.

Continual learning is a fundamental frontier for language models, whose usefulness depends on their ability to adapt to new facts, domains, languages, codebases, and user needs. For example, continual learning could help mitigate knowledge cutoffs and allow coding agents to adapt to new repositories without relying entirely on in-context learning via long prompts. Because of this promise, understanding and preventing plasticity loss has become an increasingly active area of research (Kumar et al., 2024; Lyle et al., 2024b; Chung et al., 2024; Elsayed et al., 2024; Elsayed & Mahmood, 2024; Lewandowski et al., 2024; Lyle et al., 2024a; Liu & Mou, 2026; Lillo & Cheney, 2026).

While research on plasticity loss has expanded to include a wider range of domains and architectures, there are still few studies of the phenomenon in natural-language settings with large Transformer models, especially at scale. Previous work involving Transformer architectures has either focused on vision domains (Lyle et al., 2023; Lee et al., 2024; Hernandez-Garcia et al., 2025; Lewandowski et al., 2025) or used synthetic text data (Farias & Jozefiak, 2025), often with networks containing fewer than 20 million parameters. In natural-language settings with large Transformer models, existing evaluations have considered relatively few tasks, making it difficult to characterize the onset

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

and progression of plasticity loss (Cho et al., 2026). Thus, it remains unclear whether GPT-style models trained on realistic text streams exhibit plasticity loss, how this effect changes with model scale, and whether it is a practical problem in real-world applications.

In this paper, we investigate plasticity loss in GPT-style decoder-only Transformer models trained on natural-language data. Unlike the original Transformer architecture of Vaswani et al. (2017), which used post-layer normalization, our models use the pre-layer-normalization architecture common in modern language models (Radford et al., 2019; Brown et al., 2020; DeepSeek-AI et al., 2025). To measure loss of plasticity, we introduce a multilingual next-token prediction problem in which languages are presented one at a time for a fixed number of tokens, and the process cycles through a sequence of languages so that training can be extended indefinitely. We then periodically evaluate how efficiently each model adapts to a held-out Vietnamese probing task that is not included in the continual pretraining sequence.

Our experiments show evidence of plasticity loss in models ranging from 5M to 314M non-embedding parameters, with smaller models exhibiting measurable deterioration earlier. We find that the onset of plasticity loss follows a predictable sublinear, power-law scaling with model size, suggesting diminishing returns from increasing parameter count alone as a mitigation strategy. We also find evidence of plasticity loss under stationary multilingual training, indicating that abrupt task changes, which are common in the plasticity loss literature but uncommon in practice, are not necessary for the phenomenon to emerge. Finally, we measure several correlates of plasticity loss, including parameter growth, dormant units, and increased skewness of the attention weights spectrum, which provide clues for designing methods that maintain plasticity over long training horizons. However, we have not yet found a 'smoking gun' to diagnose and counteract plasticity loss in large language models.

2 Background

The finding that artificial neural networks can lose their ability to learn after training dates back more than two decades (Smith et al., 2000; Ellis & Lambon Ralph, 2000). Recent work has shown that this phenomenon is not merely a pathology of early neural networks, but also arises in modern deep learning systems trained for extended periods (Lyle et al., 2022; Nikishin et al., 2022; Dohare et al., 2023; Abbas et al., 2023). These observations have motivated a growing body of work aimed at characterizing when plasticity loss occurs, how broadly it appears across architectures and domains, and how it might be prevented.

Loss of plasticity has been observed in feedforward networks (Dohare et al., 2024; Lyle et al., 2023; Kumar et al., 2024; Elsayed & Mahmood, 2024), convolutional networks (Abbas et al., 2023; Nikishin et al., 2022; Lee et al., 2023), residual networks (Dohare et al., 2024; Lyle et al., 2023; Lee et al., 2024), and Transformer architectures (Lyle et al., 2023; Lee et al., 2024; Hernandez-Garcia et al., 2025; Farias & Jozefiak, 2025; Cho et al., 2026; Liu & Mou, 2026). In addition, the phenomenon has been observed across multiple learning problems, such as supervised learning, reinforcement learning, and self-supervised learning (Springer et al., 2025). Thus, the generality of plasticity loss has slowly come to be recognized by the community.

Investigations into plasticity loss have provided insight into the conditions under which the phenomenon is most likely to occur. Dohare et al. (2024) showed, using synthetic experiments based on the MNIST learning problem, that plasticity loss tends to emerge after extended training on non-stationary data. Moreover, these experiments showed that the onset of the phenomenon occurs earlier in smaller networks. Similarly, Lyle et al. (2023) showed that larger networks are more robust to the phenomenon in synthetic tasks based on the CIFAR-10 and MNIST datasets. Although insightful, these investigations have been limited to synthetic problems based on vision datasets using feedforward or convolutional networks, and thus provide little evidence for the phenomenon in more realistic models and at scale.

Previous work has also investigated plasticity loss in the language domain, but existing studies leave open whether the phenomenon appears in GPT-style Transformer models trained on realistic natural-language streams for many tasks. Farias & Jozefiak (2025) found severe plasticity loss in a GPT-style model; however, the learning problem used synthetic text data, and the model contained fewer than a million parameters. Similarly, Liu & Mou (2026) found loss of plasticity in a single-layer

T5 model (Raffel et al., 2020) on a synthetic language problem. Cho et al. (2026) presented an algorithm for maintaining plasticity and preventing forgetting when learning from text data using models with more than a billion parameters. However, their evaluations of plasticity loss included only eight different tasks, which, even in synthetic problems with small networks, may be insufficient to reliably observe the onset of plasticity loss. Lastly, Springer et al. (2025) presented the most comprehensive study of plasticity loss in language models, but focused on the downstream effects of extended pretraining rather than the evolution of the phenomenon in a continual learning setting.

This paper addresses these gaps by studying plasticity loss in continual learning with realistic natural-language data. We study a range of GPT-style Transformer model sizes, complementing the stationary-learning results of Springer et al. (2025) with experiments in a non-stationary continual learning regime. Lastly, we train on a large number of tokens and tasks to test whether plasticity loss emerges in natural-language Transformer training and whether its onset changes predictably with model size. Moreover, we believe that our naturalistic multilingual training and evaluation framework provides a useful task to elicit loss of plasticity for future work to build upon.

3 Learning Problem

Our study of plasticity loss follows the continual learning setting in which a model is trained on a sequence of learning problems, or tasks (Wang et al., 2024). Let $\tau \in \{1, 2, \dots\}$ denote the sequence of tasks encountered during training. Each task τ has a corresponding space of observations $\mathcal{X}_\tau$, targets $\mathcal{Y}_\tau$, and a probability distribution over observation-target pairs, $(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}_\tau$. The goal of the learning system is to predict the target $\mathbf{y}$ corresponding to an observation $\mathbf{x}$.

We focus on learning systems based on deep neural networks, represented by functions $f_{\boldsymbol{\theta}}$ parameterized by $\boldsymbol{\theta} \in \mathbb{R}^w$, where $w \in \mathbb{N}$. For a task τ, the objective is to find parameters that minimize the expected loss

$$\mathcal{L}_\tau(\boldsymbol{\theta}) \doteq \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}_\tau}\left[\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), \mathbf{y})\right]$$

where ℓ measures the discrepancy between the model's predictions and the target. In practice, we do not have access to the full data distribution. Instead, for each task τ, the model is trained on a finite dataset $D_\tau = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N_\tau}$ by minimizing the empirical loss using mini-batch gradient descent.

A fundamental challenge in continual learning is how to incorporate information from new observations without losing previously acquired knowledge, a tension known as the stability-plasticity dilemma (Mermillod et al., 2013; Abel et al., 2026). Stability refers to the ability to retain performance on previously encountered tasks, while plasticity refers to the ability to learn from new data. Stability has been extensively studied, resulting in a large literature dedicated to preventing catastrophic forgetting in neural networks (McCloskey & Cohen, 1989; French, 1999; Kirkpatrick et al., 2017). In this paper, we focus on plasticity: whether a model trained for an extended period retains the ability to improve on a new target distribution under a fixed training budget.

To measure plasticity loss, we adopt the probing methodology proposed by Lyle et al. (2023). At selected points during training, we train the model on a held-out task and evaluate how efficiently it adapts to the new distribution. A degradation in performance on the held-out task over time indicates that the model has lost some of its ability to learn from new observations.

4 Experimental Setup

4.1 Multilingual Continual Learning Problem

We looked for three desiderata in designing a continual learning problem suitable for studying plasticity loss with large language models. First, the data should be natural-language, overcoming the shortcomings of previous studies using synthetic datasets. Second, the sequence of tasks should be extendable indefinitely to enable the training length required to observe measurable effects of plasticity loss. Lastly, the tasks should be naturalistic and challenging enough for scale to result in meaningful improvements in performance.

Guided by these desiderata, we created a multilingual next-token prediction problem based on sequences of languages from the CulturaX dataset curated by Nguyen et al. (2024). The dataset comprises 167 languages and 6.3 trillion tokens.

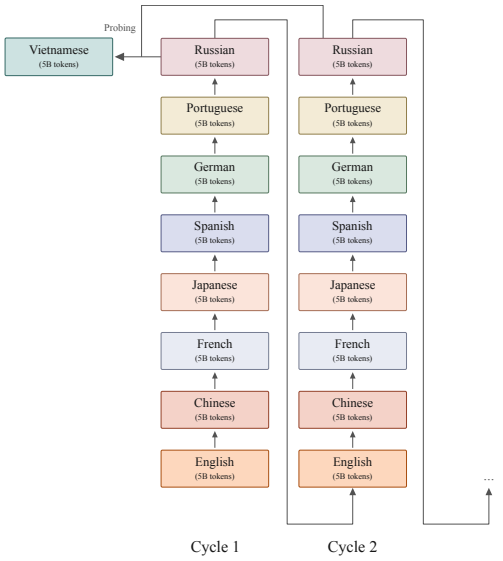


Figure 1: Multilingual continual learning problem. Models are trained on repeated cycles of eight language-modeling task instances. At the end of each cycle, plasticity is assessed using a held-out Vietnamese probing task.

We selected a subset of eight languages from the CulturaX dataset and then created training and evaluation datasets of 100 billion and 1 billion tokens, respectively, for each selected language. We trained networks on the following sequence of languages: English, written Chinese, French, Japanese, Spanish, German, Portuguese, and Russian, in that order. Each time a language appeared in the sequence, the model was trained on a fresh task instance consisting of 5 billion tokens sampled from the corresponding language's training set. We call a complete sequence of eight task instances a *cycle*, and trained networks for several cycles. We refer to training on this non-stationary sequence of task instances as *continual pretraining*.

For the purpose of assessing plasticity loss, we created separate training and validation sets from the Vietnamese language consisting of 20 billion training tokens and 1 billion validation tokens, respectively. At the end of each cycle, we conducted a probing task consisting of 5 billion training tokens randomly subsampled from the Vietnamese 20 billion-token training set. Probing was performed on a copy of the model checkpoint, and the resulting parameter updates were discarded before continual pretraining resumed. Thus, Vietnamese data were used only to measure the model's ability to learn new information and did not become part of the continual pretraining sequence. We illustrate the continual pretraining and probing process in Figure 1.

The language sequence was selected to reduce similarity between consecutive languages. Similarly, the probing language Vietnamese, as the only Austroasiatic language in the training set, was selected to reduce similarity between the probing distribution and the continual pretraining data, limiting transfer and ensuring that probing contained genuinely new information. Since Vietnamese was not included in any cycle, the Vietnamese probing task contained data that had not been observed by the model during continual pretraining, and for which we expect transfer from other languages to be low. Consequently, the performance after training on the new language indicates how well the model can learn from new observations.

During the probing task, we trained for 5B tokens on the new Vietnamese language data, and measured the validation loss using 1,280 randomly sampled sequences of length 2,048 every 95 training steps. To assess how efficiently the model learned during probing, we used the area under the validation-loss curve as a function of probe-training steps. Lower area under the curve values indicate that validation loss decreased earlier during probing, corresponding to faster adaptation.

4.2 Architectures and Training Details

We trained several GPT-style pre-norm decoder-only Transformer models (Radford et al., 2019) of varying sizes. All models use causal self-attention, absolute positional embeddings, pre-layer normalization, an MLP expansion factor of 4, GeLU activations, tied input and output embeddings, and a trained embedding table. When choosing the model configurations, we aimed to minimize architectural variation across scales so that changes in performance could be attributed primarily to model size rather than to differences in width-depth trade-offs. We therefore fixed the aspect ratio of all models, defined as d_{model}/L, to 80, where d_{model} is the Transformer hidden dimension and L is the number of layers. For the same reason, we kept the attention-head dimension fixed at 64, preventing larger models from gaining additional per-head capacity as model size increased.

We trained models at eight different sizes: 5M, 12M, 27M, 39M, 53M, 83M, 106M, and 314M non-embedding parameters. We report model size using non-embedding parameters because the

embedding table size is dominated by the tokenizer vocabulary and does not scale in the same way as the Transformer stack. This choice makes model size a better proxy for the capacity of the Transformer layers whose width and depth vary across experiments. For tokenization, we used the pretrained Qwen3 tokenizer (Yang et al., 2025), which has reasonable coverage across a wide variety of languages and a vocabulary size of 151,680. For total parameter counts and additional architectural details, see Appendix A.

All models were trained with a sequence length of 2,048 using AdamW (Loshchilov & Hutter, 2019), with $\beta_1 = 0.9$, $\beta_2 = 0.95$, a weight decay of 0.1, and a batch size of 0.5M tokens. We used a learning rate schedule with linear warm-up for the first 5% of the total training steps, followed by a constant learning rate. Each task consisted of 9,537 training steps.

We tuned the learning rate using grid searches on the smallest and largest models, training the 5M and 314M models on 5 billion English-language tokens. For each of these two models, we selected the learning rate that minimized the loss under this fixed token budget. The best learning rates for the 5M and 314M models were 3×10^{-3} and 1×10^{-3}, respectively. We then interpolated between these two endpoints to determine the learning rates for the intermediate model sizes (see Appendix A). To validate this procedure, we compared the losses obtained using the interpolated learning rates with the losses obtained during additional grid searches for intermediate model sizes. The interpolated learning rates produced lower loss than the values tested in those grid searches.

The optimizer was reset at the start of each task; this procedure has been shown to prevent training instabilities caused by sudden changes in the loss landscape (Asadi et al., 2023), and also implies that the plasticity degradations we observed were due to inherent plasticity loss in the weights rather than simply stale optimizer states. To further smooth transitions between tasks, we restarted the learning rate warm-up at the beginning of each task. For further details on hyperparameter selection and values, see Appendix A.

5 Experimental Results

5.1 Emergence and Scaling of Plasticity Loss in Continual Pretraining

At the start of our study, we asked a simple question: *is there evidence of plasticity loss in continual learning problems based on natural-language?* To answer this question, we trained each model on our Multilingual Continual Learning Problem for several cycles and probed at the end of each cycle. We measured learning efficiency using the area under the validation-loss curve (AUC) as the model learned the probing task, with lower AUC indicating faster or more effective adaptation.

In this setting, a decrease in probing AUC across cycles indicates improved probing performance, consistent with beneficial transfer from continual pretraining to the probing task—i.e., from learning other languages to learning Vietnamese. In contrast, an increasing trend in probing AUC indicates that the model is becoming less able to learn from new data—a loss of plasticity. Since models vary in their absolute performance on Vietnamese due to inherent differences in capacity, we normalize each model's probing performance relative to its performance after the first cycle. Specifically, let AUC_k denote the AUC measured on the probing task after cycle k. For each model, each point in Figure 2a corresponds to

$$100 \times \frac{\mathrm{AUC}_k - \mathrm{AUC}_1}{\mathrm{AUC}_1} = 100 \times \left(\frac{\mathrm{AUC}_k}{\mathrm{AUC}_1} - 1 \right) \tag{1}$$

The ratio $\mathrm{AUC}_k / \mathrm{AUC}_1$ measures the relative change in probing AUC after cycle k with respect to the first cycle, while subtracting one centers the first-cycle performance at zero. Multiplying by 100 expresses this change as a percentage. Thus, positive values indicate higher AUC than in the first cycle, corresponding to worse probing performance, whereas negative values indicate lower AUC and improved probing performance.

All the models show evidence of plasticity loss on the probing task. Models up to 314M parameters eventually show an increasing trend in AUC as the number of cycles increases. Moreover, the onset of this trend occurs earlier in smaller models. For example, the 5M and 12M models show an increasing trend in AUC immediately after the first cycle, whereas the 83M model shows an

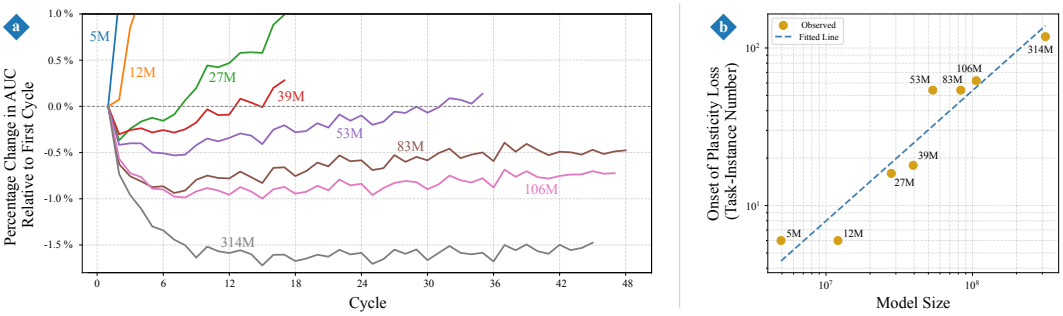


Figure 2: **a.** Percentage change in validation-loss AUC on the probing task relative to the first cycle for models ranging from 5M to 314M non-embedding parameters. All the models eventually exhibit an increasing trend in AUC, indicating a reduced ability to learn, but larger models exhibit this trend after considerably more cycles. **b.** Log-log model predicting the onset of plasticity loss, measured in number of task instances, as a function of model size. The onset of plasticity loss scales as $T = 1.3 \times 10^{-5} \cdot P^{0.8269}$, where P is the number of non-embedding parameters. We observe a qualitatively decent but not perfect fit with several arguable outliers at 12M and 53M.

increasing trend only after the seventh cycle. The results also show that beneficial transfer is more pronounced in larger models: the 53M model achieves a 0.5% reduction in AUC over time, whereas the 314M model achieves more than a 1.5% reduction.

We conclude from these results that pre-norm GPT-style Transformer models can lose plasticity when trained on the Multilingual Continual Learning Problem. However, for a fixed training duration, the effect of plasticity loss diminishes with model size: it is more severe in the 5M model than in the 314M model. Moreover, while continual pretraining initially improves performance on the probing language, excessive continual pretraining eventually has a detrimental effect.

This observation motivates our next research question: *is it possible to predict the onset of plasticity loss based on the size of the model?* In other words, is it possible to derive a *scaling law* for the onset of the loss of plasticity?

To predict the onset of plasticity loss, we first identified the task after which continual pretraining began to have a detrimental effect on probing performance. In other words, we estimated the minimum of each curve in Figure 2a. Since we probed only once per cycle in the experiments above, we lacked the granularity to precisely identify the task after which the trend began to increase, resulting in multiple models sharing the same minimum. Consequently, we performed additional probing tasks within each cycle. For more details on these additional probing tasks, see Appendix B.

From this data, we then fit a log-log model relating the estimated onset to model size, resulting in the power law

$$T = 1.3 \times 10^{-5} \cdot P^{0.8269} \tag{2}$$

where T denotes the task-instance number at which we expect to observe measurable effects of plasticity loss and P denotes the number of non-embedding parameters in the model. In Figure 2b, we show the fitted line along with the observed data. In Appendix B, we report fits for linear, log-linear, and exponential models. All three alternatives produced higher root-mean-squared error than the log-log model under leave-one-out cross-validation, providing some evidence that a power-law fit is most natural.

The log-log model has an important implication for addressing plasticity loss solely through model scaling. Equation 2 predicts that the number of task instances needed to observe plasticity loss scales sublinearly with the number of parameters. Consequently, increasing model size to prevent plasticity loss has diminishing returns and may be an inefficient approach to addressing the problem.

This scaling relationship also provides a useful point of comparison with recent results on larger language models. Springer et al. (2025) provided evidence that the effects of plasticity loss may already be observable in one-billion-parameter models, finding that these models had lower fine-tuning performance after pretraining for longer than 2 trillion tokens. Extrapolating our log-log model to this scale predicts that a model with one billion non-embedding parameters would show signs of plasticity loss after 360 task instances, or 1.8 trillion training tokens, since each task instance

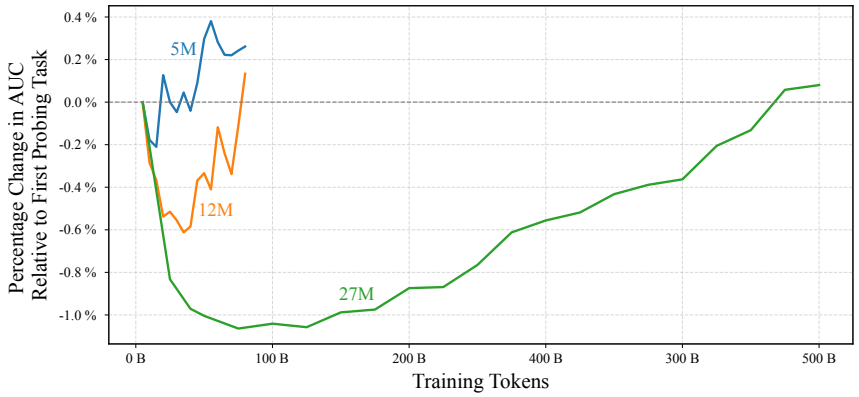


Figure 3: Percentage change in validation-loss AUC on the probing task relative to the first probing task during training on the Multilingual Stationary Learning Problem. As the number of training tokens increases, the AUC of all three models eventually increases, indicating a reduced ability to learn from new information as training progresses.

consists of 5 billion tokens. Furthermore, they reported no consistent evidence of plasticity loss after training seven-billion-parameter models for 2 trillion tokens. This is also consistent with our extrapolation, which predicts that a model of that size would show evidence of plasticity loss after approximately 9 trillion training tokens.

Although these extrapolations are consistent with the overtraining results reported by Springer et al. (2025), the learning problems are substantially different. Springer et al. (2025) evaluated models pretrained on stationary datasets, whereas our results so far are based on a non-stationary learning problem with abrupt task changes. This comparison motivates our next question: *does plasticity loss also emerge under stationary training?*

5.2 Plasticity Loss Under Stationary Training and Network-Level Correlates

To investigate whether plasticity loss occurs under stationary training, we trained the 5M, 12M, and 27M models on a stationary mixture of all eight training languages, which we refer to as the Multilingual Stationary Learning Problem. After every 5 billion training tokens, we performed a probing task by training a copy of the current model checkpoint on Vietnamese for 5 billion tokens and measuring the AUC of the validation loss. As in the continual setting, the parameter updates from probing were discarded. We used the same hyperparameters as in the Multilingual Continual Learning Problem, including the number of warm-up steps for the learning rate.

The results on the Multilingual Stationary Learning Problem show a pattern similar to that observed in the continual learning setting. Figure 3 shows the percentage change in AUC relative to the first probing task. All three models eventually show an increasing trend in AUC, indicating a decrease in the ability to learn from new data. Moreover, the onset of plasticity loss is, once again, delayed with model size: the 5M model shows plasticity loss sooner than the 27M model. However, the rate at which the models lose plasticity appears to differ between settings. This is most evident in the 27M model, which crosses the zero line after 475B tokens of training in the stationary setting and after 320B tokens in the continual setting. Nevertheless, these results suggest that models trained on stationary data are not exempt from plasticity loss, so the phenomenon is unlikely to be explained solely by abrupt task changes that are common in continual learning problems.

Having observed plasticity loss in both continual and stationary learning problems, the next natural question is: *what factors are associated with this phenomenon?* The causes of plasticity loss have been widely debated, and there is currently no clear consensus in the literature (Kumar et al., 2021; Lyle et al., 2023; Lewandowski et al., 2024; Kastner et al., 2026). Fully characterizing the correlates or causes of plasticity loss is beyond the scope of this paper. Nevertheless, we measure several correlates of plasticity loss that may provide insight into the changes occurring inside the network as its ability to learn deteriorates. Understanding these correlates may also suggest approaches for mitigating plasticity loss.

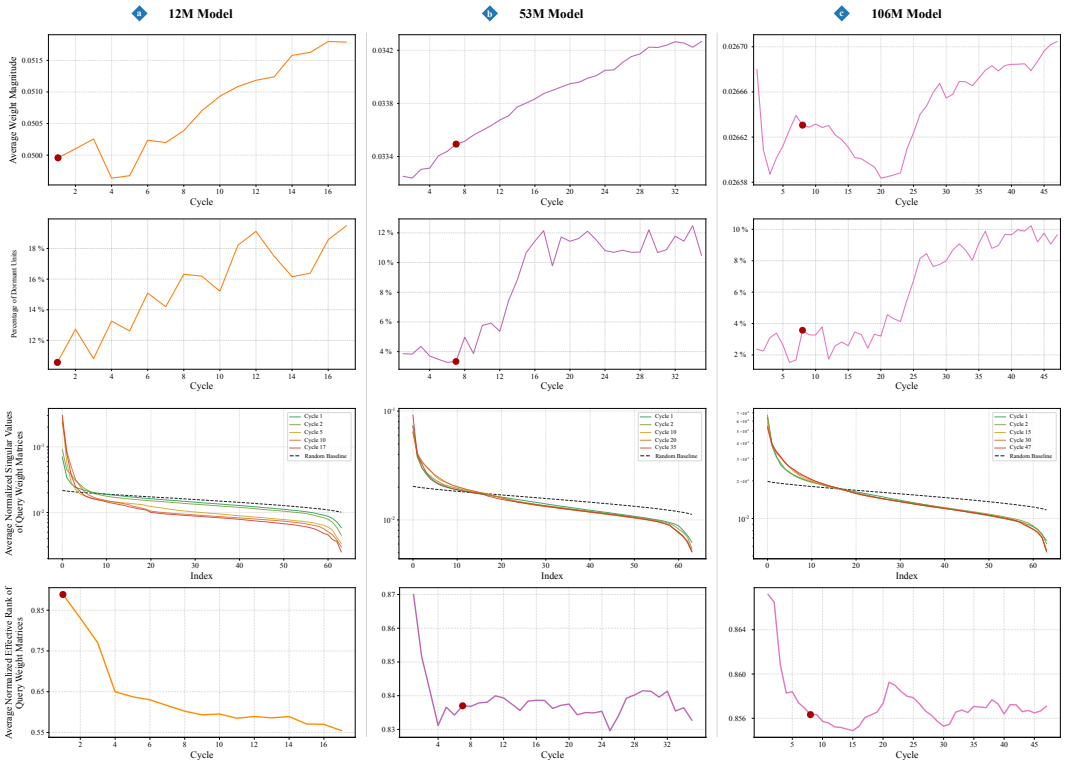


Figure 4: Average parameter magnitude, percentage of dormant units, and average normalized singular values and effective rank of query weight matrices for the (a) 12M model, (b) 53M model, and (c) 106M model trained on the Multilingual Continual Learning Problem. As the number of cycles increases, average parameter magnitude and the percentage of dormant units tend to increase across most models. However, these trends do not consistently track the deterioration in probing performance due to plasticity loss. At the same time, the mass of the spectrum concentrates on a few singular dimensions. This is reflected in a decreasing effective rank as the number of cycles increases. The red marker indicates the cycle after which the model's probing performance began to deteriorate.

In this section, we focus on models with 12M, 53M, and 106M non-embedding parameters trained on the Multilingual Continual Learning Problem. Additional measurements for other model sizes are included in Appendix C.

We recorded metrics previously observed to correlate with plasticity loss. First, we measured the average absolute magnitude of the model parameters, which has repeatedly been observed to correlate with plasticity loss (Kumar et al., 2024; Dohare et al., 2024; Hernandez-Garcia et al., 2025). In all three models, we found an increasing trend in average weight magnitude with the number of cycles (top row of Figure 4). However, this pattern did not consistently track the deterioration in probing performance. For example, the 53M model shows a steady increase in average weight magnitude throughout all the cycles (top row of Figure 4b), but the probing performance improves by half a percent during the first seven cycles.

To further investigate correlates of plasticity loss, we measured the percentage of dormant units in the network's MLP layers. We follow the definition of dormant units from Sokar et al. (2023), who categorize units as ϵ-dormant if the average absolute activation of the unit is below an $\epsilon \in [0, 1]$ fraction of the average of the entire layer. We adapt this definition to the sequence-modelling setting, which we explain in more detail in Appendix C. Using $\epsilon = 0.1$, we found an increasing trend in the percentage of dormant units in all three models (second row of Figure 4). Nevertheless, the percentage of dormant units did not exactly track the deterioration in probing performance. For example, the accumulation of dormant units in the 53M model plateaus (second row of Figure 4b) even as probing performance continues to deteriorate.

We extend our investigation by examining the spectrum of the network's weight matrices. An effective approach for mitigating plasticity loss is to prevent the singular value mass from concentrating on a few singular directions, which would produce a highly skewed spectrum (Lewandowski et al., 2025; Chung et al., 2024; Kastner et al., 2026). The intuition is that highly skewed singular value mass concentrates parameter updates along a low-dimensional subspace. To investigate whether the singular values became highly skewed, we computed the singular value decomposition, normalizing by the sum of the values in each matrix and aggregating across heads and layers (for more details, see Appendix C). We found that key and query weight matrices of attention layers were becoming increasingly skewed as the number of cycles increased (third row of Figure 4 and Figure 8 in Appendix C). The increased concentration of mass of singular values also corresponds to a decrease in effective rank (fourth row of Figure 4). This observation was consistent across model sizes, although the effect was less severe in larger models. Nevertheless, the effective rank plateaus even as the models continue to lose plasticity, suggesting it is not a perfect correlate of plasticity loss.

The additional measurements reported in Appendix C show a similar qualitative picture. None of the three metrics perfectly tracks the onset or severity of plasticity loss across all models. However, taken together, they suggest that continued training changes the networks in ways consistent with reduced effective capacity and slower adaptation. In MLP layers, this appears as an accumulation of low-activity units. In attention layers, it appears as a skewed spectrum in key and query weight matrices, resulting in a reduced effective rank. In parallel, average parameter magnitude tends to increase, which, in the presence of layer normalization, may contribute to slower learning (Lyle et al., 2024a). Together, these effects may contribute to the reduced ability to learn from new data.

This account is necessarily partial. Our measurements are correlational and do not provide a mechanistic explanation for why low-activity units, increased skewness of the spectrum of key and query weight matrices, or increased parameter magnitude emerge over training. Nevertheless, these diagnostics suggest possible directions for designing learning systems that maintain plasticity over longer training horizons.

6 Conclusion

In this paper, we studied loss of plasticity in GPT-style Transformer models trained on natural-language data. Whereas previous studies of plasticity loss have largely focused on smaller models, synthetic problems, or non-language domains, our results show that GPT-style models with more than 100 million non-embedding parameters can lose the ability to efficiently adapt to new data even when trained on real, large-scale natural-language datasets.

Beyond demonstrating plasticity loss in this setting, we are the first to fit a model predicting the onset of plasticity loss as a function of model size. To make this comparison more controlled, we kept the aspect ratio and attention-head dimension fixed across model sizes, reducing architectural variation beyond scale. The resulting model is a classical power-law 'scaling law' between the number of parameters and the number of tokens until the onset of plasticity loss. Our law predicts that the onset of plasticity loss scales sublinearly with the number of non-embedding parameters, implying diminishing returns from increasing model size alone as a strategy for preventing plasticity loss. This prediction speaks to the question posed in the title of this paper: scale alone cannot overcome plasticity loss.

Our result is even more striking given that we also observed evidence of plasticity loss under stationary multilingual training. This means that plasticity loss is not simply caused by our original, somewhat artificially non-stationary task where we cycle through languages. Rather, plasticity loss also seems to affect models which have undergone regular pretraining for a long period on a stationary dataset. Overall, then, we can tentatively conclude that scaling will delay plasticity loss, often for a substantial number of tokens, but cannot fully defeat it. Moreover, we should expect plasticity loss to eventually affect standard language models which have been trained using standard methods. This implies that loss of plasticity is not an artifact of small networks, outdated architectures, or unusual training setups, but rather a fundamental and extremely general property of neural network training.

Lastly, we measured several correlates of plasticity loss inside the network. As models lost plasticity, they tended to accumulate low-activity MLP units, to develop a skewed spectrum in query and key weight matrices, and to accumulate larger parameter magnitudes. Together, they provide useful diagnostics and suggest possible directions for mitigation.

AI USE STATEMENT

In this work, we used generative AI tools for writing code for subsampling the CulturaX dataset used for the multilingual learning problem. We have not used generative AI tools to generate synthetic datasets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, assist with translation, support qualitative and thematic data analysis, or interpret results. Additionally, we used generative AI tools for creating and editing software code, and for editing this manuscript to improve readability. We have reviewed all AI-assisted work. For example, we verified and tested the correctness of AI-generated code. For incorporating edits into this manuscript, we reviewed each edit to guarantee that the proposed changes matched the intention and voice of the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Zaheer Abbas, Rosie Zhao, Joseph Modayil, Adam White, and Marlos C. Machado. Loss of plasticity in continual deep reinforcement learning. In *2nd Conference on Lifelong Learning Agents*, 2023.

David Abel, Michael Bowling, Andre Barreto, Will Dabney, Shi Dong, Steven Stenberg Hansen, Anna Harutyunyan, Khimya Khetarpal, Clare Lyle, Razvan Pascanu, Georgios Piliouras, Doina Precup, Jonathan Richens, Mark Rowland, Tom Schaul, and Satinder Singh. Plasticity as the mirror of empowerment. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=eOZFqyE9Ok.

Kavosh Asadi, Rasool Fakoor, and Shoham Sabach. Resetting the optimizer in deep RL: An empirical study. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=AnFUgNC3Yc.

Jordan Ash and Ryan P Adams. On warm-starting neural network training. In *Advances in Neural Information Processing Systems 33*, pp. 3884–3894, 2020.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 2020.

Dongkyu Cho, Taesup Moon, Rumi Chunara, Kyunghyun Cho, and Sungmin Cha. Forget forgetting: Continual learning in a world of abundant memory. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=fvL8IIEPxG.

Wesley Chung, Lynn Cherif, Doina Precup, and David Meger. Parseval regularization for continual reinforcement learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=RB1F2h5YEx.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao,

Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. Deepseek-v3 technical report, 2025. URL https://arxiv.org/abs/2412.19437.

Shibhansh Dohare, J Fernando Hernandez-Garcia, Parash Rahman, A Rupam Mahmood, and Richard S Sutton. Maintaining plasticity in deep continual learning, 2023. Preprint at https://arxiv.org/abs/2306.13812.

Shibhansh Dohare, J Fernando Hernandez-Garcia, Qingfeng Lan, Parash Rahman, A Rupam Mahmood, and Richard S Sutton. Loss of plasticity in deep continual learning. *Nature*, 632(8026): 768–774, 2024.

Andrew W Ellis and Matthew A Lambon Ralph. Age of acquisition effects in adult lexical processing reflect loss of plasticity in maturing systems: insights from connectionist networks. *Journal of Experimental Psychology: Learning, memory, and cognition*, 26(5):1103, 2000.

Mohamed Elsayed and A. Rupam Mahmood. Addressing loss of plasticity and catastrophic forgetting in continual learning. In *12th International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=sKPzAXoylB.

Mohamed Elsayed, Qingfeng Lan, Clare Lyle, and A. Rupam Mahmood. Weight clipping for deep continual and reinforcement learning. *RLJ*, 5:2198–2217, 2024.

Vivek Farias and Adam Daniel Jozefiak. Self-normalized resets for plasticity in continual learning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=G82uQztzxl.

Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999. ISSN 1364-6613.

J. Fernando Hernandez-Garcia, Shibhansh Dohare, Jun Luo, and Richard S. Sutton. Reinitializing weights vs units for maintaining plasticity in neural networks. In *4th Conference on Lifelong Learning Agents*, 2025.

Tyler Kastner, Nimrod De La Vega, and Amir-massoud Farahmand. Singularclip: Preventing spectral collapse to maintain plasticity in continual and reinforcement learning. In *5th Conference on Lifelong Learning Agents*, 2026.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Aviral Kumar, Rishabh Agarwal, Dibya Ghosh, and Sergey Levine. Implicit under-parameterization inhibits data-efficient deep reinforcement learning. In *International Conference on Learning Representations*, 2021.

Saurabh Kumar, Henrik Marklund, and Benjamin Van Roy. Maintaining plasticity in continual learning via regenerative regularization. In *3rd Conference on Lifelong Learning Agents*, 2024.

Hojoon Lee, Hanseul Cho, Hyunseung Kim, Daehoon Gwak, Joonkee Kim, Jaegul Choo, Se-Young Yun, and Chulhee Yun. Plastic: Improving input and label plasticity for sample efficient reinforcement learning. *Advances in Neural Information Processing Systems*, 36:62270–62295, 2023.

Hojoon Lee, Hyeonseo Cho, Hyunseung Kim, Donghu Kim, Dugki Min, Jaegul Choo, and Clare Lyle. Slow and steady wins the race: Maintaining plasticity with hare and tortoise networks. In *41st International Conference on Machine Learning*, 2024.

Alex Lewandowski, Haruto Tanaka, Dale Schuurmans, and Marlos C. Machado. Curvature explains loss of plasticity, 2024. Preprint available at https://openreview.net/forum?id=SkF7NZGVr5.

Alex Lewandowski, Michał Bortkiewicz, Saurabh Kumar, András György, Dale Schuurmans, Mateusz Ostaszewski, and Marlos C. Machado. Learning continually by spectral regularization. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Hcb2cgPbMg.

Lute Lillo and Nick Cheney. Activation function design sustains plasticity in continual learning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=XZf6wObHX4.

Tianhui Liu and Lili Mou. Do neural networks lose plasticity in a gradually changing world?, 2026. Preprint available at https://arxiv.org/abs/2602.09234.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Clare Lyle, Mark Rowland, and Will Dabney. Understanding and preventing capacity loss in reinforcement learning. In *10th International Conference on Learning Representations*, 2022.

Clare Lyle, Zeyu Zheng, Evgenii Nikishin, Bernardo Avila Pires, Razvan Pascanu, and Will Dabney. Understanding plasticity in neural networks. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 23190–23211. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/lyle23b.html.

Clare Lyle, Zeyu Zheng, Khimya Khetarpal, James Martens, Hado van Hasselt, Razvan Pascanu, and Will Dabney. Normalization and effective learning rates in reinforcement learning, 2024a. Preprint available at https://arxiv.org/abs/2407.01800.

Clare Lyle, Zeyu Zheng, Khimya Khetarpal, Hado van Hasselt, Razvan Pascanu, James Martens, and Will Dabney. Disentangling the causes of plasticity loss in neural networks. In *3rd Conference on Lifelong Learning Agents*, 2024b.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of Learning and Motivation*, 24:109–165, 1989.

Martial Mermillod, Aurélia Bugaiska, and Patrick Bonin. The stability-plasticity dilemma: Investigating the continuum from catastrophic forgetting to age-limited learning effects, 2013.

Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. CulturaX: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 4226–4237, Torino, Italia, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.377.

Evgenii Nikishin, Max Schwarzer, Pierluca D'Oro, Pierre-Luc Bacon, and Aaron Courville. The primacy bias in deep reinforcement learning. In *39th International Conference on Machine Learning*, pp. 16828–16847, 2022.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1), January 2020. ISSN 1532-4435.

Mark Smith, Garrison Cottrell, and Karen Anderson. The early word catches the weights. In T. Leen, T. Dietterich, and V. Tresp (eds.), *Advances in Neural Information Processing Systems*, volume 13. MIT Press, 2000. URL https://proceedings.neurips.cc/paper_files/paper/2000/file/90599c8fdd2f6e7a03ad173e2f535751-Paper.pdf.

Ghada Sokar, Rishabh Agarwal, Pablo Samuel Castro, and Utku Evci. The dormant neuron phenomenon in deep reinforcement learning. In *40th International Conference on Machine Learning*, pp. 32145–32168, 2023.

Jacob Mitchell Springer, Sachin Goyal, Kaiyue Wen, Tanishq Kumar, Xiang Yue, Sadhika Malladi, Graham Neubig, and Aditi Raghunathan. Overtrained language models are harder to fine-tune. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=YW6edSufht.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5362–5383, 2024. doi: 10.1109/TPAMI.2024.3367329.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report, 2025. Preprint available at https://arxiv.org/abs/2505.09388.

A Further Details on Parameter Counts, Architectural Choices, and Hyperparameter Selection

Table 1 reports the architectural configurations and parameter counts for the models used in our experiments. Across model sizes, we kept the aspect ratio d/L, where d is the hidden dimension and L is the number of layers, as close to 80 as possible. We also fixed the attention-head dimension, d/Heads, to 64 for all models. The name of each model refers to the number of non-embedding parameters.

Model	**Layers** L	**Hidden Dim.** d	**Attn. Heads**	d/L	d/Heads	**Eff. Params**	**Total Params**
5M	4	320	5	80.0	64	4,932,480	54,125,440
12M	5	448	7	89.6	64	12,072,256	80,942,400
27M	7	576	9	82.3	64	27,922,752	116,470,080
39M	8	640	10	80.0	64	39,389,440	137,775,360
53M	9	704	11	78.2	64	53,610,304	161,834,816
83M	10	832	13	83.2	64	83,176,704	211,078,400
106M	11	896	14	81.5	64	106,101,632	243,841,920
314M	16	1280	20	80.0	64	314,841,600	511,613,440

Table 1: Model configurations used in our experiments. All models use an attention-head dimension, d/Heads, of 64. The aspect ratio, d/L, was kept as close to 80 as possible. Eff. Params excludes embedding layers, whereas Total Params includes all parameters in the network.

Figure 5 shows the specific Transformer architecture used in our study of plasticity loss. Notably, the architecture uses absolute positional embeddings with tied embeddings, layer normalization before the attention and feedforward blocks (pre-norm), and GeLU activations.

The learning rate, η, was tuned for the 5M and 314M models using a grid search over final validation loss after 5 billion tokens of training on English. We tested values of

$$\eta \in \{1 \times 10^{-4}, 3 \times 10^{-4}, 5 \times 10^{-4}, 8 \times 10^{-4}, \\ 1 \times 10^{-3}, 3 \times 10^{-3}, 5 \times 10^{-3}\}$$

The best learning rates were 3×10^{-3} for the 5M model and 1×10^{-3} for the 314M model. We then interpolated learning rates for the intermediate-size models by modelling the learning rate as a power law in the hidden dimension, similar in spirit to the scaling rule used in μP:

$$\eta_{\text{314M}} = \eta_{\text{5M}} \left(\frac{d_{\text{5M}}}{d_{\text{314M}}} \right)^{\alpha}, \\ \alpha = \frac{\log(\eta_{\text{314M}}/\eta_{\text{5M}})}{\log(d_{\text{5M}}/d_{\text{314M}})} = \frac{\log(1/3)}{\log(1/4)} \approx 0.79248.$$

We verified that this interpolation was sensible by comparing the interpolated learning rates with additional grid-search results for the 12M, 27M, 53M, and 106M models. In every case, the interpolated learning rate yielded a lower final validation loss on English than the other grid-search values tested. The learning rates obtained from this interpolation, together with the grid-search results for each model, are listed in Table 2.

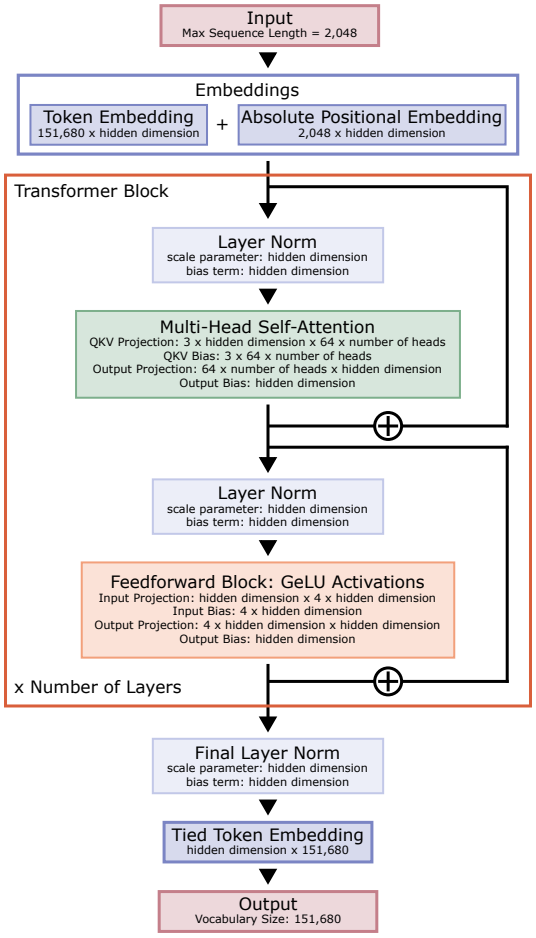


Figure 5: Diagram of the specific Transformer architecture used in our experiments.

Model	d	**Interpolation**		**Grid Search**	
		$\eta \times 10^3$	Loss	$\eta \times 10^3$	Loss
5M	320	3.00	3.9941	3.00	3.9941
12M	448	2.30	3.7973	3.00	3.8088
27M	576	1.88	3.6303	1.00	3.6394
39M	640	1.73	3.5767	–	–
53M	704	1.61	3.5100	1.00	3.5130
83M	832	1.41	3.4436	–	–
106M	896	1.33	3.4006	1.00	3.4037
314M	1280	1.00	3.2575	1.00	3.2575

Table 2: Learning rates per model size, scaled as $\eta = \eta_{5\text{M}} \cdot (d_{5\text{M}}/d)^{0.79248}$. The loss column was computed by averaging the last 10 validation-loss measurements. Each validation-loss measurement was computed using 2.5 million tokens.

Pretraining Language	AUC on Vietnamese Language		
	5M Model	12M Model	27M Model
English	3.0288	2.8189	2.6496
Chinese	3.0634	2.8602	2.6925
French	3.0498	2.8440	2.6718
Japanese	3.0661	2.8746	2.7019
Spanish	3.0502	2.8423	2.6710
German	3.0560	2.8519	2.6842
Portuguese	3.0562	2.8506	2.6712
Russian	3.0793	2.8641	2.6932
Average	3.0562	2.8508	2.6794

Table 3: Performance on Vietnamese after pretraining on different languages for the 5M, 12M, and 27M models. The area under the curve (AUC) of the validation loss measured while training on Vietnamese varies depending on the pretraining language. Pretraining on English consistently results in better transfer than pretraining on any other language. Highlighted rows correspond to the languages selected for extended probing.

B Finding the Minimum of the Plasticity Loss Curve and Creating a Predictive Model

To create the predictive model presented in the main text, we first identified, for each model, the task after which the trend in Figure 2a began to increase. Initially, we only probed after Russian, the last language in the cycle. This approach was ideal for demonstrating plasticity loss because any deterioration in performance could be attributed solely to a loss of learning ability. However, this probing approach provided a coarse level of granularity, preventing us from more precisely determining the start of the increasing trend in performance.

To increase the granularity in our probing, we decided to perform more probing tasks within each cycle. However, probing after each language resulted in noisy data because performance on Vietnamese was heavily influenced by the preceding language. To illustrate this, we show in Table 3 the AUC of the validation loss on the Vietnamese language after pretraining, individually, on each of the eight different languages in the cycle. We found that some languages consistently resulted in better transfer than others. For example, pretraining on English consistently resulted in the lowest AUC across the 5M, 12M, and 27M models. On the other hand, pretraining on Japanese consistently resulted in higher AUC than the average across languages.

To reduce noise from the pretraining language preceding probing, we selected languages with similar levels of transfer. Thus, we selected Chinese, German, and Russian for extended probing, corresponding to the second, sixth, and eighth language in a cycle, respectively. We also considered including Japanese, but the difference in transfer between Japanese and German once again introduced too much noise.

After extended probing, we normalized the AUC line by dividing it by the area under the validation loss curve when training on Vietnamese alone. The blue lines in the plots in Figure 6 correspond to the change in AUC relative to training on Vietnamese alone, i.e., the change in AUC due to continual pretraining. Note that we only extended probing in cycles near the minimum of the line to minimize computational cost.

To further account for the effect of the pretraining language immediately before probing, we computed a moving average with a window size of three. Lastly, we used the minimum point of the moving average line as the onset of plasticity loss. The yellow lines in the plots in Figure 6 correspond to the moving average line with a point indicating the minimum point. Table 4 shows the task after which the trend in AUC increased—the onset of plasticity loss—along with the parameter count of the model.

We used the data in Table 4 to fit four models for predicting the onset of plasticity loss, each encoding a different belief about how the onset, T, scales with model size, P. To encode the belief that onset increases by a constant amount per parameter, we fitted a linear model: $T = a + b \cdot P$. To encode the

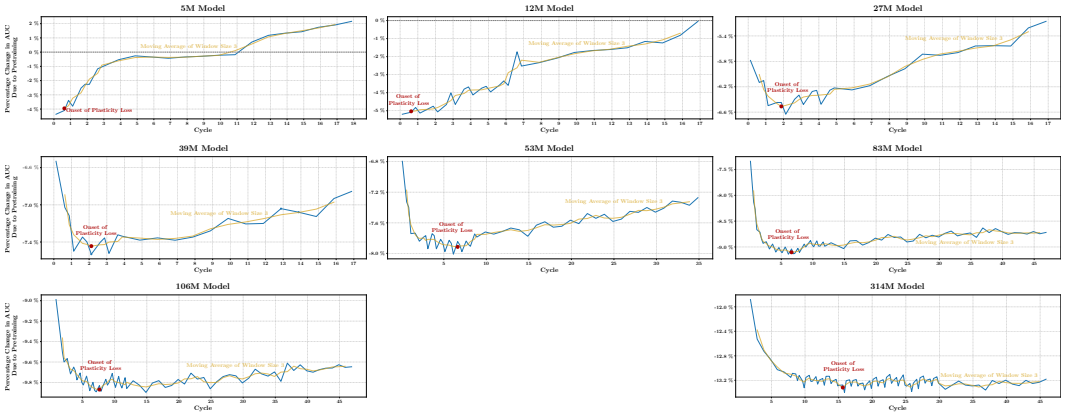


Figure 6: Extended probing in the Multilingual Continual Learning Problem. Each point is the AUC of the validation loss during the probing task divided by the AUC when training on the Vietnamese language alone. The yellow line corresponds to a moving average with a window size of 3. The red dot corresponds to the onset of plasticity loss used for building predictive models as a function of model size. The onset of plasticity loss occurs later with increasing model size.

Model	**Effective Size** P	**Onset of Plasticity Loss** T
5M	4,932,480	6
12M	12,072,256	6
27M	27,922,752	16
39M	39,389,440	18
53M	53,610,304	54
83M	83,176,704	54
106M	106,101,632	62
314M	314,841,600	118

Table 4: Onset of plasticity loss, in number of tasks, for each model size. The onset was obtained by taking the minimum of the change in AUC plots after applying a moving average of window size 3.

belief that the onset increases by a constant amount per order-of-magnitude increase in parameters, we fitted a linear-log model: $T = a + b \cdot \ln P$. We fitted the exponential model: $T = a \cdot e^{b \cdot P}$, which encodes the belief that each additional parameter has a compounding effect on the onset.

Finally, we fitted the log-log model: $\ln T = a + b \cdot \ln P$, which encodes the belief that the onset grows multiplicatively with model size, i.e., multiplying P by a scalar changes T by a constant factor.

We compared the four models using the R^2 value, the root-mean-squared error computed on the data (in-sample RMSE), and the RMSE computed using a leave-one-out cross-validation (out-of-sample RMSE). The coefficients and comparison metrics are given in Table 5. Figure 7 shows the fitted line of each model. Of all the models, the log-log model scored the best.

Model	Coefficients			R^2	RMSE	
	Name	Value	STE		In-sample	LOO
Linear $T = a + bP$	a b	12.9086 3.5937×10^{-7}	6.3401 5.12×10^{-8}	0.8914	11.827	31.113
Linear-log $T = a + b \ln P$	a b	-431.4626 26.9574	88.0119 5.0018	0.8288	14.852	24.615
Exponential $T = ae^{bP}$	a b	24.459 5.0992×10^{-9}	6.830 1.07×10^{-9}	0.7844	16.668	588.914
Log-log $\ln T = a + b \ln P$	a b	-11.2501 0.8269	1.8077 0.1027	0.8969	11.524	17.864

Table 5: Comparison of four functional forms for predicting the onset of plasticity loss, T, as a function of effective parameter count, P. Coefficients are reported with their standard errors (STE). In-sample RMSE is measured on the data points used to fit the model; LOO RMSE is the out-of-sample error from leave-one-out cross-validation.

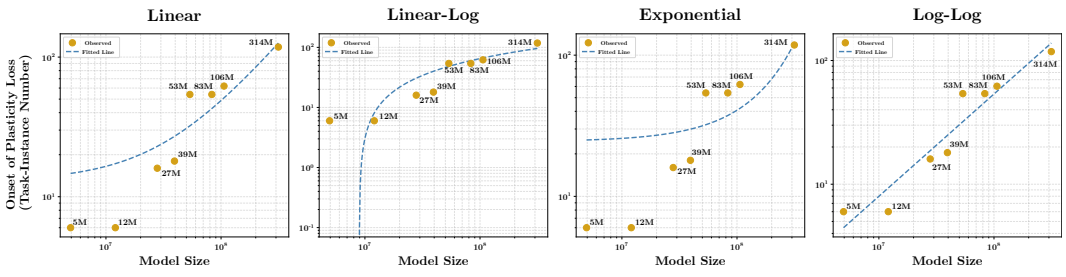


Figure 7: Comparison between different models for predicting the onset of plasticity loss.

C CORRELATES OF PLASTICITY LOSS IN THE MULTILINGUAL LEARNING PROBLEMS

In this appendix, we present more details on how the correlates of plasticity loss were computed, as well as extended results with other model sizes.

C.1 AVERAGE WEIGHT MAGNITUDE

We computed the average weight magnitude at the end of each cycle. For a parameter vector $\boldsymbol{\theta} \in \mathbb{R}^w$ for some $w \in \mathbb{N}$, the average weight magnitude is computed as:

$$\frac{1}{w}\sum_{i=1}^{w}|\boldsymbol{\theta}_i|.$$

We computed this average over all non-embedding parameters in the network, including both weights and biases; we did not compute it separately per layer before averaging.

C.2 DORMANT UNITS

We adapted the definition of dormant units used by Sokar et al. (2023) to the sequence-modeling setting.

Let $h_i^l(\mathbf{x}, j)$ denote the activation of MLP unit i in layer l at token position j for sequence $\mathbf{x}$. Given a dataset D of sequences of length m, let the average absolute activation of this unit be

$$\bar{h}_i^l = \frac{1}{|D| \cdot m}\sum_{\mathbf{x}\in D}\sum_{j=1}^{m}\left|h_i^l(\mathbf{x}, j)\right| \tag{3}$$

We then define the normalized absolute activation as

$$s_i^l = \frac{\bar{h}_i^l}{\frac{1}{N^l}\sum_{k=1}^{N^l}\bar{h}_k^l} \tag{4}$$

where N^l is the number of units in layer l. A unit i in layer l is said to be ϵ-dormant if $s_i^l \leq \epsilon$, for some small positive threshold ϵ. A high fraction of dormant units suggests that a large portion of the layer has low activity relative to other units in the same layer. In networks with smooth activations such as GeLU, dormancy should not be interpreted as exact inactivity, but rather as a diagnostic indicating that some units contribute comparatively little to the layer's representation.

In our experiments, we measured the percentage of dormant units in the network at the end of each cycle. We measured average absolute activations on 512 sequences of length 2,048 sampled from the Vietnamese validation dataset, which the model had not observed during continual pretraining. We set the threshold to $\epsilon = 0.1$. The percentage was first computed separately for each MLP block and then aggregated by averaging across blocks.

C.3 SPECTRUM OF QUERY AND KEY WEIGHT MATRICES

To compute the spectrum, we first gathered the matrices $\mathbf{W}_Q^{l,h}$ and $\mathbf{W}_K^{l,h}$, where l is the index of the layer, h the index of the head, and the subscripts denote the query and key weight matrices, respectively. For simplicity, we drop the subscript with the understanding that the computation could be done with either query or key matrices. For each layer and head, we computed the singular value decomposition of the weight matrix to obtain the diagonal matrix $\mathbf{\Sigma}^{l,h}$. We normalized the entries in $\mathbf{\Sigma}^{l,h}$ by dividing by the sum of the singular values, $\mathrm{tr}(\mathbf{\Sigma}^{l,h})$. The normalized singular values represent the proportion of the total mass each value captures, making them comparable across heads and layers. Lastly, we averaged across heads and layers, obtaining the matrix:

$$\overline{\mathbf{\Sigma}} \doteq \frac{1}{L \cdot H}\sum_{l=1}^{L}\sum_{h=1}^{H}\frac{1}{\mathrm{tr}(\mathbf{\Sigma}^{l,h})}\mathbf{\Sigma}^{l,h}.$$

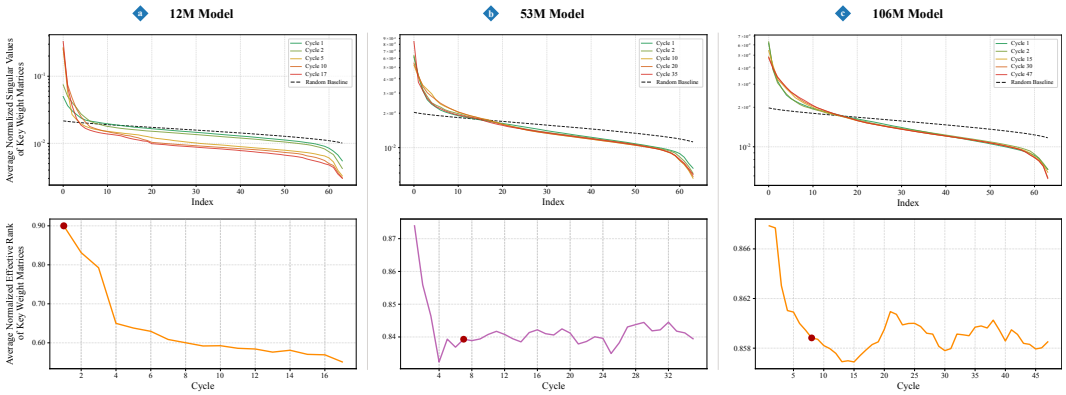


Figure 8: Average normalized singular values and effective rank for the (a) 12M, (b) 53M, and (c) 106M models. The red markers indicate the cycle after which the model's probing performance began to deteriorate.

We then plotted the diagonal entries of $\overline{\mathbf{\Sigma}}$, which are in descending order by construction.

For the effective rank, denote the m-th diagonal element of $\mathbf{\Sigma}^{l,h} / \operatorname{tr}(\mathbf{\Sigma}^{l,h})$ as $p_m^{l,h}$ and let M be the total number of singular values. The effective rank of the matrix is defined as:

$$\operatorname{erank}(\mathbf{W}^{l,h}) = \exp\left(-\sum_{m=1}^{M} p_m^{l,h} \log p_m^{l,h}\right).$$

The average effective rank is the average across layers and heads:

$$\overline{\operatorname{erank}} = \frac{1}{L \cdot H} \sum_{l=1}^{L} \sum_{h=1}^{H} \operatorname{erank}\left(\mathbf{W}^{l,h}\right).$$

We further normalized this quantity by dividing by the maximum possible effective rank, which is M, assuming all the matrices in the sum have the same shape. This normalization scales the effective rank between zero and one, and allows us to compare across layers with differently shaped weight matrices.

We also computed these summaries for the up- and down-projection matrices of MLP blocks and the value weight matrices of attention layers. However, we did not find an effect that correlated with the deterioration in probing performance.

C.4 Results with other model sizes in continual and stationary learning

In Figure 8, we present the normalized singular values and effective rank for the key weight matrices for the 12M, 53M, and 106M models, which were omitted from the main text due to space constraints. The figures paint a similar picture to the spectrum and effective rank of query weight matrices. As the number of cycles increases, the mass of singular values concentrates along fewer dimensions, resulting in a decreasing trend in effective rank.

In Figure 9 and 10, we show the correlates of loss of plasticity in the Multilingual Continual Learning Problem for the models omitted from the main text: 5M, 27M, 39M, 83M, and 314M. The results are consistent with the discussion in the main text. While average weight magnitude tends to increase, the effective rank of the query and key weight matrices decreases, and the models often accumulate dormant units; however, these effects do not exactly match the performance deterioration due to plasticity loss.

Figure 11 shows the correlates of plasticity loss for the models trained in the Multilingual Stationary Learning Problem. The correlates follow a similar pattern as in the continual learning setting. As the number pretraining tokens increases, the average weight magnitude increases, the effective rank of the key and query matrices decreases, and the model accumulates dormant units.

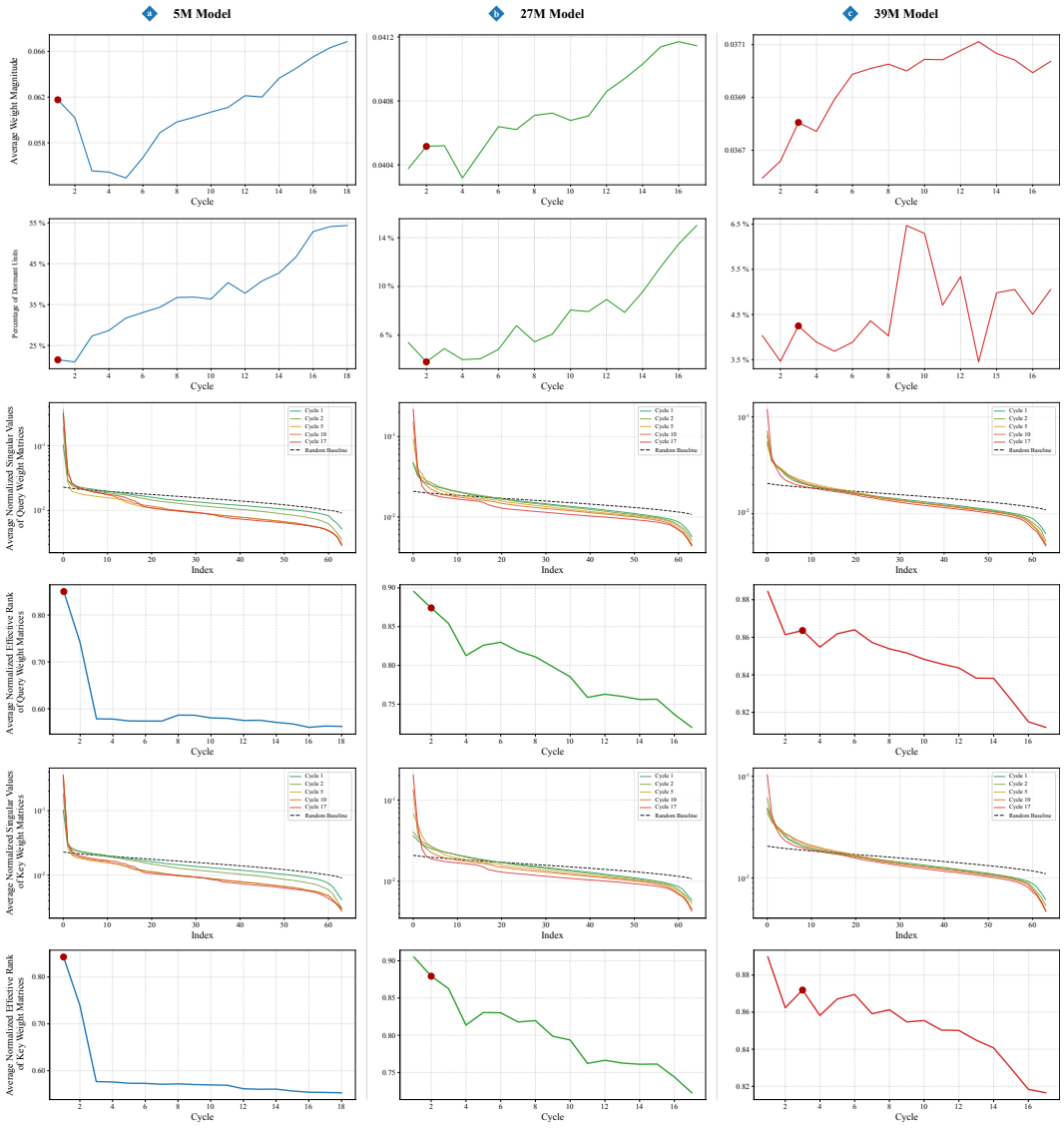


Figure 9: Average parameter magnitude, percentage of dormant units, and average normalized singular values and effective rank of query weight matrices for the 5M, 27M, and 39M models. The red markers in the panes in the first and second row indicate the cycle after which the model's probing performance began to deteriorate.

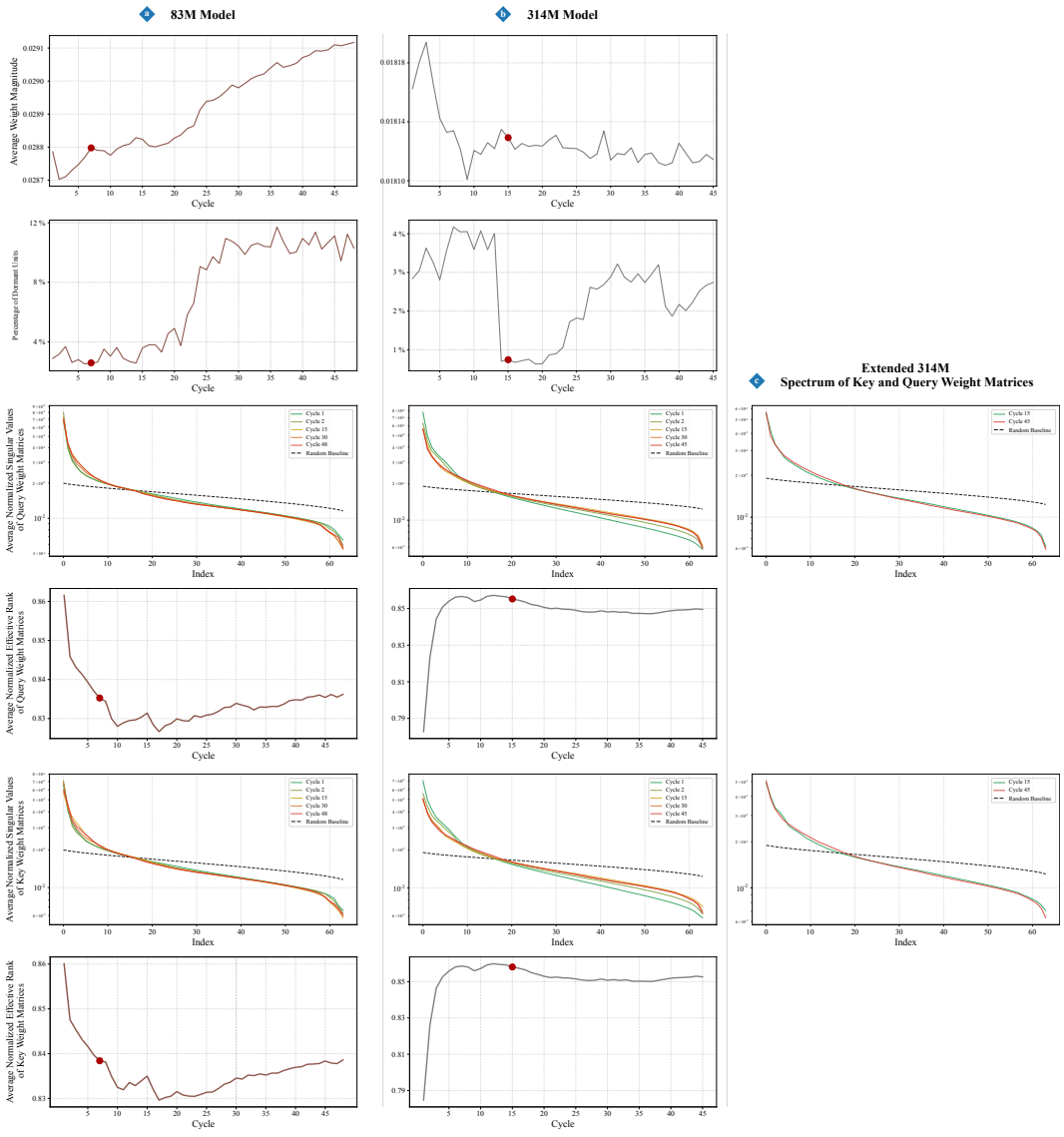


Figure 10: Average parameter magnitude, percentage of dormant units, and average normalized singular values and effective rank of query weight matrices for the 83M and 314M models. The red markers in the panes in the first and second row indicate the cycle after which the model's probing performance began to deteriorate.

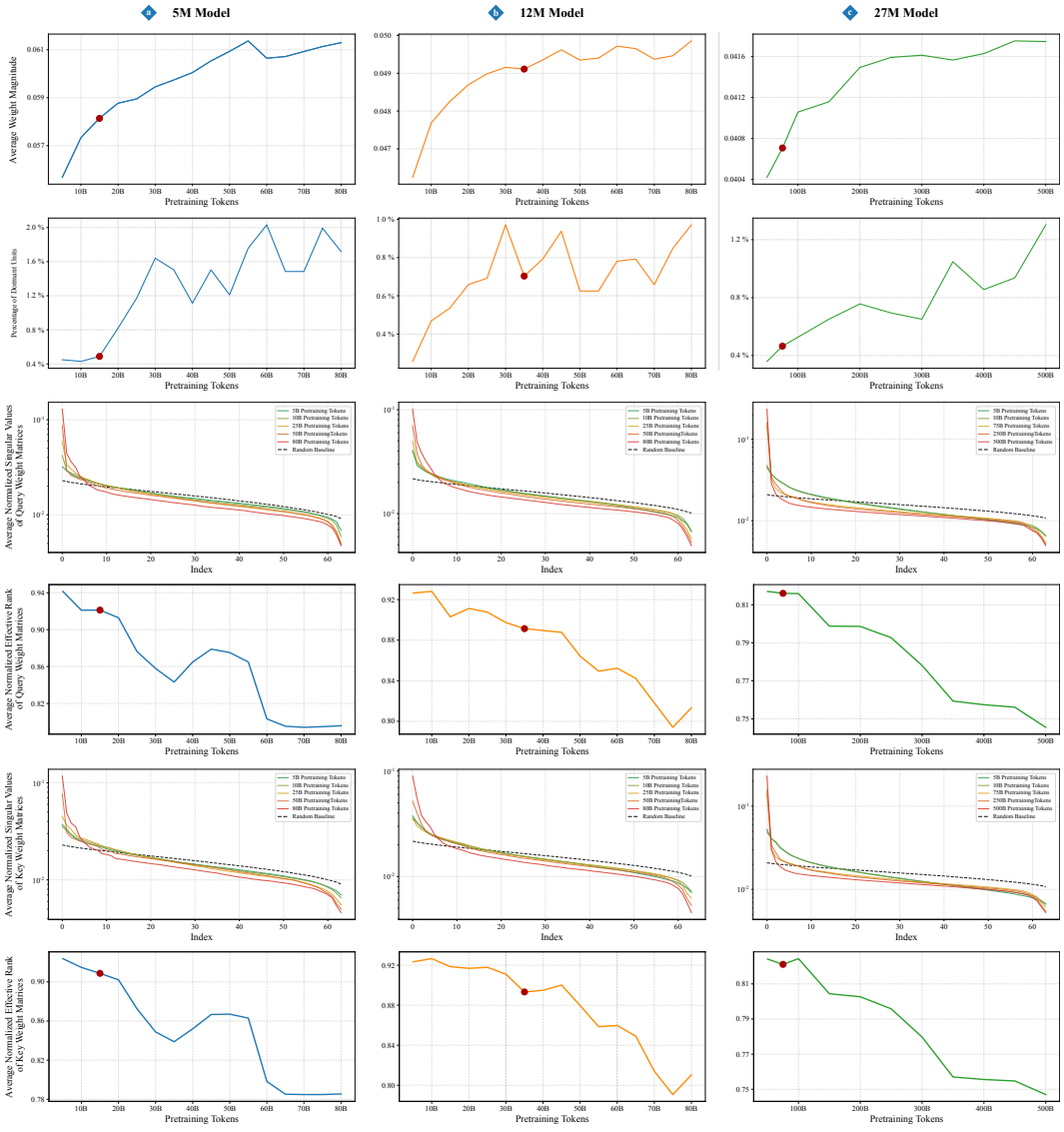


Figure 11: Correlates of loss of plasticity for the 5M, 12M, and 27M models trained on the Multilingual Stationary Learning Problem. The metrics followed a similar trend to the Multilingual Continual Learning Problem. The red markers in the panes in the first and second row indicate the point after which the model's probing performance began to deteriorate.