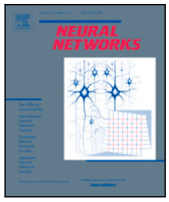
Neural Networks 179 (2024) 106492

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet



Full Length Article

Continual pre-training mitigates forgetting in language and vision

Andrea Cossu [a,*], Antonio Carta [a], Lucia Passaro [a], Vincenzo Lomonaco [a], Tinne Tuytelaars [b], Davide Bacciu [a]

[a] University of Pisa, Largo B. Pontecorvo, 3, Pisa, 56127, Italy
[b] KU Leuven, Kasteelpark Arenberg 10, Leuven, 3001, Belgium

ARTICLE INFO

Dataset link: https://github.com/AndreaCossu/continual-pretraining-nlp-vision

Keywords:
Continual-learning
Lifelong-learning
Pre-training
Self-supervised
Forgetting

ABSTRACT

Pre-trained models are commonly used in Continual Learning to initialize the model before training on the stream of non-stationary data. However, pre-training is rarely applied during Continual Learning. We investigate the characteristics of the Continual Pre-Training scenario, where a model is continually pre-trained on a stream of incoming data and only later fine-tuned to different downstream tasks. We introduce an evaluation protocol for Continual Pre-Training which monitors forgetting against a Forgetting Control dataset not present in the continual stream. We disentangle the impact on forgetting of 3 main factors: the input modality (NLP, Vision), the architecture type (Transformer, ResNet) and the pre-training protocol (supervised, self-supervised). Moreover, we propose a Sample-Efficient Pre-training method (SEP) that speeds up the pre-training phase. We show that the pre-training protocol is the most important factor accounting for forgetting. Surprisingly, we discovered that self-supervised continual pre-training in both NLP and Vision is sufficient to mitigate forgetting without the use of any Continual Learning strategy. Other factors, like model depth, input modality and architecture type are not as crucial.

1. Introduction

Continual Learning (CL) (Lesort et al., 2020) focuses on the design of agents able to learn from a stream of non-stationary data while preserving previously acquired knowledge. The tendency of neural networks to catastrophically forget when confronted with new data has been the subject of many studies (French, 1999; McCloskey & Cohen, 1989), mostly focused on the design of new CL strategies that mitigate such problem (Lange et al., 2021).

The CL scenario currently used in the literature considers a single model tackling a sequence of tasks, one after the other (Parisi et al., 2019). In this setting, the CL model needs to learn its features while, *at the same time*, leveraging the same features to solve the supervised task. However, this scenario is not the only conceivable one.

Natural Language Processing (NLP), for example, often exploits Transfer Learning techniques (Ruder et al., 2019) implemented through the so-called *pre-training fine-tuning* setup. In this setting, the more general linguistic knowledge acquired with pre-training is leveraged as a starting point to target specific downstream tasks. Specifically: (1) during pre-training, language models focus on unsupervised learning tasks (e.g. predicting masked words based on the surrounding context), and (2) during fine-tuning, the pre-trained model is further trained on supervised learning tasks (e.g. sequence labeling).

Pre-trained models are widespread also in CL (Mehta et al., 2021; Ramasesh et al., 2021; Wu et al., 2021), where they are mostly used to conveniently initialize the model weights before learning from the non-stationary stream of data. However, the generality and robustness of the pre-trained representations may be greatly impaired during the continual training on the sequence of tasks, since the model will tend to overfit to the tasks objective.

The recently proposed Continual Pre-Training scenario (Hu et al., 2021; Jang et al., 2021; Jin et al., 2022) separates the goal of building robust features *during* the continual training from that of solving the task currently faced by the model.

The importance of Continual Pre-Training can be better understood with an example: let us consider the case in which a model is pre-trained on a snapshot of Wikipedia containing articles up to 2018. Part of the knowledge contained inside the model will soon become out-dated: on one hand, the information contained in the original articles is likely to be replaced with up-to-date versions (e.g., changes in public figures such as a new President). On the other hand, outdated models do not incorporate the semantics of concepts related to more recent events. For example, the semantics of a term like COVID-19, which becomes important in a short amount of time, cannot be incorporated in the model without additional pre-training. As a consequence, an

* Corresponding author.
E-mail address: andrea.cossu@unipi.it (A. Cossu).

https://doi.org/10.1016/j.neunet.2024.106492
Received 15 June 2023; Received in revised form 3 June 2024; Accepted 24 June 2024
Available online 1 July 2024

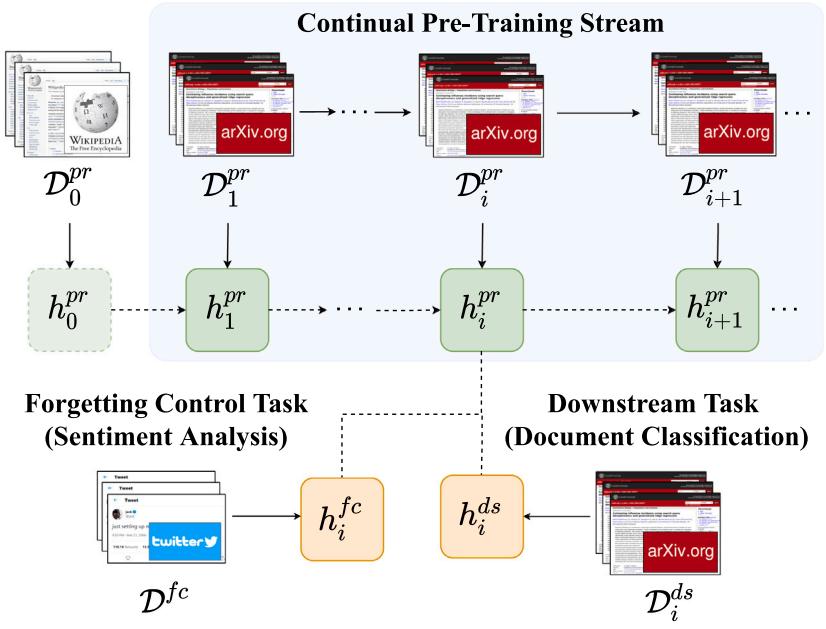


Fig. 1. The Continual Pre-Training scenario. During each stage (experience) i of Continual Pre-Training (top), the model h_i^{pr} is pre-trained (center) on the dataset $\mathcal{D}_i^{pr}$ (e.g., *scientific abstracts*). Subsequently (bottom), the model is fine-tuned against one (or more) downstream task $\mathcal{D}_i^{ds}$ (e.g. *scientific abstracts* classification). Forgetting is measured by fine-tuning on the Forgetting Control dataset $\mathcal{D}^{fc}$ (e.g. *sentiment analysis*). At each stage, only the current pre-trained and downstream datasets/models are available.

outdated language model may perform worse on tasks like language generation and Question Answering (Q/A), since it will not be able to generate sentences related to recent events (Jang et al., 2022).

In this paper, we adopt and formalize the Continual Pre-Training scenario (Fig. 1), where the model is continually updated via an appropriate pre-training objective on a non-stationary stream of (possibly unlabeled) data. After each stage of pre-training, we build a new model from the pre-trained one (e.g., by substituting its final classifier head) and we train it on a number of downstream tasks.

To study forgetting, we monitor whether Continual Pre-Training improves/worsens the performance on downstream tasks which are similar/different with respect to the ones encountered during Continual Pre-Training. For the sake of the evaluation, we specifically introduce a set of Forgetting Control datasets as downstream tasks. Each dataset contains samples different from the ones present in the non-stationary stream and more similar to the dataset used for the original pre-training phase prior to continual training. Against each Forgetting Control dataset, we compare the performance of the pre-trained model at the beginning of the sequence of tasks with the performance of the model after each stage of Continual Pre-Training.

Contributions. Our main objective is to investigate the behavior of different architectures, pre-training protocols and input modalities in the Continual Pre-Training scenario and to understand the impact these factors have on catastrophic forgetting (Table 1).

Unlike recent studies (Fini et al., 2022; Hu et al., 2021; Jang et al., 2021; Jin et al., 2022; Ke et al., 2023-02-01), *we do not employ any custom CL strategy* neither during pre-training nor during fine-tuning. We consider *both language and vision benchmarks*.

Our main contributions and findings are:

1. the formal definition of the Continual Pre-Training scenario (Jin et al., 2022) with the adoption of an evaluation methodology to assess the impact of catastrophic forgetting on separate Forgetting Control datasets (Section 3);
2. the study of forgetting in the Continual Pre-Training scenario by disentangling the effect of 3 main components: the input modality, the model architecture and the pre-training protocol. To do this, we introduce two evaluation environments: one for Natural Language Processing (NLP) and the other for Computer Vision (CV) tasks (Sections 3.1 and 3.2, respectively);
3. the identification of the pre-training protocol as the main component impacting on forgetting. We show that self-supervised pre-training protocols are able to successfully mitigate forgetting, even with a single epoch of fine-tuning. Interestingly, the role of the architecture type and depth does not have an equivalent impact (Section 4.0.0.1);
4. an interpretation of our empirical results in terms of the feature space of our continually pre-trained models by leveraging both linear probing and Centered Kernel Alignment (CKA) (Kornblith et al., 2019) (Section 4.0.0.2). We observe that keeping the hidden features fixed during linear probing exacerbates forgetting for supervised pre-training. Moreover, CKA confirms that supervised pre-training causes a larger drift in the feature space compared to self-supervised pre-training.
5. a Sample-Efficient Pre-training method (SEP) that allows to reduce the pre-training time up to 1 order of magnitude while still preserving an excellent downstream performance (Section 4.0.0.3).

Table 1
Combinations for the main components of the Continual Pre-Training scenario explored in this paper with the results we obtained in terms of forgetting. MLM=Masked Language modeling, MIM=Masked Image Modeling, CLF=Image Classification.

Pre-training	Architecture	Data	Forgetting
Self-Supervised (MLM)	Transformer	Words	×
Self-Supervised (MIM)	Transformer	Images	×
Supervised (CLF)	Transformer	Images	✓
Supervised (CLF)	CNN	Images	✓

2. Related works

The ability of pre-trained models to solve a diverse set of tasks through fine-tuning has led to consider them as almost static models. Nonetheless, it has been shown that taking a pre-trained model and performing an additional step of pre-training on domain-specific data (domain adaptation) is beneficial for the downstream performance in that domain Gururangan et al. (2020), Han et al. (2021), Lee et al. (2020). With a slight abuse of terminology, this process is sometimes

found in the literature with the term continual pre-training. However, since domain adaptation does not perform multiple steps of pre-training, it cannot be truly considered a CL scenario.

The Continual Pre-Training scenario we refer to, formalized in Section 3, has been recently introduced by Fini et al. (2022), Han et al. (2021), Jang et al. (2022), Lazaridou et al. (2021). In Jin et al. (2022), the authors studied Continual Pre-Training on NLP by leveraging a sequence of domain-specific datasets (e.g., multi-domain research papers). Jang et al. (2021), Loureiro et al. (2022) also leveraged NLP data, but they focused on the temporal generalization ability induced by Continual Pre-Training. In particular, Loureiro et al. (2022) showed that models continually pre-trained on a chronologically ordered stream of tweets are able to better predict unseen, future tweets. In line with the results from domain adaptation, Jang et al. (2021), Jin et al. (2022) showed that performing continual pre-training on a sequence of domains is beneficial for downstream performance on tasks coming from the same domains. The studies of Continual Pre-Training in NLP are however limited in terms of forgetting analysis. In fact, the pre-training objective is always an unsupervised one (masked/causal language modeling). This prevents to study the impact this important component has on forgetting.

Few recent works tackled the problem of Continual Pre-Training for Computer Vision tasks (Fini et al., 2022; Hu et al., 2021). In particular, Hu et al. (2021) compared a range of self-supervised methods (e.g., contrastive self-supervised methods like MoCo-v2 by Chen et al. (2020)) with supervised pre-training. They showed that self-supervised approaches are indeed able to mitigate forgetting on previous tasks present in the stream. However, it remains unclear whether self-supervision is the only factor accounting for the boost in performance. The CaSSLe approach (Fini et al., 2022) developed an ad-hoc strategy by combining self-supervision with distillation (Hinton et al., 2015) to mitigate forgetting. Differently from our work, none of the aforementioned works employed separate Forgetting Control datasets to monitor forgetting. They instead computed forgetting directly on the downstream tasks belonging to the CL stream. This makes the analysis dependent on the specific domains learned during Continual Pre-Training.

Instead, we study the problem of forgetting in the Continual Pre-Training scenario by evaluating, for *both Vision and Language* data, the performance of *multiple models* with *multiple pre-training protocols* against *Forgetting Control datasets* not present in the CL stream. We are the first to leverage the *Masked Image Modeling pre-training* (Bao et al., 2021), thus mirroring the Masked Language Modeling protocol used in NLP. Unlike prior works, we did not use any CL strategy, showing that the design of ad-hoc strategies is not always necessary.

3. Continual pre-training scenario

The CL scenario (Lomonaco et al., 2021) trains a model h_0 on a (possibly infinite) stream of experiences $\mathcal{S} = (e_1, e_2, e_3, \ldots)$, where each experience e_i contains a dataset $\mathcal{D}_i$ representing the current task. For example, in the popular case of supervised CL, each $\mathcal{D}_i$ will be composed by a set of N_i input-target pairs $\{x_j, y_j\}_{j=1,\ldots,N_i}$. The model h is trained on $\mathcal{S}$, one experience after the other, and needs to address the non-stationarity and drifts occurring between experiences without having access to the previously encountered data.

Before starting training, the model h_0 is sometimes initialized with the weights of a pre-trained model. The pre-training dataset $\mathcal{D}^{pr}$ is not available during CL.

We provide a formal characterization of the Continual Pre-Training scenario and we highlight the differences with respect to the CL scenario. We also introduce the concept of Forgetting Control (FC) datasets for the purpose of evaluation.

The Continual Pre-Training scenario leverages a model h_0^{pr} originally pre-trained on dataset $\mathcal{D}_0^{pr}$, not available anymore. The model is presented with a (possibly infinite) stream of experiences, where each experience e_i brings a dataset $\mathcal{D}_i^{pr}$ for pre-training and a downstream dataset $\mathcal{D}_i^{ds}$ for fine-tuning. For each experience e_i, the last pre-trained model h_{i-1}^{pr} is further pre-trained on $\mathcal{D}_i^{pr}$. After the pre-training step, the model h_i^{pr} is fine-tuned on $\mathcal{D}_i^{ds}$, resulting in h_i^{ds}. We adopt naive fine-tuning, without any CL strategies.

In order to measure catastrophic forgetting, we leverage a FC dataset $\mathcal{D}^{fc}$ in place of the $\mathcal{D}_0^{pr}$ originally used during the first pre-training phase. While each $\mathcal{D}_i^{ds}$ contains samples similar to the ones encountered during pre-training, the FC dataset contains knowledge more similar to the one in $\mathcal{D}_0^{pr}$ than the one in $\bigcup_{i=1,2,3,\ldots} \mathcal{D}_i^{pr}$. This allows to assess the impact of forgetting after each experience e_i by comparing the performance of h_0^{pr} fine-tuned on $\mathcal{D}^{fc}$ with the performance of h_i^{pr} fine-tuned on the same dataset.

We use h_i^{ds} to verify that the Continual Pre-Training step actually contributes to learning meaningful features for the downstream task. This avoids the uninteresting case where pre-training leaves features (mostly) unchanged, resulting in no catastrophic forgetting of previous knowledge, but also in a lower performance on the downstream task.

Algorithm 1 provides a high-level description of the Continual Pre-Training scenario, showing the steps of continual pre-training, downstream fine-tuning and catastrophic forgetting evaluation against the FC dataset.

Algorithm 1 Continual Pre-training scenario

Require: Pre-trained model h_0^{pr}, stream of experiences $\mathcal{S} = (e_1, e_2, e_3, \ldots)$, FC dataset $\mathcal{D}^{fc}$.

$h_0^{fc} \leftarrow$ fine-tune$(h_0^{pr}, \mathcal{D}^{fc})$ {Evaluate model on FC dataset before continual pre-training}
for $e_i \in \mathcal{S}$ **do**
  $\mathcal{D}_i^{pr}, \mathcal{D}_i^{ds} \leftarrow$ split$(\mathcal{D}_i)$
  $h_i^{pr} \leftarrow$ pre-train$(h_{i-1}^{pr}, \mathcal{D}_i^{pr})$ {Choose appropriate pre-train objective}
  $h_i^{ds} \leftarrow$ fine-tune$(h_i^{pr}, \mathcal{D}_i^{ds})$
  $h_i^{fc} \leftarrow$ fine-tune$(h_i^{pr}, \mathcal{D}^{fc})$ {Evaluate model on FC dataset}
  Compare performance of h_i^{fc} with h_0^{fc} to assess forgetting.
end for

return y

The Continual Pre-Training scenario has different characteristics with respect to the CL scenario. Firstly, the Continual Pre-Training scenario updates continually the pre-trained model and then adapts it to specific tasks. The CL scenario does not consider this important distinction, using the same model both for representation learning and to solve incoming tasks. Secondly, model evaluation in Continual Pre-Training requires an additional training phase on the target task, while CL usually requires the model to be readily able to tackle all tasks seen so far without any additional training. Therefore, the model has to focus on the new task without the opportunity to build robust, general features via pre-training protocols.

As our experiments will show, the additional cost of a training phase in Continual Pre-Training can be largely mitigated by a quick adaptation phase (in our case, one epoch of training is enough). This enables *fast remembering of previous knowledge*, which is considered one of the objectives of CL (Hadsell et al., 2020).

Ultimately, *the Continual Pre-Training scenario aims at building models which are general learners, able to quickly adapt to unseen data while still preserving the original knowledge*.

We studied Continual Pre-Training by introducing two evaluation environments: one for NLP and one for CV. They are designed to investigate the impact on forgetting of specific components of the scenario (Table 1), namely the input modality, the pre-training protocol and the model architecture.

3.1. Natural language processing environment

Our NLP environment employs a self-supervised pre-training protocol (masked language modeling) and different Transformer architectures (Vaswani et al., 2017). We use RoBERTa (Liu et al., 2019) pre-trained on WikiPedia and BERT (Devlin et al., 2019) pre-trained on Wikipedia and Toronto BookCorpus (Zhu et al., 2015). In addition, we introduce and study a variant of RoBERTa in which the vocabulary is dynamically expanded with the addition of new tokens. We select the most frequent tokens of the Continual Pre-Training dataset which were not present in the pre-trained tokenizer. Vocabulary expansion is beneficial for downstream performance, as showed by recent works on dynamic token expansion in both CV (Douillard et al., 2022) and NLP (Han et al., 2021; Zhang et al., 2020). Our aim is to understand whether the addition of new tokens may result in a larger forgetting of existing knowledge.

We apply Continual Pre-Training on a dataset of `scientific abstracts` from arXiv (Geiger, 2019). The motivation behind the choice of this dataset is that `scientific abstracts` represent a very specific domain for NLP both in terms of syntactic structures and domain-specific terminology.

The downstream task is a document classification problem aiming to associate `scientific abstracts` to their corresponding arXiv classes. The CL stream includes 5 experiences, with 2 scientific domains (classes) in each experience (as in common CL benchmarks like Split-MNIST/CIFAR-10). We used $10k$ and $1k$ examples per class during downstream fine-tuning for train and test, respectively. The same, with held-out examples, applies during continual pre-training. We leverage 3 FC datasets: `sentiment analysis` from tweets (Saravia et al., 2018), Question Answering Natural Language Inference (`QNLI`) (Wang et al., 2018) and the `SentEval` benchmark (Conneau & Kiela, 2018), which contains 20 different datasets.

The idea behind these choices is that the dataset of `scientific abstracts` contains domain-specific knowledge which is not related neither to sentiments nor about generic facts for language inference. Therefore, we would expect pre-training on scientific abstracts to disrupt the knowledge contained in the original language model.

3.2. Computer vision environment

We found CV to be a useful test-bed to disentangle the importance of the three components in the Continual Pre-Training scenario. In particular, we design the CV environment to understand *to what extent forgetting depends on the input modality (natural language against vision), on the architecture (Transformer against CNN) and on the pre-training protocol (self-supervised against supervised)*.

To limit the large number of experiments needed to explore these three factors, in the CV environment we do not measure the performance on the downstream task after each step of Continual Pre-Training. Instead, we focus on the study of forgetting on the FC dataset. In fact, the impact of pre-training on downstream tasks similar to the ones in the pre-training stream is assessed both in the discussion of related works (Section 2 above) and in the experiments with `scientific abstracts` classification in NLP environment (results presented below in Section 4 and expanded in the Supplementary Material).

The CV environment uses `iNaturalist` (Horn et al., 2018) for Continual Pre-Training and `CORe50` (Lomonaco & Maltoni, 2017) as FC dataset for catastrophic forgetting. We use ResNet101 (He et al., 2016), Vision Transformer (ViT) (Dosovitskiy et al., 2020) and BEiT (Bao et al., 2021) originally pre-trained on ImageNet. The choice of ResNet and ViT is fundamental to disentangle the role of the architecture (NLP uses only Transformers) and the pre-training protocol (NLP uses only self-supervised pre-training). In fact, ResNet and ViT are pre-trained via *supervised image classification*.

Table 2
Accuracy on the downstream dataset of `scientific abstracts` classification after Continual Pre-Training and for the Base models without pre-training. The split used for downstream classification and pre-training contains different documents. "Model i" means that Model has been pre-trained on i experiences before being fine-tuned on the downstream dataset. "NT" refers to the model where domain-specific New Tokens have been added to the vocabulary and their embeddings have been fine-tuned during training.

Model	Accuracy	1-epoch Accuracy
RoBERTa Base	80.35	78.10
BERT Base	80.97	78.39
RoBERTa Pr. 1	82.59	79.88
BERT Pr. 1	82.64	80.91
RoBERTa Pr. NT 1	82.37	80.58
RoBERTa Pr. 5	83.24	81.19
BERT Pr. 5	83.08	81.84
RoBERTa Pr. NT 5	83.06	81.22

The choice of BEiT, instead, allows to understand the role of the input modality. BEiT uses the recent self-supervised *masked image modeling* pre-training (Bao et al., 2021), which closely resembles the masked language modeling one used in NLP. The proposed setup allows to run experiments by changing one factor at a time among the three we studied and to keep fixed the other two. In this way, we are able to properly compare results between the NLP and CV environments.

3.3. Experimental setup

For NLP, we use the Huggingface's pre-trained BERT and RoBERTa with 12 layers. The NLP datasets, `SentEval` excluded, are also taken from Huggingface. For `SentEval`, we train our models using the original code. We use the same pre-training protocol across all experiments, with a learning rate of 5e-5 and 30 epochs with early stopping with 2 epochs patience. For fine-tuning, we use a learning rate of 1e-5 and 20 epochs. For CV, we use ResNet101 and iNaturalist from Torchvision, while we retrieve ViT and BEiT models from Huggingface, using the version with 12 layers in order to properly compare results with NLP experiments. We use Avalanche (Lomonaco et al., 2021) to run the Continual Pre-Training and fine-tuning. For fine-tuning on the FC dataset, we try few combinations of learning rates ($1e-5$, $1e-4$, $1e-3$) and batch sizes (64, 128, 256) on a held-out validation set built from `CORe50`. We report the best performance in terms of accuracy on the test set. The full experimental setup is described in Appendix A.

4. Results

Our experiments provide strong empirical evidence supporting the hypothesis that *the Continual Pre-Training scenario is less impacted by catastrophic forgetting than the CL one*. In particular, we found the *self-supervised pre-training objective to be the common factor for the resistance to forgetting* in the NLP and CV environments. Our result expands the evidences discussed in Section 2 about the robustness of self-supervised protocols with respect to catastrophic forgetting. We provide additional insights related to different input modalities and by highlighting the *fast remembering* capabilities (1-epoch performance) of the final models on the FC datasets.

Continual pre-training improves performance on the downstream task. We verified that Continual Pre-Training positively impacts on the performance on the downstream `scientific abstracts` classification task (Table 2). That is, we observed that acquiring domain knowledge on `scientific abstracts` helps when solving the abstracts classification task (on held-out data, not seen during continual pre-training).

While the improvement is relatively small, we were able to achieve it by using a smaller number of pre-training samples (20k) with respect to common pre-training datasets like Wikipedia. This supports the idea that, in line with the objective of CL, Continual Pre-Training is able to

Table 3
Accuracy on the entire dataset of `sentiment analysis` (`SA`, top) and `QNLI` (bottom) with RoBERTa model. Continual Pre-Training has been performed sequentially over each experience of `scientific abstracts`. Base refers to the model pre-trained on Wikipedia, while NT refers to the model with vocabulary expansion.

RoBERTa-SA	Accuracy					1-epoch Accuracy				
Base	93.40					92.40				
Exp.	e1	e2	e3	e4	e5	e1	e2	e3	e4	e5
Pretr	93.40	93.15	93.35	93.20	92.90	92.40	91.80	92.30	91.85	92.20
Pretr. NT	93.75	93.70	93.75	93.60	94.10	91.75	91.15	92.00	92.30	92.45
RoBERTa-QNLI										
Base	92.73					91.76				
Exp.	e1	e2	e3	e4	e5	e1	e2	e3	e4	e5
Pretr.	91.96	91.87	91.96	91.76	92.07	90.68	91.32	90.70	90.83	90.85
Pretr. NT	92.09	91.62	91.31	91.45	91.51	91.49	91.05	91.31	89.99	90.99

accumulate knowledge over time without the need to access large amount of data all at once.

As a sanity check, in Table B.11 of Appendix B.3 we showed that, in a domain adaptation setup, pre-training remains beneficial. To this end, we pre-trained the model on the entire corpus of `scientific abstracts` (one step of pre-training) and then we fine-tune it on the held-out samples of the same dataset. The performance increases with respect to a model pre-trained only on Wikipedia. The same phenomenon occurs when starting from a randomly initialized Transformer (RoBERTa) instead of one pre-trained on Wikipedia.

Continual pre-training mitigates forgetting in the FC datasets. Quite surprisingly, we show that after Continual Pre-Training both RoBERTa (Table 3) and BERT (Table 4) achieve almost zero forgetting on the sentiment analysis `SA` and `QNLI` FC datasets. They reach an accuracy comparable to the one originally obtained by the model before Continual Pre-Training. Moreover, a single epoch of gradient descent is sufficient to retain most of the original performance, showing the quick adaptation capabilities of the pre-trained models.

Notably, the additional pre-training steps on domain-specific texts and the expansion of the RoBERTa vocabulary do not worsen the effects of catastrophic forgetting.

We also conducted a broader empirical assessment on a diverse set of NLP tasks by using the `SentEval` benchmark. Fig. 2 shows the downstream performance of BERT and RoBERTa after the entire Continual Pre-Training stream (Table B.10 in Appendix B.2 provides the full numerical results). GloVe and fastText results are used as baselines and are taken from Conneau and Kiela (2018), except on `SNLI` and on all probing tasks, for which they were not available. We computed these missing results using original code and confirmed our findings. The main objective of SentEval is to assess the linguistic properties of the sentence embeddings computed by the language models. From Fig. 2 and Table B.10, we observe how continual pre-training can indeed impact on such linguistic properties. Both BERT and RoBERTa show a decrease in performance when pre-trained continuously (although not a catastrophic one). The largest drop occurs for the WC (Word Content) task with respect to the word embeddings. WC requires to recognize the presence of a set of target words from a given vocabulary. This task is likely easier for a method based on word embeddings, as, unlike in BERT and RoBERTa, it maintains an embedding for each of the target words to be recognized. Moreover, the performance of the Base models (the ones that did not undergo Continual Pre-Training) is also inferior with respect to the word embeddings and closer to the continually pre-trained models. This suggests that the difference in performance on WC is not due to Continual Pre-Training, but rather to the different approaches used to solve the task. Overall, Continual Pre-Training seems to specialize the vocabulary of the language models on the "scientific abstracts" dataset. This can cause a decrease in performance for some SentEval tasks. Interestingly, this change does not affect the performance of the same models on the FC datasets of `SA` and `SNLI`. In Appendix B.4, we briefly showed that in a CL scenario where pre-training is not performed, models suffer from forgetting even when using popular CL strategies. This rules out the hypothesis that the lack of forgetting depends on the specific type of data used in our work.

Self-supervised continual pre-training mitigates forgetting. We found out that self-supervised Continual Pre-Training is the main responsible for the mitigation of forgetting in Continual Pre-Training.

Since all NLP models use the self-supervised masked language modeling task for pre-training, we turned our attention to the CV environment. In fact, ResNet and ViT both use a supervised image classification during pre-training. In contrast, BEiT uses the recent self-supervised protocol of masked image modeling (Bao et al., 2021) (mirroring the NLP setting). We show (Table 5) that BEiT shares the same properties of the NLP transformers, showing little forgetting with respect to the original version on the FC dataset. Also in the CV environment, one epoch of fine-tuning is sufficient to recover the original performance. Interestingly, ResNet and ViT exhibit a qualitatively different trend, with a substantial accuracy drop of around 20% and 13%, respectively. This difference in performance hints towards the fact that *supervised pre-training* in *both* ResNet and ViT is the main responsible of forgetting.

4.0.0.1. The negligible role of the architecture. The type of Transformer used in the experiments does not appear to be a fundamental component: we experimented with larger vision models with 24 layers instead of 12 (Appendix B.1) without being able to appreciate any significant difference with respect to smaller architectures. The difference between convolutional networks like ResNet and attention-based transformers does not seem to have an impact, either. While ResNet sometimes exhibits worse performance than ViT, there is no clear evidence that this kind of model is more susceptible to forgetting.

4.0.0.2. Feature space analysis: supervised pre-training induces large drifts. We verified the coherence of our findings by studying the feature space of the models. We leveraged linear probing for a quantitative analysis and Centered Kernel Alignment (CKA) (Kornblith et al., 2019) for a qualitative analysis. Linear probing (i.e., training only the linear classifier and keeping the rest of the model fixed) is a powerful tool to understand the impact of the learned model representations in terms of catastrophic forgetting (Davari et al., 2022). A model which exhibits forgetting during linear probing is likely to posses features which are not representative of the task. Conversely, a good linear probing performance points to a set of strong features, since it means that the task is linearly separable in that feature space. We adopted this approach for the Continual Pre-Training scenario. In the NLP environment (Table 6), the features built by the models during Continual Pre-Training are robust and do not cause a large deviation of performance with respect to the original pre-trained model. The lower training accuracy with respect to fine-tuning is expected, since linear probing keeps a fixed feature extractor (to evaluate the quality of the learned features) and only trains a minimal set of parameters, corresponding to the final linear classifier only. Therefore, with respect to a fine-tuning process

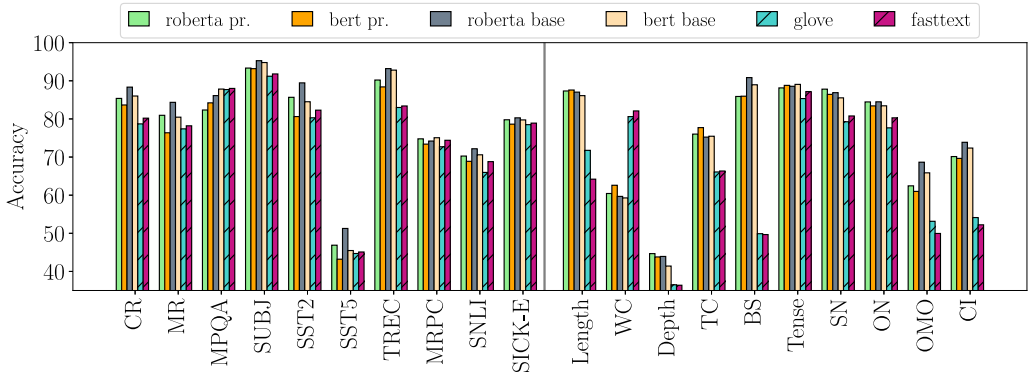


Fig. 2. Accuracy on the 10 transfer tasks (left) and 10 probing tasks (right) of `SentEval`. Transformers are fine-tuned after 5 experiences of pre-training on the `scientific abstracts`. Base refers to the model pre-trained on Wikipedia.

Table 4
Accuracy on the entire dataset of `sentiment analysis (SA)` and `QNLI` with BERT model. Continual Pre-Training has been performed sequentially over each experience of `scientific abstracts`. Base refers to the model pre-trained on Wikipedia.

BERT	Accuracy					1-epoch Accuracy				
Base `SA`	93.05					92.70				
Base `QNLI`	90.43					90.43				
Exp.	e1	e2	e3	e4	e5	e1	e2	e3	e4	e5
Pr. `SA`	92.95	92.90	92.90	92.65	92.45	92.25	92.35	91.90	92.15	91.90
Pr. `QNLI`	90.28	89.75	90.50	89.93	90.01	90.01	89.49	89.31	89.11	89.29

Table 5
Fine-tuning accuracy on the entire dataset of `CORe50`. Pre-training has been performed sequentially over each experience of `iNaturalist`.

Model	Accuracy					1-epoch Accuracy				
ResNet	94.72					94.28				
ViT Base	90.56					90.56				
BEiT Base	90.15					82.51				
Exp.	e1	e2	e3	e4	e5	e1	e2	e3	e4	e5
ResNet Pr.	89.88	81.29	80.82	77.78	74.35	88.40	69.93	70.43	65.91	57.60
ViT Pr.	90.29	81.36	81.47	79.71	77.42	88.48	79.33	78.60	75.01	75.72
BEiT Pr.	88.37	86.45	86.73	87.07	86.46	80.55	78.06	78.88	77.27	77.06

Table 6
Linear probing accuracy on the `sentiment analysis (SA)` and `QNLI` datasets. Pre-training has been performed sequentially over each experience of `scientific abstracts`.

Model	SA					QNLI				
RoBERTa Base	60.05					69.43				
BERT Base	59.85					77.87				
Exp.	e1	e2	e3	e4	e5	e1	e2	e3	e4	e5
RoBERTa Pr.	59.15	59.85	57.00	54.10	58.05	68.88	68.97	67.16	68.08	67.55
BERT Pr.	60.15	59.15	59.35	58.20	56.70	75.62	74.15	72.93	73.37	73.44

Table 7
Linear probing accuracy on the entire dataset of `CORe50`. Pre-training has been performed sequentially over each experience of `iNaturalist`.

Model	Accuracy				
ResNet	82.50				
ViT Base	91.90				
BEiT Base	52.75				
Exp.	e1	e2	e3	e4	e5
ResNet Pr.	61.99	31.02	34.71	26.41	22.01
ViT Pr.	79.38	55.20	57.98	60.49	48.25
BEiT Pr.	52.34	51.71	51.31	53.12	52.51

where all parameters of the model are adapted to the data, the linear probing necessarily shows a lower accuracy.

In the CV environment (Table 7), both ResNet and Vit suffer from forgetting, while BEiT does not (although it reaches a lower absolute accuracy). Following (Hu et al., 2021), we used CKA with linear kernel (Kornblith et al., 2019) to compute layers similarity between the original pre-trained model and its continually pre-trained versions. From Fig. 3, we can see that all models show large correlations across bottom layers (features are not drifting much). Instead, ViT and ResNet show lower correlation values for the final layers than BEiT. This corresponds to a larger drift (full set of results in Appendix B.5) in those layers. These results are compatible with what showed by Madaan et al. (2021) for unsupervised CL, namely that unsupervised models in the CL scenario have larger correlations in the lower layers than supervised ones. Our results further extend this conclusion to Continual Pre-Training, supporting the idea that pre-training acts mainly in the

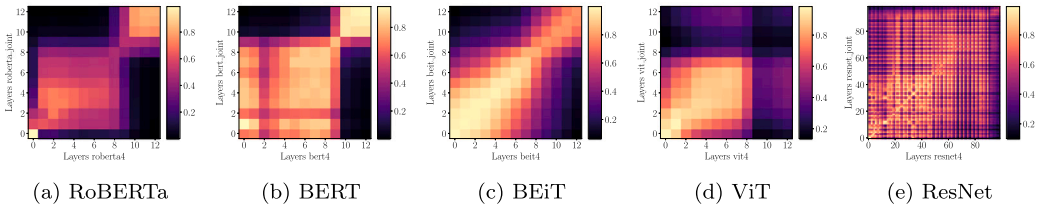


Fig. 3. CKA for RoBERTa (QNLI), BERT (Tweets), BEiT, Vit and ResNet. Pre-trained model h_5^{ds} after the last experience (x axis) is compared with the original pre-trained model h_0^{ds} (y axis). Each row is the layer similarity with respect to each layer of the other model.

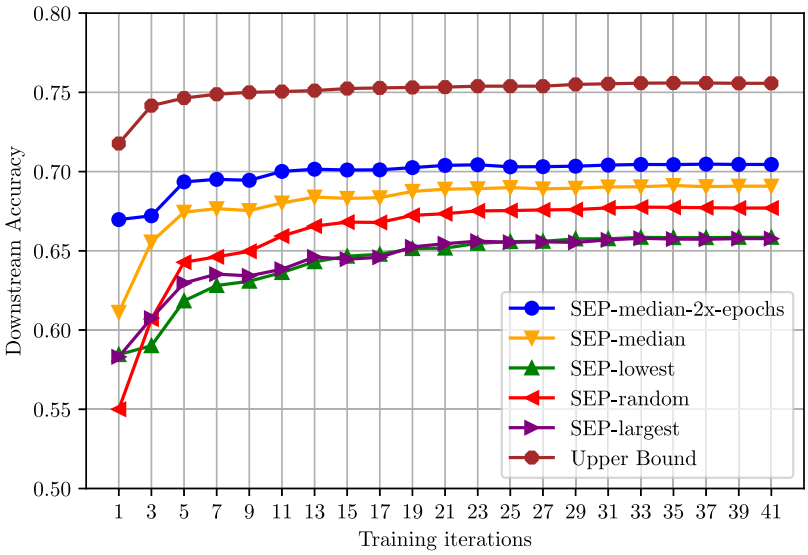


Fig. 4. Downstream accuracy on `abstracts classification` task for SEP. The linear evaluation is the same across all models, but each curve starts from a different pre-trained model. SEP uses 10k examples during pre-training (10% of original pre-training dataset). The upper bound starts from a pre-trained RoBERTa pre-trained on the entire dataset (100k examples). All curves use 30 epochs for pre-training. The blue curve shows that doubling the number of pre-training epochs (i.e., 60 epochs) also reduces the gap with respect to the upper bound.

upper layer of the networks (the ones containing more specific domain knowledge) and that heavy changes in these layers are enough to cause a deterioration of performance on the FC dataset, resulting in forgetting.

4.0.0.3. Sample efficient pre-training.

We introduce a method, Sample Efficient Pre-Training (SEP), that provides an excellent trade-off between the computational cost (training time) of the pre-training stage and its downstream performance. In order to reduce the computational cost, SEP selects important examples from the continual pre-training dataset based on the loss of the current pre-trained model. Given a budget of examples and a pre-trained model h, SEP pre-trains h on a subset of the pre-training dataset containing only the examples within the budget. Like in replay strategies for CL (Merlin et al., 2022), the budget can be chosen according to the computational resources and/or training time available. SEP computes the histogram of the per-example loss distribution on the pre-training dataset and then selects a subset of the examples *for which the loss is closest to the median loss value*. We empirically show that such examples provide the best computational cost vs. downstream accuracy trade-off compared to other, more intuitive choices (e.g., largest loss values). We ran our experiments by pre-training RoBERTa on the `scientific abstracts` pre-training dataset and by linearly evaluating the resulting model on the `scientific abstracts` classification task (held-out set). As in the previously discussed feature space analysis, linear evaluation allows a fair comparison of the hidden representations learned during pre-training. The downstream task consists in predicting the abstract class on a set of held-out examples. The setup for both pre-training and fine-tuning is the same as for the other experiments, except that pre-training is now performed with SEP.

Fig. 4 shows that SEP with 10% of examples only reduces the downstream performance by 7%. Conversely, the training time is reduced by one order of magnitude (10% of the original pre-training time). We also considered SEP with statistics different than the median loss (random sampling, largest/lowest loss values). They exhibit up to a 14% drop in accuracy (twice as SEP with median) compared to using the full pre-training dataset. We used SEP with median in the rest of the experiments.

We found a very promising trade-off by using 50% of the pre-training dataset, hence halving the pre-training time (Fig. 5). In this setup, SEP achieves 73.6% accuracy while full pre-training achieves 75.7% (only a 2.6% decrease in downstream accuracy). We also experimented with SEP considering 5% and 20% of the original pre-training dataset. As expected, these experiments show a consistent decrease in predictive performance as less data is selected. Therefore, SEP can be tuned based on the amount of pre-training examples and computational resources available for each specific application.

As an additional study, we also show that SEP can be easily combined with other pre-training approaches. Fig. 6 reports the downstream accuracy when the first half of the layers of the model is frozen during pre-training. Freezing layers speeds up pre-training and reduces its memory footprint, since no gradient is required for the frozen layers.

Importantly, SEP does not impact on the performance on the FC dataset. For example, RoBERTa pre-trained with SEP still achieves 93% accuracy on the `sentiment analysis` task, on par with the original

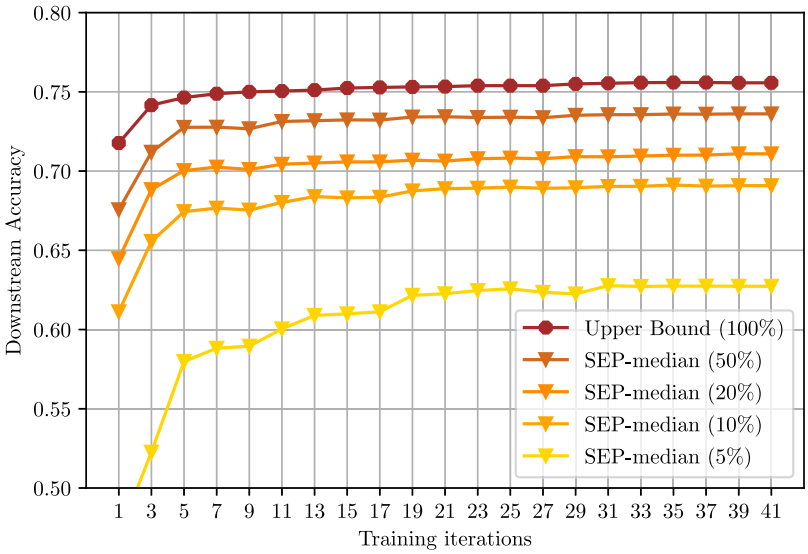


Fig. 5. Downstream accuracy on `abstracts classification` task for SEP pre-trained on different number of examples (in percentage with respect to the total number of examples). The linear evaluation is the same across all models, but each curve starts from a different pre-trained model. Darker colors associated with more pre-training examples. The upper bound starts from a pre-trained RoBERTa pre-trained on the entire dataset (100k examples). All curves use 30 epochs for pre-training.

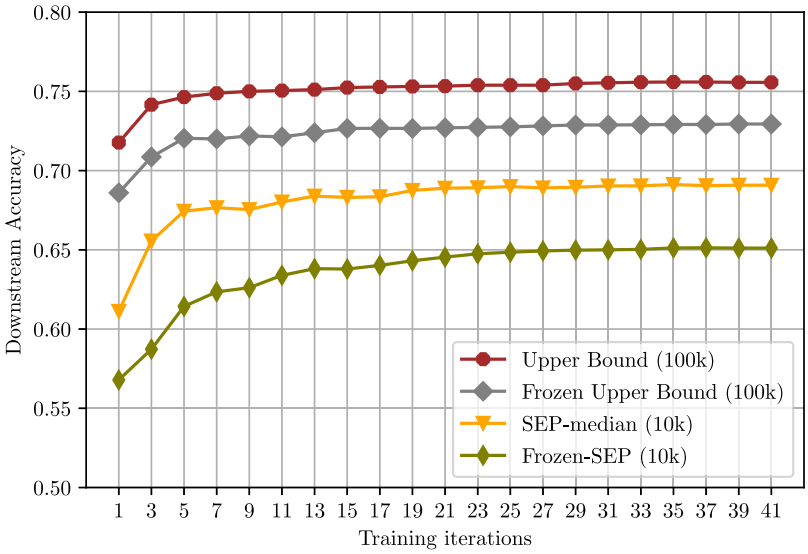


Fig. 6. Downstream accuracy on `abstracts classification` task of SEP when freezing the first half of the model during pre-training. The experimental setup is the same as in Fig. 4. SEP uses 10k examples. As a comparison, we show the upper-bound performance on the entire dataset (100k examples) when freezing the first half of the model during pre-training (Frozen Upper Bound) and when pre-training the entire model (Upper Bound). For comparison, we also report SEP-median from Fig. 4.

RoBERTa model (Table 3). Therefore, our findings about mitigation of forgetting still hold when using SEP.

5. Discussion and limitations

Our empirical evaluation provides evidence that forgetting is mitigated in Continual Pre-Training by the usage of self-supervised pre-training protocols.

We also discovered that fine-tuning for only one epoch allows to recover most of the performance: this is important since an expensive fine-tuning phase might reduce the applicability of Continual Pre-Training in environments with constrained resources. Similarly, our experiments highlight that the continual pre-training stage improves down-stream performance even when leveraging only a small set of data. These two results, taken together, suggests that a model could learn from long streams composed of small chunks of data. Usually, the impact of forgetting would make learning in such long streams more challenging. However, unlike traditional supervised CL, we did not observe any specific correlation between forgetting and stream length in the Continual Pre-Training scenario. Future works may bring this approach to its extreme, by pre-training in an online fashion (one/few samples at a time) from a stream of unstructured data. When needed, the continually pre-trained models could be fine-tuned to solve a set of down-stream tasks. Fine-tuning would only produce a minimal overhead and would only require few annotated samples. Such *Streaming Continual Pre-Training* scenario falls out of the scope of this paper. However, we believe it could provide useful insights on the behavior of pre-trained models. SEP showed that pre-training on substantially less data greatly reduces the pre-training cost, while maintaining a competitive performance in terms of predictive accuracy. Further investigation in this direction can provide robust solutions to perform pre-training on an online stream of data. Knowledge could be built on-the-fly, over time, and when needed.

The Streaming Continual Pre-Training paradigm would likely reduce the strong focus on forgetting currently present in CL. It would also foster the design of novel approaches aimed at improving other important CL metrics, like sample-efficiency and forward transfer to future tasks. While often over-looked in the existing literature, these metrics may find fertile ground in the Continual Pre-Training scenario.

Notwithstanding its advantages, we do not think that Continual Pre-Training should entirely substitute other CL paradigms. In fact, as previously discussed, the properties of Continual Pre-Training do not fit the case in which a single model has to be readily applicable to different tasks. A step of fine-tuning, no matter how quick and efficient, is still required. Continual Pre-Training remains the best option whenever the model can benefit from incorporating unstructured knowledge over time. As recent Machine Learning literature suggests (Bommasani, et al., 2022), this is actually the case for many real-world applications.

Our study will require further validation, especially in terms of the scale of the experiments. For both NLP and CV, we were able to study only a limited number of datasets and configurations. While the computational cost of each experiment was reasonable (each experiment took from few hours - fine-tuning - to around one day - continual pre-training on a single A100), the number of experiments per environment was large. We preferred to thoroughly evaluate few environments rather than trying to address a wide range of different datasets without being able to properly explore them. We are well aware that a comprehensive exploration of Continual Pre-Training in both NLP and CV domains is an ambitious objective, possible only in the context of a broad research program. However, we are confident that this study has shed some light on the subject and clearly pointed towards promising research directions.

6. Conclusion

Continual Pre-Training represents a promising CL scenario. We showed the fundamental role played by pre-training on catastrophic forgetting, in both NLP and CV environments and with different architectures. Our results highlight that forgetting can be effectively mitigated by means of self-supervised pretraining, even with a single epoch of fine-tuning on the FC dataset and without any additional CL strategies. The pre-training stage can be further enhanced via techniques, like SEP, that aim to find an optimal accuracy-efficiency trade-off.

Ultimately, this work opens up the possibility to continually train large pre-trained models in a scalable and efficient way. Much like Deep Learning has advanced by disentangling the representation learning objective from the solution to specific tasks, Continual Pre-Training aims to focus on the incremental development of robust features which are kept updated over time. This is a fundamental property towards the achievement of agents who can truly learn continually in the real-world.

Funding

This work has been partially supported by the H2020 TEACHING project (GA 871385) and by the EU EIC project EMERGE (Grant No. 101070918).

CRediT authorship contribution statement

Andrea Cossu: Writing – original draft, Software, Methodology, Conceptualization. **Antonio Carta:** Writing – review & editing, Supervision, Conceptualization. **Lucia Passaro:** Writing – review & editing, Supervision. **Vincenzo Lomonaco:** Writing – review & editing, Supervision. **Tinne Tuytelaars:** Writing – review & editing, Supervision. **Davide Bacciu:** Writing – review & editing, Resources, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The code and data is publicly available on Github: https://github.com/AndreaCossu/continual-pretraining-nlp-vision.

Appendix A. Extended experimental setup

We describe the experimental setup we adopted in our work for both the NLP environment and the CV environment. All our experiments were run on a single A100 GPU with 80 GB of memory, on a server with 96 cores.

A.1. NLP

The continual pre-training dataset of `scientific abstracts` is taken from GitHub.[1] We selected 10 ArXiv classes to build our continual pre-training stream, namely 'hep-ph', 'astro-ph', 'hep-th', 'quant-ph', 'cond-mat.mes-hall', 'gr-qc', 'cond-mat.mtrl-sci', 'cond-mat.str-el', 'cond-mat.stat-mech' and 'astro-ph.SR'. For both pre-training and downstream fine-tuning, we selected 10,000 abstracts for each of the 10 classes for the training set and 1000 for the test set. Hence, an abstract present in one of the training/test set of continual pre-training or downstream fine-tuning is not present in the other partitions. We chose similar abstract categories since being able to distinguish very different kinds of abstracts may greatly simplify the problem (e.g., one term may be enough to classify the entire abstract). We will publicly release our version of the scientific abstract dataset used in the experiments. The dataset can be easily loaded via Huggingface.

In order to select new tokens for the expansion of RoBERTa vocabulary at each experience of continual pre-training, we trained from scratch a tokenizer on the WikiText dataset (Merity et al., 2016). This tokenizer quickly approximates the tokens present in Wikipedia. We also trained a tokenizer on our `scientific abstracts` dataset and ranked the tokens which were occurring in the latter but not in the former tokenizer. That is, the domain tokens related to the `scientific abstracts` datasets. We selected 426 new tokens for joint training experiments (Appendix B.3) and 39/42/28/30/10 for each of the 5 experiences of continual pre-training.

We added tokens to the tokenizer such that new tokens have precedence over already existing tokens during tokenization process. Within new tokens, we sorted inversely by token length and the precedence is given by the order of addition (First In First Out). The list of new tokens is embedded in the released code. We also ran few experiments (not reported here) by adding with the same procedure sub-word tokens (BPE encoding) instead of word tokens.

The FC dataset `QNLI` is available from Huggingface as part of the GLUE benchmark (Wang et al., 2018). The `sentiment analysis` from tweets dataset is also taken from Huggingface.[2] `Senteval` benchmark is taken from the official codebase.[3]

During linear probing, we removed the feedforward layer right before the classifier. We observed that keeping it frozen yielded a very low training performance. On the other side, fine-tuning it together with the linear classifier did not show the issue but resulted in a non-linear fine-tuning procedure, making it difficult to compare results

[1] R. Stuart Geiger (2020), ArXiV Archive: A Tidy and Complete Archive of Metadata for Papers on arxiv.org, Zenodo: https://doi.org/10.5281/zenodo.1463242.

[2] https://huggingface.co/datasets/emotion

[3] https://github.com/facebookresearch/SentEval

Table B.8
Fine-tuning accuracy on the entire dataset of CORe50 with large transformers. Pre-training has been performed sequentially over each experience of iNaturalist.

Model	Accuracy					1-epoch Accuracy				
ViT Large	92.95					90.77				
BEiT Large	90.41					89.41				
Exp.	e1	e2	e3	e4	e5	e1	e2	e3	e4	e5
ViT Pr.	91.50	89.37	89.93	89.12	87.72	91.39	89.22	89.30	89.12	87.70
BEiT Pr.	89.78	89.90	89.18	88.50	90.09	86.81	85.94	87.50	88.50	88.50

against the CV setup. Therefore, linear probing is performed by taking the representation built for the special CLF token by the last hidden layer of the Transformer and decoding it with a trained linear classifier.

A.2. Computer vision

We adopted the Masked Image Modeling task for self-supervised pre-training with BEiT. Following the original BEiT paper, we leveraged the DALL-E encoder, which is kept fixed during continual pre-training. Experiments which continually pre-train also the encoder may constitute interesting future works.

Following the original TorchVision code, for continual pre-training and fine-tuning on FC dataset with ResNet we used a chain of augmentations: RandomResizedCrop with bilinear interpolation, RandomHorizontalFlip and normalization of mean and standard deviation. On the test sets, we resized the image to 256×256, applied center crop and normalization. ViT uses the same setup without normalization. BEiT applies the ViT setup on the FC dataset only.

For all CKA experiments, we used the Python library,[4] which provides the unbiased minibatch estimator of the CKA.

Appendix B. Additional results

B.1. Experiments with larger CV models

We report in Tables B.8 and B.9 the performance obtained by larger Vision Transformers models with 24 transformers layers for fine-tuning and linear probing, respectively. The results are in line with our main findings with smaller models, except for the ViT, which shows a smaller degree of forgetting. However, the training curves for the large ViT show an unstable trend: the best accuracy is reached usually after one epoch, after which the value quickly degrades to a lower performance. We believe that future works investigating the impact of model depth on our results may shed a light on this phenomenon.

B.2. SentEval results

Table B.10 shows the complete set of results for the SentEval benchmark. We compare the performance of continual pre-training after 5 experiences on scientific abstracts against two baselines (GloVe and fastText) and the original pre-trained model. For RoBERTa, we also provide the results in case of vocabulary expansion. We used one hidden layer of 50 units for probing tasks and logistic regression for the transfer tasks.

B.3. Effect of pre-training on the downstream domain task

Table B.11 shows the accuracy on the entire dataset of scientific abstracts classification after pre-training on the entire dataset of scientific abstracts (held-out sets). Therefore, this setup uses only one step of pre-training to assess its effectiveness on the performance on the downstream task. We show that pre-training is beneficial to the final performance with respect to the original model pre-trained on Wikipedia.

[4] https://github.com/AntixK/PyTorch-Model-Compare

Table B.9
Linear probing accuracy on the entire dataset of CORe50 with large Transformers. Pre-training has been performed sequentially over each experience of iNaturalist.

Model	Accuracy				
ViT Large	82.39				
BEiT Large	52.04				
Exp.	e1	e2	e3	e4	e5
ViT Pr.	85.62	73.75	73.73	75.89	68.27
BEiT Pr.	56.67	55.62	56.12	55.74	56.76

B.4. Results with CL scenario

Table B.12 shows that in a CL scenario, fine-tuning a single model on scientific abstracts classification tasks continuously leads to large forgetting on the same scientific abstracts classification task (held-out dataset), unless CL strategies are employed. We measure the popular ACC metric (Lopez-Paz & Ranzato, 2017) which computes the accuracy on all tasks after training on the last task. The lower its value, the larger the forgetting effect. This shows that, although in the CL scenario we always have a model ready to tackle all the previous tasks without retraining, the loss in terms of performance (accuracy in this case) is very large with respect to the continual pre-training scenario.

B.5. CKA plots

CKA is computed incrementally in minibatches, following (Nguyen et al., 2020). We provide the full set of CKA plots in Fig. C.7 for the NLP environment and in Fig. C.8 for the CV environment. We include the CKA against the original pre-trained model and its continually pre-trained version after each experience of continual pre-training. The upper-right corner of each image represents the upper layers of the models. The correlation is very low only for ViT and ResNet, while it stays large for BEiT, RoBERTa and BERT on all FC datasets.

Appendix C. Extended related works

The Continual Pre-Training scenario appeared very recently in the literature. In this section, we provide a more detailed description of the existing works exploring Continual Pre-Training and the differences with respect to our work. The Related Works section in the main text already provides a brief description but, due to lack of space, we were unable to thoroughly discuss the few existing studies.

Among existing works, the CL scenario used in Jin et al. (2022) constitutes the most similar setup with respect to our definition of continual pre-training. Like us, the authors used a dataset of research papers as pre-training stream and leveraged RoBERTa in their experiments. Differently from us, though, their work is focused on NLP tasks and on the impact that different CL strategies have on the final performance, rather than on the kind of pre-training protocol and on its impact on a separate FC task. Moreover, the downstream tasks used to measure performance are strongly related to the pre-training stream, making it difficult to understand the impact of each pre-training step on catastrophic forgetting. The results they provided show that the amount

Table B.10
Accuracy on 10 transfer and 10 probing tasks from `SentEval`. For comparison, we report the performance of the pre-trained models at the end of pre-training on the last experience (e5) of `scientific abstracts` dataset.

Task	GloVe	fastText	RoBERTa			BERT	
			Base	Pretr.	Pretr. NT	Base	Pretr.
CR	78.70	80.20	88.34	85.38	86.20	86.01	83.66
MR	77.40	78.20	84.35	80.95	80.65	80.46	76.37
MPQA	87.70	88.00	86.12	82.34	82.04	87.83	84.22
SUBJ	91.20	91.80	95.28	93.34	93.36	94.79	93.19
SST2	80.30	82.30	89.46	85.67	85.17	84.51	80.62
SST5	44.70	45.10	51.27	46.88	46.65	45.48	43.21
TREC	83.00	83.40	93.20	90.20	90.40	92.80	88.40
MRPC	72.70	74.40	74.20	74.78	74.67	75.07	73.39
SNLI	65.97	68.80	72.18	70.26	70.69	70.59	68.88
SICK-E	78.50	78.90	80.29	79.78	79.16	79.74	78.63
Length	71.76	64.20	87.03	87.33	86.17	86.11	87.58
Word Content	80.61	82.10	59.68	60.44	62.63	59.28	62.60
Depth	36.50	36.38	43.93	44.67	44.21	41.41	43.80
Top Constituents	66.09	66.34	75.23	76.02	75.91	75.46	77.72
Bigram Shift	49.90	49.67	90.84	85.89	85.75	88.96	85.96
Tense	85.34	87.18	88.56	88.14	87.88	89.06	88.80
Subj Number	79.26	80.78	86.89	87.81	87.44	85.53	86.44
Obj Number	77.66	80.29	84.49	84.46	84.80	83.44	83.42
Odd Man Out	53.15	49.96	68.65	62.45	61.67	65.86	60.99
Coord Inv	54.13	52.23	73.87	70.13	70.33	72.36	69.65

Table B.11
Accuracy on the entire downstream dataset of `scientific abstracts` classification after joint training on the entire pre-training dataset of `scientific abstracts`. The *scratch* term indicates that the model is randomly initialized at the beginning and not pre-trained on Wikipedia.

Model	Accuracy	1-epoch Accuracy
RoBERTa Base	82.25	79.27
BERT Base	82.57	79.37
RoBERTa NT	81.84	77.88
RoBERTa Pr.	82.26	81.01
BERT Pr.	83.49	82.62
RoBERTa Pr. NT	83.51	81.94
RoBERTa scratch	80.48	75.79
RoBERTa scratch Pr.	82.50	81.50

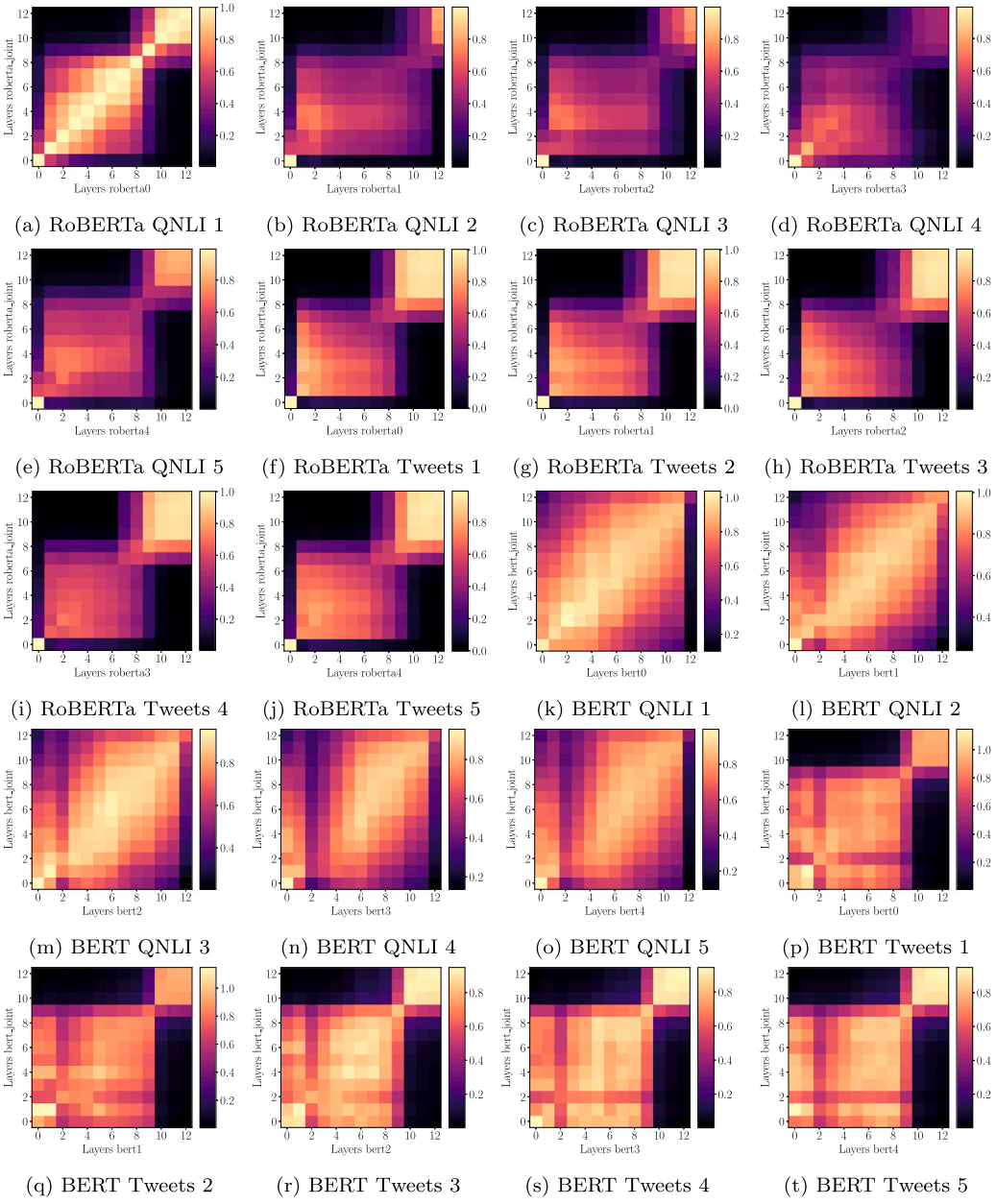
Table B.12
ACC on `scientific abstracts` classification for 5 experiences with RoBERTa. Pre-trained only on the first experience of `scientific abstracts` dataset. Replay memory size is 500. Joint training from Table B.11. ACC around 20.00 means complete forgetting (only the last task is correctly classified).

Model	Joint	Naive	Replay	DSLDA
RoBERTa Base	80.00	19.95	52.94	69.22
RoBERTa Pr.	82.26	19.90	50.78	72.03
RoBERTa Pr. NT	83.51	19.90	51.37	73.32

of forgetting does not depend on the specific CL strategy used. In line with our findings, a naive fine-tuning approach is robust and does not show a catastrophic loss in performance.

The Continual Knowledge Learning (CKL) framework (Jang et al., 2021) shares some similarities with the continual pre-training scenario adopted in our work. The CKL considers a pre-trained model updated continuously and, throughout its training, focuses on different objectives: recognizing invariant knowledge which does not change over time, incorporating new knowledge not present before and updating knowledge which is outdated. The proposed benchmark is entirely based on NLP: it consists of a continual pre-training dataset of news, a "time-invariant knowledge" dataset hand-crafted from relations dataset and an "updated knowledge" and "new knowledge" datasets built from scratch through Amazon Mechanical Turk and validated by a set of external experts. The empirical evaluation provided in the paper is based on a new metric, called FUAR, which condenses the performance of the pre-trained model in these three tasks into a single number. The experiments are conducted on the T5 transformer endowed with existing CL strategies. The authors found out that that parameter expansion methods are amongst the best performing ones, although they require a larger number of parameters with respect to static alternatives.

The study of Hu et al. (2021) focused on the impact of self-supervised pre-training on streaming data subjected to different types of drifts (some of them ascribable to existing CL scenarios like domain-incremental, data-incremental, class-incremental). The authors adopted the MoCo-v2 self-supervised technique for pre-training and a vast set of downstream tasks to measure forgetting, all belonging to CV. Importantly for our work, the authors discussed the problem of catastrophic forgetting. However, differently from our work, the evaluation is performed on the same data used for pre-training instead of relying on a separate downstream task. In our opinion, reporting results on a FC dataset better fits the continual pre-training scenario and delivers a clearer picture of the effect of continual pre-training. Nonetheless, the results obtained by Hu et al. (2021) are compatible with our findings, showing that self-supervised pre-training reduces features drift and mitigates forgetting. The CKA analysis provided by the authors, similar to ours, supports the experimental results.



(a) RoBERTa QNLI 1 (b) RoBERTa QNLI 2 (c) RoBERTa QNLI 3 (d) RoBERTa QNLI 4

(e) RoBERTa QNLI 5 (f) RoBERTa Tweets 1 (g) RoBERTa Tweets 2 (h) RoBERTa Tweets 3

(i) RoBERTa Tweets 4 (j) RoBERTa Tweets 5 (k) BERT QNLI 1 (l) BERT QNLI 2

(m) BERT QNLI 3 (n) BERT QNLI 4 (o) BERT QNLI 5 (p) BERT Tweets 1

(q) BERT Tweets 2 (r) BERT Tweets 3 (s) BERT Tweets 4 (t) BERT Tweets 5

Fig. C.7. CKA for RoBERTa and BERT. Pre-trained models after each experience are compared with the original pre-trained model.

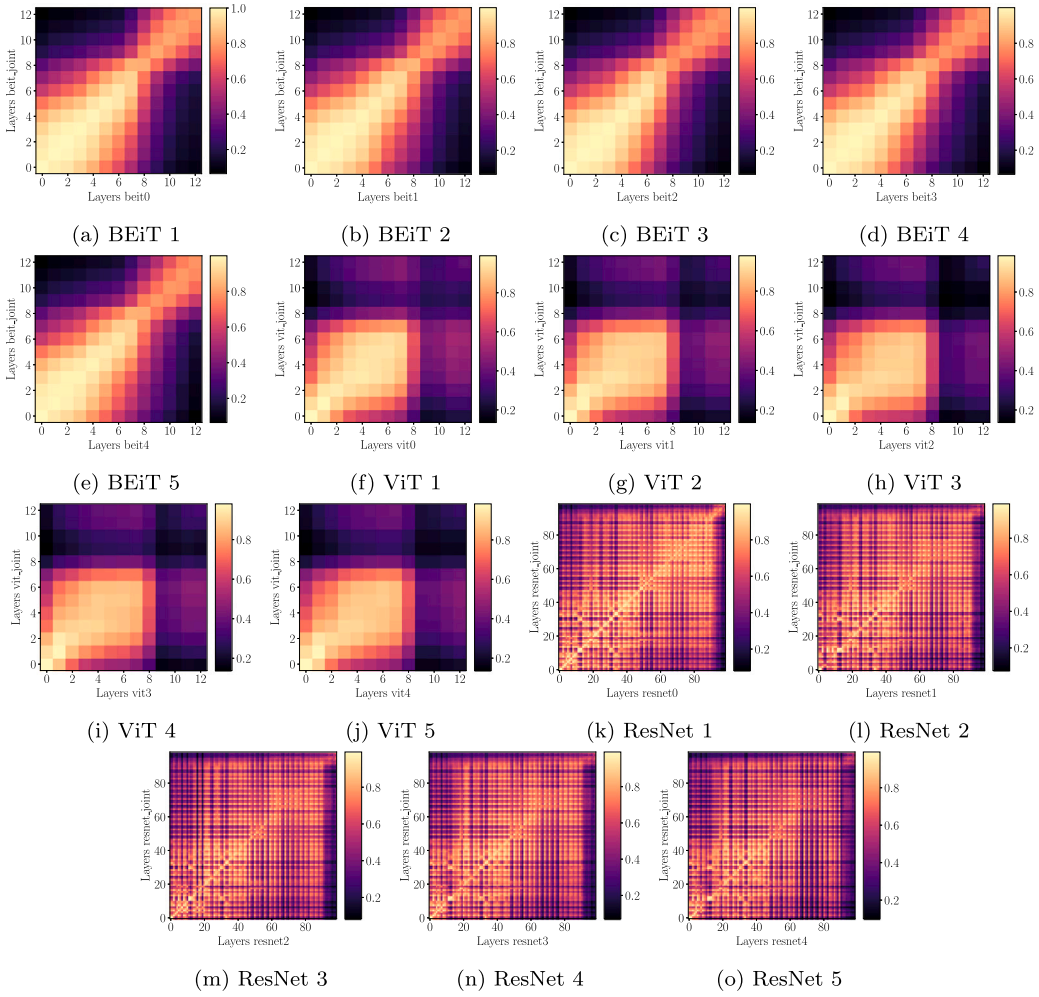


Fig. C.8. CKA for BEiT, Vit and ResNet. Pre-trained models after each experience are compared with the original pre-trained model.

References

Bao, H., Dong, L., Piao, S., & Wei, F. (2021). BEiT: BERT pre-training of image transformers. In *International conference on learning representations*. URL: https://openreview.net/forum?id=p-BhZSz59o4.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., ... Liang, P. (2022). On the opportunities and risks of foundation models. http://dx.doi.org/10.48550/arXiv.2108.07258, arXiv:2108.07258, URL: http://arxiv.org/abs/2108.07258.

Chen, X., Fan, H., Girshick, R., & He, K. (2020). Improved baselines with momentum contrastive learning. arXiv:2003.04297 [cs], URL: http://arxiv.org/abs/2003.04297.

Conneau, A., & Kiela, D. (2018). SentEval: An evaluation toolkit for universal sentence representations. In N. C. C. chair, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, & T. Tokunaga (Eds.), *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA), URL: http://www.lrec-conf.org/proceedings/lrec2018/pdf/757.pdf.

Davari, M., Asadi, N., Mudur, S., Aljundi, R., & Belilovsky, E. (2022). Probing representation forgetting in supervised and unsupervised continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. arXiv:2203.13381.

De Lange, M., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G., & Tuytelaars, T. (2021). A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, http://dx.doi.org/10.1109/TPAMI.2021.3057446.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)* (pp. 4171–4186). Minneapolis, Minnesota: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/N19-1423, URL: https://aclanthology.org/N19-1423.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. In *International conference on learning representations*. URL: https://openreview.net/forum?id=YicbFdNTTy.

Douillard, A., Ramé, A., Couairon, G., & Cord, M. (2022). DyTox: transformers for continual learning with dynamic token expansion. In *IEEE/CVF conference on computer vision and pattern recognition*.

Fini, E., da Costa, V. G. T., Alameda-Pineda, X., Ricci, E., Alahari, K., & Mairal, J. (2022). Self-supervised models are continual learners. In *CVPR*. arXiv:2112.04215.

French, R. (1999). Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, *3*(4), 128–135.

Geiger, R. S. (2019). Arxiv archive: A tidy and complete archive of metadata for papers on arxiv.org, 1993–2019. http://dx.doi.org/10.5281/zenodo.2533436, Zenodo, URL: https://zenodo.org/record/2533436.

Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., & Smith, N. A. (2020). Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 8342–8360). Online: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/2020.acl-main.740, URL: https://aclanthology.org/2020.acl-main.740.

Hadsell, R., Rao, D., Rusu, A. A., & Pascanu, R. (2020). Embracing change: Continual learning in deep neural networks. *Trends in Cognitive Sciences*, http://dx.doi.org/10.1016/j.tics.2020.09.004.

Han, R., Ren, X., & Peng, N. (2021). ECONET: Effective continual pretraining of language models for event temporal reasoning. In *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp. 5367–5380). Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/2021.emnlp-main.436, URL: https://aclanthology.org/2021.emnlp-main.436.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE conference on computer vision and pattern recognition* (pp. 770–778). http://dx.doi.org/10.1109/CVPR.2016.90.

Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. In *NIPS deep learning and representation learning workshop*. URL: http://arxiv.org/abs/1503.02531.

Hu, D., Yan, S., Lu, Q., Hong, L., Hu, H., Zhang, Y., Li, Z., Wang, X., & Feng, J. (2021). How well does self-supervised pre-training perform with streaming data? In *International conference on learning representations*. URL: https://openreview.net/forum?id=EwqEx5ipbOu.

Jang, J., Ye, S., Lee, C., Yang, S., Shin, J., Han, J., Kim, G., & Seo, M. (2022). TemporalWiki: A lifelong benchmark for training and evaluating ever-evolving language models. arXiv:2204.14211 [cs].

Jang, J., Ye, S., Yang, S., Shin, J., Han, J., Kim, G., Choi, S. J., & Seo, M. (2021). Towards continual knowledge learning of language models. In *International conference on learning representations*. URL: https://openreview.net/forum?id=vfsRB5MImo9.

Jin, X., Zhang, D., Zhu, H., Xiao, W., Li, S.-W., Wei, X., Arnold, A., & Ren, X. (2022). Lifelong pretraining: Continually adapting language models to emerging corpora. In *Proceedings of the 2022 conference of the North American chapter of the association for computational linguistics*. arXiv:2110.08534.

Ke, Z., Shao, Y., Lin, H., Konishi, T., Kim, G., & Liu, B. (2023-02-01). Continual pre-training of language models. URL: https://openreview.net/forum?id=m_GDIItaI3o.

Kornblith, S., Norouzi, M., Lee, H., & Hinton, G. (2019). Similarity of neural network representations revisited. In *Proceedings of the 36th international conference on machine learning* (pp. 3519–3529). PMLR, URL: https://proceedings.mlr.press/v97/kornblith19a.html.

Lazaridou, A., Kuncoro, A., Gribovskaya, E., Agrawal, D., Liska, A., Terzi, T., Gimenez, M., d'Autume, C. d. M., Kočiský, T., Ruder, S., Yogatama, D., Cao, K., Young, S., & Blunsom, P. (2021). Mind the gap: Assessing temporal generalization in neural language models. In *Thirty-fifth conference on neural information processing systems*. URL: https://openreview.net/forum?id=73OmmrCfSyy.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., & Kang, J. (2020). BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, *36*(4), 1234–1240. http://dx.doi.org/10.1093/bioinformatics/btz682.

Lesort, T., Lomonaco, V., Stoian, A., Maltoni, D., Filliat, D., & Díaz-Rodríguez, N. (2020). Continual learning for robotics: definition, framework, learning strategies, opportunities and challenges. *Information Fusion*, *58*, 52–68. http://dx.doi.org/10.1016/j.inffus.2019.12.004.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: a robustly optimized BERT pretraining approach. arXiv:1907.11692 [cs], arXiv:1907.11692.

Lomonaco, V., & Maltoni, D. (2017). CORe50: A new dataset and benchmark for continuous object recognition. In S. Levine, V. Vanhoucke, & K. Goldberg (Eds.), *Proceedings of machine learning research*: *vol. 78*, *Proceedings of the 1st annual conference on robot learning* (pp. 17–26). PMLR, URL: http://proceedings.mlr.press/v78/lomonaco17a.html.

Lomonaco, V., Pellegrini, L., Cossu, A., Carta, A., Graffieti, G., Hayes, T. L., De Lange, M., Masana, M., Pomponi, J., van de Ven, G., Mundt, M., She, Q., Cooper, K., Forest, J., Belouadah, E., Calderara, S., Parisi, G. I., Cuzzolin, F., Tolias, A., ... Maltoni, D. (2021). Avalanche: An end-to-end library for continual learning. In *cLVision workshop at CVPR*. http://dx.doi.org/10.1109/CVPRW53098.2021.00399.

Lopez-Paz, D., & Ranzato, M. (2017). Gradient episodic memory for continual learning. In *NIPS*. URL: https://arxiv.org/abs/1706.08840.

Loureiro, D., Barbieri, F., Neves, L., Espinosa Anke, L., & Camacho-collados, J. (2022). TimeLMs: Diachronic language models from Twitter. In *Proceedings of the 60th annual meeting of the association for computational linguistics: system demonstrations* (pp. 251–260). Dublin, Ireland: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/2022.acl-demo.25, URL: https://aclanthology.org/2022.acl-demo.25.

Madaan, D., Yoon, J., Li, Y., Liu, Y., & Hwang, S. J. (2021). Representational continuity for unsupervised continual learning. In *International conference on learning representations*. URL: https://openreview.net/forum?id=9Hrka5PA7LW.

McCloskey, M., & Cohen, N. J. (1989). Catastrophic interference in connectionist networks: the sequential learning problem. In G. H. Bower (Ed.), *vol. 24*, *Psychology of learning and motivation* (pp. 109–165). Academic Press, http://dx.doi.org/10.1016/S0079-7421(08)60536-8, URL: http://www.sciencedirect.com/science/article/pii/S0079742108605368.

Mehta, S. V., Patil, D., Chandar, S., & Strubell, E. (2021). An empirical investigation of the role of pre-training in lifelong learning. arXiv:2112.09153 [cs].

Merity, S., Xiong, C., Bradbury, J., & Socher, R. (2016). Pointer sentinel mixture models. arXiv:1609.07843 [cs], URL: http://arxiv.org/abs/1609.07843.

Merlin, G., Lomonaco, V., Cossu, A., Carta, A., & Bacciu, D. (2022). Practical recommendations for replay-based continual learning methods. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*: *vol. 13374 LNCS*, (pp. 548–559). http://dx.doi.org/10.1007/978-3-031-13324-4_47.

Nguyen, T., Raghu, M., & Kornblith, S. (2020). Do wide and deep networks learn the same things? Uncovering how neural network representations vary with width and depth. In *International conference on learning representations*. URL: https://openreview.net/forum?id=KJNcAkY8tY4.

Parisi, G. I., Kemker, R., Part, J. L., Kanan, C., & Wermter, S. (2019). Continual lifelong learning with neural networks: A review. *Neural Networks*, *113*, 54–71. http://dx.doi.org/10.1016/j.neunet.2019.01.012, URL: http://www.sciencedirect.com/science/article/pii/S0893608019300231.

Ramasesh, V. V., Lewkowycz, A., & Dyer, E. (2021). Effect of scale on catastrophic forgetting in neural networks. In *International conference on learning representations*. URL: https://openreview.net/forum?id=GhVS8_yPeEa.

Ruder, S., Peters, M. E., Swayamdipta, S., & Wolf, T. (2019). Transfer learning in natural language processing. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: tutorials* (pp. 15–18). Minneapolis, Minnesota: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/N19-5004, URL: https://aclanthology.org/N19-5004.

Saravia, E., Liu, H.-C. T., Huang, Y.-H., Wu, J., & Chen, Y.-S. (2018). CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 3687–3697). Brussels, Belgium: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/D18-1404, URL: https://www.aclweb.org/anthology/D18-1404.

Van Horn, G., Mac Aodha, O., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P., & Belongie, S. (2018). The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8769–8778). URL: https://openaccess.thecvf.com/content_cvpr_2018/html/Van_Horn_The_INaturalist_Species_CVPR_2018_paper.html.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*: *vol. 30*, (pp. 5998–6008). Curran Associates, Inc., URL: http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., & Bowman, S. (2018). GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP workshop blackboxNLP: analyzing and interpreting neural networks for NLP* (pp. 353–355). Brussels, Belgium: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/W18-5446, URL: https://aclanthology.org/W18-5446.

Wu, T., Caccia, M., Li, Z., Li, Y.-F., Qi, G., & Haffari, G. (2021). Pretrained language model in continual learning: A comparative study. In *International conference on learning representations*. URL: https://openreview.net/forum?id=figzpGMrdD.

Zhang, R., Gangi Reddy, R., Sultan, M. A., Castelli, V., Ferritto, A., Florian, R., Sarioglu Kayi, E., Roukos, S., Sil, A., & Ward, T. (2020). Multi-stage pre-training for low-resource domain adaptation. In *Proceedings of the 2020 conference on empirical methods in natural language processing* EMNLP, (pp. 5461–5468). Online: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/2020.emnlp-main.440, URL: https://aclanthology.org/2020.emnlp-main.440.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Aligning books and movies: towards story-like visual explanations by watching movies and reading books. In *2015 IEEE international conference on computer vision* (pp. 19–27). http://dx.doi.org/10.1109/ICCV.2015.11.