# Lifelong Language Pretraining with Distribution-Specialized Experts

**Wuyang Chen** [*1] **Yanqi Zhou** [2] **Nan Du** [2] **Yanping Huang** [2] **James Laudon** [2] **Zhifeng Chen** [2] **Claire Cui** [2]

## Abstract

Pretraining on a large-scale corpus has become a standard method to build general language models (LMs). Adapting a model to new data distributions targeting different downstream tasks poses significant challenges. Naive fine-tuning may incur catastrophic forgetting when the overparameterized LMs overfit the new data but fail to preserve the pretrained features. Lifelong learning (LLL) aims to enable information systems to learn from a continuous data stream across time. However, most prior work modifies the training recipe assuming a static fixed network architecture. We find that additional model capacity and proper regularization are key elements to achieving strong LLL performance. Thus, we propose Lifelong-MoE, an extensible MoE (Mixture-of-Experts) architecture that dynamically adds model capacity via adding experts with regularized pretraining. Our results show that by only introducing a limited number of extra experts while keeping the computation cost constant, our model can steadily adapt to data distribution shifts while preserving the previous knowledge. Compared to existing lifelong learning approaches, Lifelong-MoE achieves better few-shot performance on 19 downstream NLP tasks.

## 1. Introduction

Language models (LMs), from word embeddings/vectors (Mikolov et al., 2013), to recurrent neural networks (Sutskever et al., 2014), and to the latest self-attention-based Transformer networks (Vaswani et al.), play increasingly important roles in natural language processing (NLP) tasks, including both language generation and language understanding. Recent works on scaling up both pretraining data and large models (Shazeer et al., 2017; Huang et al., 2019; Kaplan et al., 2020) enable the inference on complicated NLP tasks with much less data, and fewer or even no additional label for downstream tasks. For example, BERT (Xu et al., 2019) and GPT-3 (Brown et al.) demonstrate that for few-shot or even zero-shot generalization on downstream corpus, current LMs only require very few labeled examples to achieve good generalization on unseen tasks. More recently, GLaM (Du et al., 2022) proposes using a sparsely activated mixture-of-experts architecture to scale the model capacity while incurring substantially less training cost compared to dense variants.

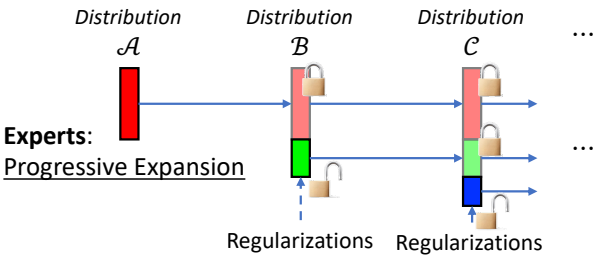


Figure 1: Overview of our Lifelong-MoE method: 1) During pretraining, the expanded experts (and gatings) are specialized for each data distribution; 2) We freeze the pretrained old experts and gatings; 3) We further introduce regularizations to the MoE to avoid the catastrophic forgetting.

Pretraining large language models (LMs) has become the *de facto* standard before adapting NLP models to downstream tasks. This is extremely successful when the pretraining and downstream task are drawn from the same corpus distribution. Most of time, benchmarking large LMs blindly assumes the existence of a static and well-balanced pretraining dataset. While being accurate, the performance of large LMs on downstream tasks heavily relies on the high quality of large-scale pretraining, which is not always guaranteed in the wild for several reasons. First, at the data level, new language corpus (online forum conversations, new wikipedia pages, websites, book chapters, etc.) mostly emerges in a streaming online fashion. That means to keep our pretraining dataset up-to-date, new data distributions will be collected continuously, instead of being statically stored offline in batches. However, in real-world scenarios, sequentially pretraining LMs on new corpus samples with changing distributions will cause catastrophic forgetting on

[*]Work is done during the research internship with Google. [1]The University of Texas at Austin [2]Google. Correspondence to: Yanqi Zhou <yanqiz@google.com>, Nan Du <dunan@google.com>.

*Proceedings of the $40^{th}$ International Conference on Machine Learning*, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

previously learned knowledge. In addition, the collection and maintenance of such high-quality corpora is intensive in manual labor. Second, at the optimization level, pretraining a large LM is time and resource consuming, especially on an increasingly large pretraining corpus. For example, pretraining a GPT-3 model with 280B language tokens requires over 500 TPU hours (Du et al., 2022). As the number of tokens in the pretraining set increases, the pretraining cost will keep rising.

In practice, it is highly preferred to continually pretrain LMs whenever a new corpus is collected, in order to reduce training codecodingst and enhance performance on previously out-of-domain data. Despite its importance, the challenge of continually pretraining a large LM over online data streams is largely under-explored. Lifelong learning (LLL) is a research topic on solving this data/task shifting issue. As opposed to computer vision or robotics, LLL is particularly challenging and nascent in the NLP domain (Greco et al., 2019; Sun et al., 2020c), as natural language is compositional and context-dependent. Prior works in LLL primarily focus on *task-incremental* settings with *boundary-aware* data streams. Starting from the same pretrained checkpoint, these LLL methods are usually evaluated on a sequence of downstream tasks instead of pretraining data distributions (Aljundi et al., 2019). However, this task-level lifelong learning is not the most practically common setting in NLP, because: 1) pretaining is usually agnostic to downstream tasks; 2) as LMs are shown to be few-shot learners, a steam of downstream tasks will incur marginal or zero impact on the pretrained weights. Instead, any shift in pretraining data will pose real forgetting issues.

In this work, we target solving the *data-level* lifelong pretraining with shifting distributions in NLP tasks, especially for large language models. We aim at task-agnostic preservation of domain-specific knowledge from a sequence of online pretraining corpus distributions. We start our method on top of the mixture-of-experts (MoE) (Shazeer et al., 2017; Lepikhin et al., 2021; Du et al., 2022), with an intuition that MoE can increase its model capacity for fitting changing corpus distributions along the online data streams without incurring extra computation cost. Our finding is that, by only introducing extra expert layers plus proper expert regularizations, we can continuously pretrain a mixture-of-experts model on a sequence of data distributions without forgetting old knowledge, and achieve competitive or even better one-shot performance in downstream tasks. The expanded experts will not increase the computation overhead, since they are always sparsely activated and only a fixed number of experts will be selected for each token. Specifically, we show the benefits from three key lifelong learning strategies for MoE: 1) partially expanded experts and gating dimensions; 2) frozen old experts and gatings with only newly expanded ones to be optimized; 3) output-level regularization from previously pretrained knowledge. With these three methods, we aim at creating a well-balanced trade-off between maintaining old knowledge and fitting new distributions. Compared with the dense counterpart, our method can achieve competitive or even better decoding scores on one-shot downstream tasks, including the QA (question answering) task and the translation task. Our contributions are summarized below:

- We propose the first lifelong pretraining framework for large-scale mixture-of-experts (MoE) language models that is agnostic to downstream tasks.
- We progressively expand the number of experts to increase model capacity and fit new pretraining data distributions, and preserve old knowledge by freezing previously trained old experts and gatings.
- We carefully study the output-level regularization to allow dense layers in MoE to fit new data distribution without forgetting old distributions.
- We achieve state-of-the-art decoding scores on downstream one/zero-shot tasks, including the QA task, the translation task, and other language understanding tasks.

## 2. Related Work

**Pretraining and Fine-tuning in Language Models** Deep networks are shown to be powerful in many NLP tasks. Works using recurrent networks such as RNNs and LSTMs (Mikolov et al., 2010; Sutskever et al., 2011) for word/sentence representations (Dai & Le; Kiros et al.) show that language models can improve diverse NLP understanding tasks. More recently, self-attention and transformers (Vaswani et al.) demonstrate that larger models with unsupervised pretraining on unlabeled data can yield significant generalization on NLP problems (Devlin et al., 2019; Yang et al., 2019; Liu et al., 2019; Clark et al., 2020). Abundant computation resources and corpus data makes the pretraining of increasingly large language models possible. These large language models leverage the scaling power of model size and the network's remarkable fitting capacity. Transfer learning based on pretraining and finetuning (Raffel et al., 2020; Houlsby et al., 2019) has been extensively studied and shows good performance on few-shot downstream tasks. The problem of current pretraining and fine-tuning paradigm is that, updating the pretraining dataset will incur repeated heavy re-training cost.

**Sparsely Gated Networks** Despite the success of large and dense language models, training these networks requires significant amounts of computing resources. To keep scaling up NLP models without incurring heavy computational cost, mixture-of-experts (MoE) is recently developed to enable sparse activations in dense layers, and demonstrates

significant advantages. For language modeling and machine translation, Shazeer et al. (2017) shows that they can use a large number of parameters while only activating a small subset for each inference. The choice of dense layers to activate is controlled by a learnable gating function. There is an increasing number of works on scaling sparsely activated MoE architectures (Hestness et al., 2017; Shazeer et al., 2018; Lepikhin et al., 2021; Kudugunta et al., 2021), including Switch-C (Fedus et al., 2021) and GLaM (Du et al., 2022). All these MoE efforts show greatly reduced training energy and computation cost, while still achieving better overall zero, one, and few-shot performance across diverse NLP tasks and domains (Gururangan et al., 2021). In this work, we will show a further advantage of MoE: the expanded experts and gatings can enlarge the model capacity of multiple data distributions without introducing computation overhead. Besides, we only implicitly "assign" experts to different domains instead of any explicit conditions.

**Continual Learning for NLP.** In general, solutions proposed for lifelong learning can be classified into the following categories: i) replay based approaches (Robins, 1995; Rebuffi et al., 2017; Shin et al.; Lopez-Paz & Ranzato; Chaudhry et al., 2018); ii) regularization based approaches (Kirkpatrick et al., 2017; Li & Hoiem, 2018); iii) architecture based approaches (Rusu et al., 2016; Yoon et al., 2018; Mallya & Lazebnik, 2018; Wen et al., 2020). Recently, lifelong learning is drawing attention for NLP problems (Wang et al., 2019b; Biesialska et al., 2020; Sun et al., 2020a; Huang et al., 2021; Hussain et al., 2021; Ahrens et al., 2021; Jin et al., 2021; Lin et al., 2022). A number of lifelong learning methods have also been proposed, including embedding aligned episodic memory replay (Wang et al., 2019a); memory-based parameter adaptation with sparse experience replay (MbPA++) (d'Autume et al., 2019); language modeling for lifelong language learning (Sun et al., 2020b); and meta-learning with sparse experience replay (Holla et al., 2020). The primary challenge to address in LLL literature is to overcome the catastrophic forgetting. However, most works still focus on the traditional settings on sequential downstream tasks, ignoring the fact that pretrained large language models have the capability to quickly adapt to downstream tasks with only a few samples. This task-level lifelong learning is not directly beneficial to most of the real-world scenarios of deployed NLP models, as downstream tasks marginally update model parameters. In contrast, we focus on continually pretraining language models on a steam of changing data distributions (i.e. the data-level lifelong pretraining). This setting is more close to practical scenarios to continually deploying and updating language models.

# 3. Pretraining MoE without Forgetting

Experts and gatings play a vital role in determining MoE's capability of adapting to new data distributions. This motivates us to develop a lifelong pretraining method by only focusing on the customization of experts and gatings. Our strategy is designed as follows: 1) to ensure enough capacity of the MoE whenever it fits a new data distribution, we will expand (and only expand) the number of experts and gating dimensions, keeping the network's depth and width unchanged; 2) to avoid the expanded MoE from overfitting the training data, we will introduce proper regularization on experts and gatings and encourage the preservation of previously learned knowledge.

## 3.1. Model Architecture

We leverage GLaM (Du et al., 2022) as our base model, a family of sparsely activated Mixture-of-Experts (MoE) (Shazeer et al., 2017; Fedus et al., 2021). We are motivated in solving the lifelong pretraining problem in NLP by only introducing more parameters without introducing extra computation overhead (as we will always only use token-wise top-2 experts during both training and inference).

Based on the GShard Transformer (Lepikhin et al., 2021), GLaM replaces the feed-forward component of every other transformer layer with an MoE layer. Each MoE layer consists of a collection of independent feed-forward dense layers as the "experts". A gating function uses softmax to calculate a probability distribution to indicate the preference of the input token to each expert. The dimension of gating's weight equals to the number of experts by the feature size $M$. The experts are sparsely activated: for a given input token, each MoE layer's learnable gating function is trained to activate the token-wise best two experts. During inference, the learned gating network dynamically picks the two best experts for each token. This will results in a model with more capacity while limiting the computation cost.

## 3.2. Progressive Expert Expansion

In the case where only a predefined data distribution exists in the training set, always maintaining a fixed model capacity could be sufficient to fit the pretraining task. However, when the previously learned language representations cannot account for new data distributions, additional parameters need to be introduced to the network. Increasing the model capacity via naively expanding the depth/width of networks will also largely increase the computation cost (Zhou et al., 2012; Rusu et al., 2016; Yoon et al., 2018). To facilitate the memorization of new corpus without incurring extra computations, we choose to leverage the advantage of MoE: we only increase the number of experts while still sparsely activating two experts for each token.

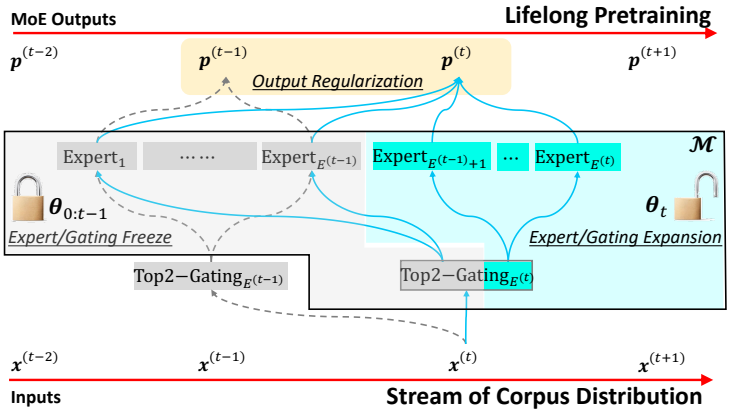


Figure 2: Overview of our lifelong pretraining method for the MoE model ($\mathcal{M}$): 1) When pretraining on each data distribution ($\boldsymbol{x}^{(t)}$), we expand the number of experts and gatings (from $E^{(t-1)}$ to $E^{(t)}$) for larger model capacity; 2) We freeze the pretrained old experts and gatings; 3) We further regularize the MoE on the output level to avoid the catastrophic forgetting. Embedding, dense, and attention layers (omitted in this figure) are shared across all data distributions. See details of our method in Section 3 and pretraining settings in Section 5.1. We omit the interleaving dense layers to make this figure simple and clear.

We need to decide how to expand and initialize new experts and gatings. We empirically observed that randomly initializing expanded experts and gatings leads to poor performance, potentially due to mismatched gradient directions and magnitudes from new experts/gatings and pretrained dense/attention layers. Therefore, inspired by the "Net2WiderNet" approach (Chen et al., 2015), a better way is to initialize each new expert and gating dimension from pretrained ones, helping both the preservation of old knowledge and the warming-up for the subsequent pretraining.

A vanilla expansion strategy would be to *duplicate* the number of experts in order to fully leverage and inherit all the pretrained knowledge. However, this will lead to an exponentially increasing model size, which is not scalable. In our work, we choose to partially expand the number of experts and gating dimensions. We study differ expansion choices, and will show that by expanding a limited number of experts for each data distribution we can achieve competitive performance without further introducing extra model size. That means, we selectively expand (and only expand) the experts when necessary to accommodate incoming new data distribution that is not covered by the older corpora. We do not increase the number of dense layers.

### 3.3. Expert/Gating Regularization

The purpose of our expert/gating expansion is to enlarge the model capacity for incoming new data distributions. At this moment, pretrained experts and gatings store the knowledge about previous distributions. Continuous training will still erase these pretrained knowledge and overfit on the new data, which is not desired. In this section, we propose two approaches to effectively preserve old knowledge.

**Implicit Regularization via Distillation from Old Experts/Gatings** We try to find possible ways to implicitly regularize parameters, including the newly expanded experts, gating dimensions, embeddings, and dense/attention layers. Inspired by (Li & Hoiem, 2017), we choose to distill the knowledge from old experts and gatings. Specifically, denoting the model as $\mathcal{M}$, we minimize the combination of perplexity loss $\mathcal{L}_{\text{Perp}}$ (for the next-token prediction) and the KL divergence $\mathcal{L}_{\text{KL}}$ of outputs from two models:

$$\mathcal{L} = \mathcal{L}_{\text{Perp}} + \lambda \mathcal{L}_{\text{KL}} \tag{1}$$

$$\mathcal{L}_{\text{Perp}} = -\sum_{\boldsymbol{x}_i \in \boldsymbol{X}} \log P\left(\boldsymbol{x}_{i+1} | \mathcal{M}\left(\boldsymbol{x}_{0:i}, \theta_{0:t-1}, \theta_t, \theta_d\right)\right) \tag{2}$$

$$\mathcal{L}_{\text{KL}} = -\sum_{\boldsymbol{x}_i \in \boldsymbol{X}} \mathcal{M}\left(\boldsymbol{x}_i, \theta_{0:t-1}, \theta_d\right) \log\left(\mathcal{M}\left(\boldsymbol{x}_i, \theta_{0:t-1}, \theta_t, \theta_d\right)\right). \tag{3}$$

$\theta_d$ indicates parameters for dense layers that are shared across distributions, $\theta_{0:t-1}$ indicates parameters for old experts and gatings, and $\theta_t$ for parameters of newly expanded experts and gating dimensions. $\boldsymbol{x}$ is the embedding of the current token and $\boldsymbol{X}$ represents the whole corpus of current data distribution. This auxiliary loss $\mathcal{L}_{\text{KL}}$ will implicitly avoid the model parameters from being updated too far from pretrained ones. It is multiplied with a scaling factor $\lambda$ to control its impact to the original pretraining loss value, and we will study different $\lambda$s.

**Explicit Regularization via Partial Experts and Gatings Freezing** To explicitly preserve pretrained knowledge, an intuitive way is to completely freeze neurons specifically responsible for previous data distributions, and only allow parameters for the current distribution to be updated. In our method, the dense/attention layers are always being optimized, since they are trained to fit all data distributions.

Newly expanded experts and gating dimensions are also optimized on the new distribution. Therefore, we only optimize $\mathcal{L}$ regarding $\theta_t, \theta_d$:

$$\theta_t^*, \theta_d^* \leftarrow \arg\min_{\theta_t, \theta_d}(\mathcal{L}) \tag{4}$$

We will study different freezing strategies: freeze old experts, old gating dimensions, or both. Old experts and gatings can be regularized (frozen) since we explicitly associate them with each data distribution. However, since all dense and attention layers are shared across all distributions, we cannot simply freeze their parameters.

# 4. Experiment Setup

Here, we elaborate our datasets, architecture setting, hyperparameters, pretraining procedure, and evaluation protocol.

## 4.1. Training Datasets

To simulate the distribution-level lifelong pretraining setting, we build a sequence of billions of tokens that are representative of a wide range of natural language distributions (both English and non-English), based on the GLaM dataset (Du et al., 2022). We collect webpages and Wikipedia pages (with a combination ratio of $81\% : 19\%$ following (Du et al., 2022)) as our first distribution, denoted as "$\mathcal{A}$". i18n ("internationalization"), the non-English corpus, will be our second distribution "$\mathcal{B}$". Finally, the conversations from public domain social media (Adiwardana et al., 2020) constitutes our third distribution "$\mathcal{C}$". Table 1 shows the details of our data component sizes and mixture weights.

Table 1: Data distributions in our lifelong pretraining set.

| Distribution | Corpus | Tokens (B) |
|---|---|---|
| $\mathcal{A}$ | Wikipedia (19%) | 3 |
| | Filtered Webpages (81%) | 143 |
| $\mathcal{B}$ | i18n | 366 |
| $\mathcal{C}$ | Conversations | 174 |

**Why these three distributions?** We design large gaps between these distributions such that catastrophic forgetting issues can be easily observed. The intuition behind this is that these selections span their contributions to different downstream tasks with less overlap. The English corpus in the distribution $\mathcal{A}$ will contribute to the downstream QA task (Joshi et al., 2017). The dialogs in $\mathcal{C}$ further diversify the English corpus but contribute less to QA. In contrast, the non-English materials in distribution $\mathcal{B}$ has zero (or possibly negative) contribution to English-based tasks and will only benefit to translations. The order of these three distributions is highly related to the study on our downstream tasks: 1) after distribution $\mathcal{A}$, keep pretraining on $\mathcal{B}$ and $\mathcal{C}$ will lead to the forgetting issue on the QA task; 2) after distribution $\mathcal{B}$, keep pretraining on $\mathcal{C}$ will lead to the forgetting issue on the translation task. We show more studies on influences from these distributions to downstream tasks in our Appendix A.

As we will see in Section 5.1 and Figure 3, this design explicitly introduces a challenging scenario for our experiments, leading to sharp transitions and a high risk of forgetting issues between corpus distributions. Similar forgetting issues can also be observed in previous works (e.g. Figure 2 in (Hussain et al., 2021).

## 4.2. Architecture Setting

Table 2 shows the hyperparameter settings of different models, ranging from 145 million to 1.878 billion activated parameters. Here, $E$ is the number of experts (or the dimension of the gating's weight) in each MoE layer, $M$ is the feature/embedding dimension, $H$ is the hidden dimension of the feed-forward layers, $L$ is the number of attention or dense blocks. In addition, $n_{\text{params}}$ is the total number of trainable model parameters, and $n_{\text{act-params}}$ is the number of *activated* model parameters per input token. $n_{\text{heads}}$ is the number of self-attention heads, and $d_{\text{head}}$ is the hidden dimension of each attention head.

## 4.3. Hyperparameters

We use the same learning hyperparameters for all models and for all data distributions. More specifically, We use a maximum sequence length of $1024$ tokens in each mini-batch, and pack each input example to have up to 1 million tokens per batch. The dropout rate is set to $0$ since the number of available tokens in the training corpus is much greater than the number of processed tokens during training. Our optimizer is Adafactor (Shazeer & Stern, 2018) with first-moment decay $\beta_1 = 0$, second-moment decay $\beta_2 = 0.99$ with a $1 - t^{-0.8}$ decay schedule, update clipping threshold of $1.0$, and factored second-moment estimation. When pretraining on each data distribution, we keep the initial learning rate as $0.01$ for the first 10K training steps, and then decay it with inverse square root schedule $\text{lr}\langle t\rangle \propto \frac{1}{\sqrt{t}}$. We use the SentencePiece (Kudo & Richardson, 2018) subword tokenizer with a vocabulary of size of 256K. During training, we use *float32* for model weights and *bfloat16* for activations. The largest Lifelong-MoE model has 1.878B activated parameters with 40 experts (per expert-layer) and is trained on 128 Cloud TPU-V4 chips.

## 4.4. Pretraining Procedure

The pretraining task is to predict the next token in a given sequence with a cross-entropy loss. To simulate the lifelong pretraining setting, unless explicitly stated, otherwise we will sequentially pretrain models on a distribution streams $\mathcal{A} \rightarrow \mathcal{B} \rightarrow \mathcal{C}$. On each distribution, the model will first

Table 2: Sizes and architectures of both our Lifelong-MoE and dense models (Gshard) that we will study in our experiments. All trained models share the same learning hyperparameters described in Session 4.3.

| $E$ | Type | $n_{\text{params}}$ | $n_{\text{act-params}}$ | $L$ | $M$ | $H$ | $n_{\text{heads}}$ | $d_{\text{head}}$ |
|---|---|---|---|---|---|---|---|---|
| 4∼16 | MoE | 241∼573M | 145M | 12 | 768 | 3,072 | 12 | 64 |
| - | Dense | 1.7B | 1.700B | 24 | 2,048 | 8,192 | 16 | 128 |
| 16∼32 | MoE | 11∼22B | 1.878B | | | | | |

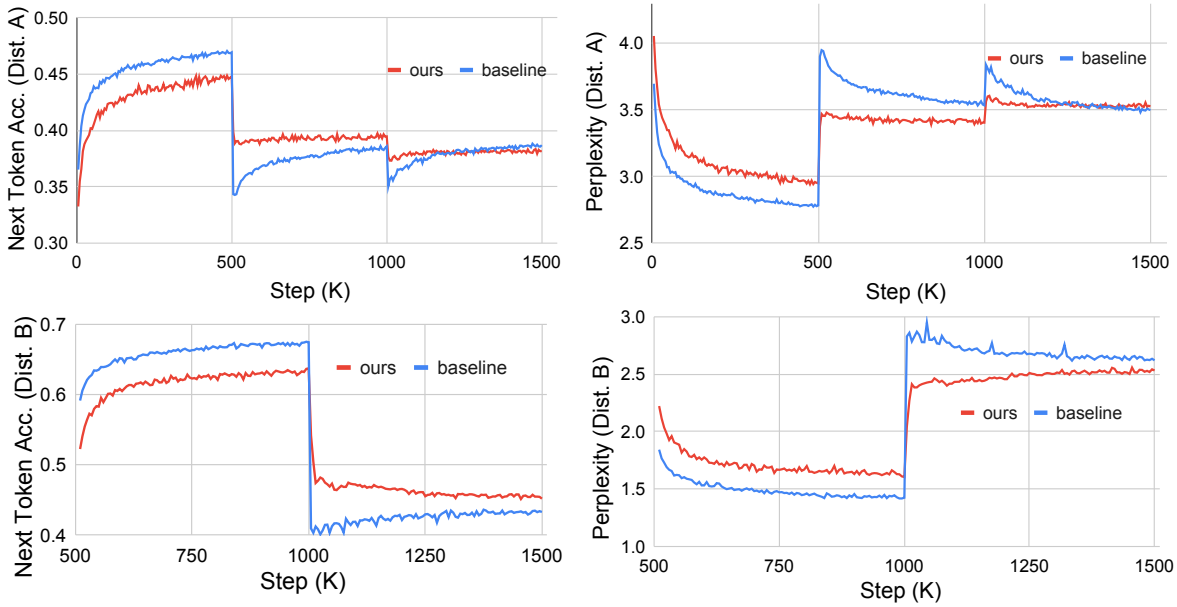


Figure 3: Our method can ameliorate catastrophic forgetting issue in large LMs. Left: next-token accuracy. Right: perplexity. Top/bottom: evaluation on distribution $\mathcal{A}/\mathcal{B}$ during lifelong pretraining. We pretrain models on a sequence of data distributions $\mathcal{A} \rightarrow \mathcal{B} \rightarrow \mathcal{C}$. We train on each data distribution for 500K steps. "0∼500"/"500∼1000" (K) steps in top/bottom rows represent the pretraining phase on $\mathcal{A}/\mathcal{B}$, and subsequent steps stand for forgetting phases (i.e. pretraining on other distributions).

restore the previous checkpoint, and start the pretraining on the new distribution with the same set of hyperparameters. After pretraining on all three distributions, the model will be evaluated on downstream tasks (described below). The next-token accuracy and perplexity on all three distributions will be monitored throughout all pretraining phases.

### 4.5. Downstream Evaluations

**Protocol.** To clearly demonstrate the effectiveness of Lifelong-MoE models, we mainly focus on evaluating the one-shot and zero-shot decoding tasks suggested by Radford et al.; Brown et al.. We randomly draw one example from the target task's training set serving as the only demonstration and context. Such a demonstration is concatenated with the evaluation example with two newlines in between, and then fed into the model.

**Natural Language Generation Tasks.** To allow for an apples-to-apples comparison between GShard (densly connected LM) (Lepikhin et al., 2021) and our method, we follow the evaluation tasks in Brown et al.. We mainly study the one-shot decoding task on TriviaQA (Joshi et al., 2017) and the translation task on WMT16 (Bojar et al., 2016). We compare the language sequences decoded by the models to the ground truth in generative tasks. The performance is measured by the accuracy of exact match (EM) and F1 score, following the standard for each task in Brown et al.. We use beam search with a width of 4 to generate the sequences. For WMT16, we calculate the bleu score (bilingual evaluation understudy).

**Natural Language Understanding Tasks.** Most language understanding tasks require the model to select one correct answer from multiple options. All binary classification tasks are formulated into the form of selecting among two options ('Yes' or 'No'). The prediction is based on the maximum log-likelihood of each option given the context $\log P(\text{option}|\text{context})$ normalized by the token length of each option. On a few tasks, such as ReCoRD (Zhang et al., 2018) and COPA (Gordon et al., 2012), the non-normalized loss can yield better results and thus is adopted. We use the average of the scores reported in all datasets to report the overall few-shot performance of models on NLU tasks. The F1 scores has been normalized to lie between 0 and 100.

# 5. Experiments

## 5.1. Lifelong Pretraining

We first verify that our method can ameliorate the catastrophic forgetting issue during lifelong pertaining (Figure 3). As we pretrain our lifelong-GLaM sequentially on distributions $\mathcal{A} \rightarrow \mathcal{B} \rightarrow \mathcal{C}$, we expect two forgetting phases for $\mathcal{A}$ (when the model is being pretrained on $\mathcal{B}$ and $\mathcal{C}$), and one forgetting phase for $\mathcal{B}$ (when the model is being pretrained on $\mathcal{C}$). For both next-token accuracy (higher the better) and perplexity (lower the better), we can see huge drops of blue lines at phase transitions. However, our method (red lines) can clearly reduce the drop, retaining the pretrained knowledge from previous distributions.

It is worth noting that this experiment is *to our disadvantage*: the baseline has a constant 10 experts (per expert layer) throughout all three pretraining phases, whereas we progressively expand the experts "$4 \rightarrow 7 \rightarrow 10$". That means, during some phases (e.g. evaluation on $\mathcal{A}$ during 500$\sim$1000K steps), our model with less experts (model capacity) can outperforms the GLaM with more experts.

## 5.2. Ablation Study

In this section, we step-by-step study the contributions of expert regularization and expansions to downstream one-shot decoding tasks after the lifelong pretraining.

**Output Regularization.** We first study the choices of different scaling factor ($\lambda$) for our output regularization on a basic GLaM model with four experts. By increasing $\lambda$ from 0, 0.1, to 1, we can improve our F1 score on TriviaQA from 5.93 to 6.96 (row 1$\sim$3 in Table 3). We also find that $\lambda$ larger than 1 will cause unstable pretraining.

**Expert/Gating Freeze.** An intuitive goal to expand experts is to inherit all pretrained experts into newly expanded ones. Therefore, starting from 4 experts, our basic expansion strategy is to expand into 8 and 16 experts.

We now study whether to freeze pretrained experts or gating dimentions during training on new distributions. As shown in Table 3 row 4$\sim$7, freezing either the experts or the gating dimensions are not effective, and only freezing both performs the best.

**Partial Expert Expansion.** Naively duplicating experts and gating dimensions will exponentially increase the model capacity and introduce redundancy. In our experiments, we study how to achieve comparable performance with reduced experts and gating dimensions. We explore different expansion ratios, and observe that with "4$\rightarrow$7$\rightarrow$10" expert expansion (row 9), we can reduce the model size and achieve even slightly better performance than naive expert duplication.

## 5.3. Lifelong-MoE Mitigates Forgetting Issues in Downstream Tasks

Finally, we compare our method with the dense GShard (Lepikhin et al., 2021), GLaM (Du et al., 2022), and classic lifelong learning methods.

**Our Final Large Lifelong-MoE.** We scale up our final large model of over 1 billion parameters (Table 2) based on the best expert expansion strategy we found in the last row in Table 2. We start our lifelong pretraining on distribution $\mathcal{A}$ with 16 experts per expert-layer, and subsequently expand into 28 and 32 for pretraining on distribution $\mathcal{B}$ and $\mathcal{C}$.

**Online L2 Regularization.** The most popular yet simple way of preventing catastrophic forgetting is to regularize the network parameters from deviating too much from its pretrained values using $\ell_2$-regularization (Lin et al., 2022), as follows:

$$\min_{\boldsymbol{W}^{(t)}} \mathcal{L}(\boldsymbol{W}^{(t)}; \boldsymbol{X}^{(t)}) + \lambda \|\boldsymbol{W}^{(t)} - \boldsymbol{W}^{(t-1)}\|_2^2 \quad (5)$$

where $t$ indicates the training step for the current distribution, $\boldsymbol{W}^{(t-1)}$ stands for all weights pretrained on the the previous distribution, and $\lambda$ is the regularization scaling factor. This $\ell_2$-regularization will explicitly enforce the solution $\boldsymbol{W}^{(t)}$ to be close to $\boldsymbol{W}^{(t-1)}$. We set $\lambda = 1$ in our experiment.

**Memory Replay** The other important group of lifelong learning methods is based on retraining on previous samples. Experience Replay (ER) (Rolnick et al.) is a simple yet effective replay method that stores the previous examples into a growing memory module and periodically sample a small subset of the memory as additional training samples for model training. We follow the *most competitive setting* in the recent benchmarking work (Lin et al., 2022), which sampled one mini-batch of previous data per three mini-batch of current data. In our experiment, we always keep 25% historic data when training on a new distribution, i.e., $\mathcal{A} \rightarrow 25\%\mathcal{A} + 75\%\mathcal{B} \rightarrow 25\%(\mathcal{A} + \mathcal{B}) + 75\%\mathcal{C}$.

**Joint Pretraining on Multi-distributions** We can also jointly train a dense LM on our three distributions (with a predefined mixture ratio in (Du et al., 2022), as shown in Table 1). The LM will see all corpus and serve as the oracle model for comparison. We denote this result as "Oracle".

**Results** Our Lifelong-MoE is strong on TriviaQA, WMT16, Ubuntu, and other 19 NLU tasks. These downstream tasks are associated with our pretraining distributions: The corpus of TriviaQA is similar to distribution $\mathcal{A}$ (wikipedia + webpages); WMT16 is similar to distribution $\mathcal{B}$ (i18n); Ubuntu and other NLU tasks are similar to distribution $\mathcal{C}$ (conversations). Therefore, these tasks can faithfully reflect the quality of lifelong pretraining on each

Table 3: Ablation study of our proposed progressive experts expansion and regularization methods. Results are evaluated on downstream TriviaQA few-shot decoding task after pretraining on $\mathcal{A} \rightarrow \mathcal{B} \rightarrow \mathcal{C}$.

| # | Expert Expansion | Freeze | Regularization ($\lambda$) | F1 score |
|---|---|---|---|---|
| 1 | 4→4→4 | N/A | 0 | 5.93 |
| 2 | 4→4→4 | N/A | 0.1 | 5.64 |
| 3 | 4→4→4 | N/A | 1 | 6.96 |
| 4 | 4→8→16 | N/A | 0 | 6.90 |
| 5 | 4→8→16 | Experts | 0 | 6.39 |
| 6 | 4→8→16 | Gatings | 0 | 6.82 |
| 7 | 4→8→16 | Experts + Gatings | 0 | 6.98 |
| 8 | 4→5→6 | Experts + Gatings | 1 | 5.82 |
| 9 | 4→7→10 | Experts + Gatings | 1 | **7.06** |

Table 4: Comparison between our Lifelong-MoE with dense GShard (Lepikhin et al., 2021), GLaM (Du et al., 2022), and classic lifelong learning methods. F1 score is evaluated on TriviaQA. Bleu is evaluated on WMT16.

| Experts | F1 Score | Bleu | Ubuntu | Avg. of 19 NLU Tasks |
|---|---|---|---|---|
| Dense + Online L2 Reg. | 12.99 | 5.66 | 27 | 48.65 |
| Dense + Memory Replay | 14.18 | 7.54 | 26 | 48.65 |
| Dense Oracle | 21.25 | 11.14 | 26 | 49.03 |
| GLaM | 21.76 | 6.97 | 26 | 50.9 |
| Lifelong-MoE (ours) | 20.22 | 19.16 | 27 | 50.26 |

Table 5: Decoding results during sequential pretraining on "$\mathcal{A} \rightarrow \mathcal{B} \rightarrow \mathcal{C}$".

| Method | Phase | TriviaQA F1 | WMT Bleu |
|---|---|---|---|
| Online L2 Reg. | $\mathcal{A}$ | 25.23 | 2.84 |
| | $\mathcal{A} \rightarrow \mathcal{B}$ | 17 (-32.6%) | 20.77 |
| | $\mathcal{A} \rightarrow \mathcal{B} \rightarrow \mathcal{C}$ | 12.99 (-48.5%) | 5.66 (-72.7%) |
| Memory Replay | $\mathcal{A}$ | 25.23 | 2.84 |
| | $\mathcal{A} \rightarrow \mathcal{B}$ | 12.23 (-51.5%) | 12.34 |
| | $\mathcal{A} \rightarrow \mathcal{B} \rightarrow \mathcal{C}$ | 14.18 (-43.7%) | 7.54 (-38.8%) |
| Ours | $\mathcal{A}$ | 33.66 | 4.41 |
| | $\mathcal{A} \rightarrow \mathcal{B}$ | 26.81 (-20.4%) | 22.63 |
| | $\mathcal{A} \rightarrow \mathcal{B} \rightarrow \mathcal{C}$ | 20.22 (-39.9%) | 19.16 (-15.3%) |

distribution. As shown in Table 4, even comparing with the "Dense Oracle", we still achieve better Bleu and NLU scores, with a competitive F1 score on TriviaQA. Note that GLaM achieves better performance on TriviaQA mainly because it starts with much more experts when training on "$\mathcal{A}$".

Moreover, as shown in Table 5, our method not only demonstrates the best decoding results on TriviaQA and WMT, but also achieves the lowest performance drop (shown in parentheses) when switching to new data distributions.

## 6. Conclusion

In this work, we for the first time aim at solving the data-level lifelong pretraining problem, which considers a stream of online changing distributions in pretraining data resources in NLP tasks, especially for large language models. Our results demonstrate that, for an MoE architecture, by only introducing extra expert layers, together with appropriate expert/gating regularizations, we can continuously pretrain the MoE on a sequence of data distributions with preserved old knowledge, achieving competitive or even better pretraining quality for downstream tasks. The expanded experts allocate extra model capacity for new corpus distribution but will not increase computation overhead as the MoE is sparsely activated. With our method, not only the forgetting issue can be largely mitigated during online pretraining, but each new distribution can be fitting with specific experts. We can achieve state-of-the-art performance on downstream NLU decoding tasks under the lifelong pretraining setting. We hope our paper could motivate more works and raise more attentions on realistic NLP scenarios during model pretraining, include the distribution shift in pretraining corpus and online pretraining.

## Acknowledgements

We thank Andrew Dai for the dataset preparation, Tao Lei for research ideas on the conditional computation, and Martin Abadi and Jeff Dean for insightful discussions and general support.

## References

Adiwardana, D., Luong, M., So, D. R., Hall, J., Fiedel, N., Thoppilan, R., Yang, Z., Kulshreshtha, A., Nemade, G., Lu, Y., and Le, Q. V. Towards a human-like open-domain chatbot. *CoRR*, abs/2001.09977, 2020. URL https://arxiv.org/abs/2001.09977.

Ahrens, K., Abawi, F., and Wermter, S. Drill: Dynamic representations for imbalanced lifelong learning. In *International Conference on Artificial Neural Networks*, pp. 409–420. Springer, 2021.

Aljundi, R., Caccia, L., Belilovsky, E., Caccia, M., Lin, M., Charlin, L., and Tuytelaars, T. Online continual learning with maximally interfered retrieval. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2019. Curran Associates Inc. URL https://dl.acm.org/doi/abs/10.5555/3454287.3455350.

Biesialska, M., Biesialska, K., and Costa-jussà, M. R. Continual lifelong learning in natural language processing: A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6523–6541, Barcelona, Spain (Online), 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.574. URL https://aclanthology.org/2020.coling-main.574.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Yepes, A. J., Koehn, P., Logacheva, V., Monz, C., et al. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pp. 131–198, 2016.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, pp. 1877–1901. Curran Associates, Inc.

Chaudhry, A., Ranzato, M., Rohrbach, M., and Elhoseiny, M. Efficient lifelong learning with a-gem. *arXiv preprint arXiv:1812.00420*, 2018.

Chen, T., Goodfellow, I., and Shlens, J. Net2net: Accelerating learning via knowledge transfer. *arXiv preprint arXiv:1511.05641*, 2015.

Clark, K., Luong, M.-T., Le, Q. V., and Manning, C. D. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*, 2020.

Dai, A. M. and Le, Q. V. Semi-supervised sequence learning. In Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*. Curran Associates, Inc.

d'Autume, C. d. M., Ruder, S., Kong, L., and Yogatama, D. Episodic memory in lifelong language learning. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 13143–13152. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/9471-episodic-memory-in-lifelong-language-learnin pdf.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. 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)*, 2019.

Du, N., Huang, Y., Dai, A. M., Tong, S., Lepikhin, D., Xu, Y., Krikun, M., Zhou, Y., Yu, A. W., Firat, O., et al. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning*, pp. 5547–5569. PMLR, 2022.

Fedus, W., Zoph, B., and Shazeer, N. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *CoRR*, abs/2101.03961, 2021. URL https://arxiv.org/abs/2101.03961.

Gordon, A., Kozareva, Z., and Roemmele, M. SemEval-2012 task 7: Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pp. 394–398, Montréal, Canada, 7-8 June 2012. Association for Computational Linguistics. URL https://aclanthology.org/S12-1052.

Greco, C., Plank, B., Fernández, R., and Bernardi, R. Psycholinguistics meets continual learning: Measuring catastrophic forgetting in visual question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3601–3605, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1350. URL https://www.aclweb.org/anthology/P19-1350.

Gururangan, S., Lewis, M., Holtzman, A., Smith, N. A., and Zettlemoyer, L. Demix layers: Disentangling domains for modular language modeling. *arXiv preprint arXiv:2108.05036*, 2021.

Hestness, J., Narang, S., Ardalani, N., Diamos, G. F., Jun, H., Kianinejad, H., Patwary, M. M. A., Yang, Y., and Zhou, Y. Deep learning scaling is predictable, empirically. *CoRR*, abs/1712.00409, 2017. URL http://arxiv.org/abs/1712.00409.

Holla, N., Mishra, P., Yannakoudakis, H., and Shutova, E. Meta-learning with sparse experience replay for lifelong language learning. *arXiv preprint arXiv:2009.04891*, 2020.

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., and Gelly, S. Parameter-efficient transfer learning for NLP. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/houlsby19a.html.

Huang, Y., Cheng, Y., Bapna, A., Firat, O., Chen, D., Chen, M. X., Lee, H., Ngiam, J., Le, Q. V., Wu, Y., and Chen, Z. Gpipe: Efficient training of giant neural networks using pipeline parallelism. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 103–112, 2019.

Huang, Y., Zhang, Y., Chen, J., Wang, X., and Yang, D. Continual learning for text classification with information disentanglement based regularization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2736–2746, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.218. URL https://aclanthology.org/2021.naacl-main.218.

Hussain, A., Holla, N., Mishra, P., Yannakoudakis, H., and Shutova, E. Towards a robust experimental framework and benchmark for lifelong language learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

Jin, X., Lin, B. Y., Rostami, M., and Ren, X. Learn continually, generalize rapidly: Lifelong knowledge accumulation for few-shot learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pp. 714–729, Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.62. URL https://aclanthology.org/2021.findings-emnlp.62.

Joshi, M., Choi, E., Weld, D., and Zettlemoyer, L. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, Vancouver, Canada, 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL https://aclanthology.org/P17-1147.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N. C., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., and Hadsell, R. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114:3521 – 3526, 2017.

Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., and Fidler, S. Skip-thought vectors. In Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*. Curran Associates, Inc.

Kudo, T. and Richardson, J. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *EMNLP*, 2018.

Kudugunta, S., Huang, Y., Bapna, A., Krikun, M., Lepikhin, D., Luong, M.-T., and Firat, O. Beyond distillation: Task-level mixture-of-experts for efficient inference. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pp. 3577–3599, 2021.

Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., Krikun, M., Shazeer, N., and Chen, Z. GShard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=qrwe7XHTmYb.

Li, Z. and Hoiem, D. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Li, Z. and Hoiem, D. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018.

Lin, B. Y., Wang, S., Lin, X. V., Jia, R., Xiao, L., Ren, X., and Yih, W.-t. On continual model refinement in out-of-distribution data streams. *arXiv preprint arXiv:2205.02014*, 2022.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Lopez-Paz, D. and Ranzato, M. Gradient episodic memory for continual learning. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 6467–6476.

Mallya, A. and Lazebnik, S. Packnet: Adding multiple tasks to a single network by iterative pruning. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7765–7773, 2018.

Mikolov, T., Karafiát, M., Burget, L., Cernocký, J. H., and Khudanpur, S. Recurrent neural network based language model. In *INTERSPEECH*, 2010.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. Efficient estimation of word representations in vector space. In Bengio, Y. and LeCun, Y. (eds.), *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013. URL http://arxiv.org/abs/1301.3781.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Rebuffi, S., Kolesnikov, A., Sperl, G., and Lampert, C. H. icarl: Incremental classifier and representation learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 5533–5542. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.587. URL https://doi.org/10.1109/CVPR.2017.587.

Robins, A. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 7(2):123–146, 1995.

Rolnick, D., Ahuja, A., Schwarz, J., Lillicrap, T. P., and Wayne, G. Experience replay for continual learning. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 348–358.

Rusu, A. A., Rabinowitz, N. C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., and Hadsell, R. Progressive neural networks. *ArXiv preprint*, abs/1606.04671, 2016. URL https://arxiv.org/abs/1606.04671.

Shazeer, N. and Stern, M. Adafactor: Adaptive learning rates with sublinear memory cost. *ArXiv*, abs/1804.04235, 2018.

Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q. V., Hinton, G. E., and Dean, J. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=B1ckMDqlg.

Shazeer, N., Cheng, Y., Parmar, N., Tran, D., Vaswani, A., Koanantakool, P., Hawkins, P., Lee, H., Hong, M., Young, C., Sepassi, R., and Hechtman, B. Mesh-tensorflow: Deep learning for supercomputers. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 10435–10444, Red Hook, NY, USA, 2018. Curran Associates Inc.

Shin, H., Lee, J. K., Kim, J., and Kim, J. Continual learning with deep generative replay. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 2990–2999.

Sun, F., Ho, C., and Lee, H. LAMOL: language modeling for lifelong language learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020a. URL https://openreview.net/forum?id=Skgxcn4YDS.

Sun, F., Ho, C., and Lee, H. LAMOL: language modeling for lifelong language learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020b. URL https://openreview.net/forum?id=Skgxcn4YDS.

Sun, F.-K., Ho, C.-H., and Lee, H.-Y. LAMOL: LAnguage MOdeling for Lifelong Language Learning. In *International Conference on Learning Representations (ICLR)*, 2020c. URL https://openreview.net/forum?id=Skgxcn4YDS.

Sutskever, I., Martens, J., and Hinton, G. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pp. 1017–1024, Madison, WI, USA, 2011. Omnipress. ISBN 9781450306195.

Sutskever, I., Vinyals, O., and Le, Q. V. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pp. 3104–3112, 2014.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*. Curran Associates, Inc.

Wang, H., Xiong, W., Yu, M., Guo, X., Chang, S., and Wang, W. Y. Sentence embedding alignment for lifelong relation extraction. 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. 796–806, Minneapolis, Minnesota, June 2019a. Association for Computational Linguistics. doi: 10.18653/v1/N19-1086. URL https://www.aclweb.org/anthology/N19-1086.

Wang, H., Xiong, W., Yu, M., Guo, X., Chang, S., and Wang, W. Y. Sentence embedding alignment for lifelong relation extraction. 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. 796–806, Minneapolis, Minnesota, 2019b. Association for Computational Linguistics. doi: 10.18653/v1/N19-1086. URL https://aclanthology.org/N19-1086.

Wen, Y., Tran, D., and Ba, J. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. In *International Conference on Learning Representations (ICLR)*, 2020. URL https://openreview.net/forum?id=Sklf1yrYDr.

Xu, H., Liu, B., Shu, L., and Yu, P. BERT post-training for review reading comprehension and aspect-based sentiment analysis. 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. 2324–2335, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1242. URL https://aclanthology.org/N19-1242.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

Yoon, J., Yang, E., Lee, J., and Hwang, S. J. Lifelong learning with dynamically expandable networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=Sk7KsfW0-.

Zhang, S., Liu, X., Liu, J., Gao, J., Duh, K., and Durme, B. V. Record: Bridging the gap between human and machine commonsense reading comprehension. *CoRR*, abs/1810.12885, 2018.

Zhou, G., Sohn, K., and Lee, H. Online incremental feature learning with denoising autoencoders. In *Artificial intelligence and statistics*, pp. 1453–1461. PMLR, 2012.

## A. Influence of different distributions on downstream decoding performance.

We also study the influence of different corpus distributions (Table 1) on the downstream TriviaQA F1 decoding task. As shown in Table 6, $\mathcal{A}$ is the most important to TriviaQA, whereas $\mathcal{B}$ will do harms.

Table 6: Influence of different distributions on TriviaQA F1 decoding performance.

| Distribution | F1 |
|---|---|
| $\mathcal{A}$ | 10.2 |
| $\mathcal{B}$ | 4.64 |
| $\mathcal{C}$ | 7.60 |
| $\mathcal{A}+\mathcal{C}$ | 9.29 |