Bayesian Parameter-Efficient Fine-Tuning for Overcoming Catastrophic Forgetting

Haolin Chen, Philip N. Garner

***Abstract*—We are motivated primarily by the adaptation of text-to-speech synthesis models; however we argue that more generic parameter-efficient fine-tuning (PEFT) is an appropriate framework to do such adaptation. Nevertheless, catastrophic forgetting remains an issue with PEFT, damaging the pre-trained model's inherent capabilities. We demonstrate that existing Bayesian learning techniques can be applied to PEFT to prevent catastrophic forgetting as long as the parameter shift of the fine-tuned layers can be calculated differentiably. In a principled series of experiments on language modeling and speech synthesis tasks, we utilize established Laplace approximations, including diagonal and Kronecker-factored approaches, to regularize PEFT with the low-rank adaptation (LoRA) and compare their performance in pre-training knowledge preservation. Our results demonstrate that catastrophic forgetting can be overcome by our methods without degrading the fine-tuning performance, and using the Kronecker-factored approximation produces a better preservation of the pre-training knowledge than the diagonal ones.**

***Index Terms*—parameter-efficient fine-tuning, Bayesian transfer learning, Laplace approximation, catastrophic forgetting.**

I. Introduction

In the context of text-to-speech synthesis (TTS), it has long been of interest to adapt a generic model to a specific domain such as a given speaker identity, language, or emotion. The process is termed *adaptation*; typically the generic model would be well-trained on a large dataset, whereas the (domain-specific) adaptation dataset would be too small to train a bespoke model. Adaptation proved particularly useful in statistical parametric and neural TTS [1], [2], and remains a goal of the recent Blizzard challenge [3]. More recently, the state of the art in TTS is represented by more generic generative models that have arisen in the machine learning community, with advances made in the domains of text [4], [5], vision [6], [7], and audio [8], [9], all feeding through to TTS.

A key paradigm that has emerged in the development and application of such generic models is the pre-training-fine-tuning approach, which involves initially training a model on a large dataset (pre-training) and subsequently fine-tuning it on a task-specific dataset. The paradigm has proven to be highly effective, leading to substantially more accurate and robust outcomes. More recent large pre-trained models have increasingly been equipped with in-context or zero-shot learning capabilities [6], [9], [10]. However, when there are more data available for the target task, fine-tuning is still useful to further improve the performance considerably [11]. Notice that, whilst the vocabulary differs slightly, the goal is the same as for TTS. It follows that current research in fine-tuning provides the means to adapt current TTS models.

The performance gains achieved by large pre-trained models are undeniably linked to their scale. Larger models, with their increased capacity, tend to deliver superior performance. However, as the size of pre-trained models increases, the costs associated with fine-tuning and storing all parameters become prohibitively high, making it practically infeasible. This has led to the study of parameter-efficient fine-tuning (PEFT) techniques [12]–[15], which optimize a small subset of the model parameters (either original parameters or additional ones) while leaving the rest unchanged, significantly reducing computation and storage costs. PEFT techniques have not only facilitated fine-tuning of large pre-trained models on low-resource devices but also enabled the easy sharing and deployment of customized models as far fewer parameters need to be stored and transferred.

Despite the benefits of (parameter-efficient) fine-tuning, it is not without its pitfalls. One significant risk is catastrophic forgetting [16]–[18], where the model loses much of the knowledge it gained during pre-training. This loss can adversely affect the model's ability to generalize to unseen data, a critical aspect of any machine learning model. The phenomenon is even more unfavorable on modern large pre-trained models that are usually multi-functional by training on a diverse range of tasks and data. For example, a language model may forget its general knowledge after continual instruction tuning [19], or hypothetically, the controllability of emotions of a speech synthesizer may be compromised after fine-tuning on a specific voice.

Bayesian learning theory provides a principled solution to overcoming catastrophic forgetting. Considering optimizing the neural network as performing a maximum a posteriori (MAP) estimation of the network parameters given the fine-tuning data, it tries to find the optimal trade-off between the likelihood of the fine-tuning data and the prior knowledge of the pre-trained model, of which the latter is accessible in the form of the posterior over the parameters given the pre-training data. Although the true posterior is intractable, it can be approximated by fitting a Gaussian distribution with a mean equal to the MAP solution and a precision equal to the observed Fisher information. The technique is known as the Laplace approximation [20] and has been thoroughly studied [21]–[24].

Haolin Chen is with the Idiap Research Institute, 1920 Martigny, Switzerland and École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland (email: haolin.chen@idiap.ch).

Philip N. Garner is with the Idiap Research Institute, 1920 Martigny, Switzerland (email: phil.garner@idiap.ch).

In this paper, we demonstrate quite generally that existing Bayesian learning techniques can be applied to PEFT to overcome catastrophic forgetting. Deriving from the Bayesian transfer learning framework, we show that it is viable to regularize the PEFT to preserve the pre-training knowledge as long as the parameter shift of the fine-tuned layers can be expressed in a differentiable manner. Utilizing established Laplace approximation techniques including diagonal [21], [25] and Kronecker-factored [22], [26] approximations of the Hessian, we conduct a series of experiments on language modeling and speech synthesis tasks with the low-rank adaptation (LoRA) [15] to demonstrate the effectiveness and compare the performance of different methods. Specifically, we start from a study on text classification and causal language modeling tasks, the quantitative nature of which allows both rigorous comparison of techniques and comparison with existing literature. We then verify our findings on our target task of speaker adaptation of speech synthesis, where the results are typically more subjective and more onerous to generate. Our results demonstrate that catastrophic forgetting can be overcome by such methods without degrading the fine-tuning performance, and the Kronecker-factored approximations generate a better preservation of the pre-training knowledge than the diagonal ones. Audio samples and source code are available[1].

II. Related Work

A. Laplace Approximation

The Laplace approximation [20] is an established technique in statistics and machine learning to approximate a complex posterior distribution with a Gaussian distribution. This is achieved by identifying the mode of the posterior distribution, which is the maximum a posteriori estimate, and then approximating the distribution around this mode using a second-order Taylor expansion. Two popular kinds of Laplace approximation are the diagonal approximation [21], [25], which only considers the variance of each parameter itself and ignores the interactions between parameters, and the Kronecker-factored approximation [22] that also takes the covariance between parameters within each layer into account. Thanks to the additional information on the off-diagonal elements of the Hessian, the Kronecker-factored approximation has been shown to be more accurate than the diagonal approximation in capturing the loss landscape [26].

The Laplace approximation has been widely applied in neural network optimization (natural gradient descent) [22], [23], [27], [28], improving calibration of neural networks (predictive uncertainty estimation) [24], [29]–[31], and overcoming catastrophic forgetting in transfer and continual learning [21], [26], [32]. In this work, we focus on its application in mitigating catastrophic forgetting in the PEFT setting.

B. Parameter-Efficient Fine-Tuning

There exists a variety of PEFT techniques taking different approaches to adding new trainable components to, or modifying existing parameters of the pre-trained model. Representative PEFT techniques include

1) inserting serial or parallel adapters with a bottleneck structure to the model [12], [33], [34],
2) prepending trainable tokens to the input and hidden states of the transformer block [13], [35],
3) fine-tuning the bias terms inside the model only [14],
4) optimizing the low-rank approximation of the change of weights [15], [36]–[38], and
5) the combination of the above methods [34], [39].

C. Continual Learning

Continual learning aims to enable the model to learn from non-stationary streams of data. [40] categorizes continual learning into three types: task-, domain-, and class-incremental learning. In the context of the adaptation of TTS models, we are interested in the scenario where the pre-trained model is fine-tuned to solve the same task as the pre-training one using data from different domains. This is an example of the domain-incremental type. Despite close ties with continual learning, the scenario concerned aligns better with *transfer learning* and *domain adaptation*. Further constraints that should be considered include that not all pre-training data are accessible and that the pre-training process cannot be replayed. All such constraints limit the usage of techniques designed for task- and class-incremental learning, such as Learning without Forgetting [41] and Synaptic Intelligence [42].

There have been attempts to utilize PEFT techniques, mainly the low-rank adaptation (LoRA), in the continual learning setting. C-LoRA [43] leverages a self-regularization mechanism with LoRA to prevent catastrophic forgetting in continual customization of text-to-image models; O-LoRA [44] continually learns tasks in different low-rank subspaces that are kept orthogonal to each other to minimize interference. For general fine-tuning, [45] proposes to regularize the LoRA weights with Elastic Weight Consolidation [21] when fine-tuning language models on question-answering tasks while preserving their general inference abilities.

III. Bayesian Transfer Learning

A. Framework

The optimization of neural networks can be interpreted as performing a maximum a posteriori (MAP) estimation of the network parameters θ given the training data. In the transfer learning setting, the model has been pre-trained on a task $\mathcal{A}$ using data $\mathcal{D}_\mathcal{A}$, and is then fine-tuned on a downstream task $\mathcal{B}$ using data $\mathcal{D}_\mathcal{B}$. The overall objective is to find the optimal parameters on task $\mathcal{B}$ while preserving the prior knowledge of the pre-trained model on task $\mathcal{A}$. The posterior to be maximized in the MAP estimation can be written as:

$$\begin{aligned} p(\theta|\mathcal{D}_\mathcal{A}, \mathcal{D}_\mathcal{B}) &= \frac{p(\mathcal{D}_\mathcal{B}|\theta, \mathcal{D}_\mathcal{A})p(\theta|\mathcal{D}_\mathcal{A})}{p(\mathcal{D}_\mathcal{B}|\mathcal{D}_\mathcal{A})} \\ &= \frac{p(\mathcal{D}_\mathcal{B}|\theta)p(\theta|\mathcal{D}_\mathcal{A})}{p(\mathcal{D}_\mathcal{B})} \end{aligned} \tag{1}$$

[1] https://github.com/idiap/bayesian-peft

where $\mathcal{D}_\mathcal{B}$ is assumed to be independent of $\mathcal{D}_\mathcal{A}$. Taking a logarithm of the posterior, the MAP objective is therefore:

$$\begin{aligned}\theta^* &= \arg\max_\theta \log p(\theta|\mathcal{D}_\mathcal{A}, \mathcal{D}_\mathcal{B}) \\ &= \arg\max_\theta [\log p(\mathcal{D}_\mathcal{B}|\theta) + \log p(\theta|\mathcal{D}_\mathcal{A})]\end{aligned} \tag{2}$$

The first term $p(\mathcal{D}_\mathcal{B}|\theta)$ is the likelihood of the data $\mathcal{D}_\mathcal{B}$ given the parameters θ, which can be expressed as the training loss function on task $\mathcal{B}$, denoted by $\mathcal{L}_\mathcal{B}(\theta)$. The second term $p(\theta|\mathcal{D}_\mathcal{A})$ is the posterior of the parameters given the pre-training data $\mathcal{D}_\mathcal{A}$. If training the network from scratch, i.e., assuming $\mathcal{D}_\mathcal{A}$ and $\mathcal{D}_\mathcal{B}$ to be one dataset $\mathcal{D}$, this term is usually approximated by a zero-mean isotropic Gaussian distribution, i.e., $p(\theta|\mathcal{D}) = \mathcal{N}(\theta|0, \sigma^2\mathbf{I})$, corresponding to the $\mathcal{L}_2$ regularization. However, for transfer learning, this posterior must encompass the prior knowledge of the pre-trained model to reflect which parameters are important for task $\mathcal{A}$. Despite the true posterior being intractable, $\log p(\theta|\mathcal{D}_\mathcal{A})$ can be defined as a function $f(\theta)$ and approximated around the optimum point $f(\theta_0)$ [20], where θ_0 is the pre-trained values and $\nabla f(\theta_0) = 0$. Performing a second-order Taylor expansion on $f(\theta)$ around θ_0 gives:

$$\begin{aligned}\log p(\theta|\mathcal{D}_\mathcal{A}) &\approx f(\theta_0) + \frac{1}{2}(\theta-\theta_0)^\top \nabla^2 f(\theta_0)(\theta-\theta_0) \\ &= f(\theta_0) + \frac{1}{2}(\theta-\theta_0)^\top \mathbf{H}(\theta-\theta_0)\end{aligned} \tag{3}$$

where $\mathbf{H}$ is the Hessian matrix of $f(\theta)$ at θ_0. The second term suggests that the posterior of the parameters on the pre-training data can be approximated by a Gaussian distribution with mean θ_0 and covariance $\mathbf{H}^{-1}$. Note that the negation of the expected value of the Hessian over the data distribution is the Fisher information matrix (FIM) $\mathbf{F}$, i.e., $\mathbf{F} = -\mathbb{E}_{\mathcal{D}_\mathcal{A}}[\mathbf{H}]$. Following Equation 2, the training objective becomes:

$$\theta^* = \arg\min_\theta [\mathcal{L}_\mathcal{B}(\theta) - \frac{1}{2}(\theta-\theta_0)^\top \mathbf{H}(\theta-\theta_0)] \tag{4}$$

Finally, the loss function that we minimize during fine-tuning can be written as:

$$\mathcal{L}(\theta) = \mathcal{L}_\mathcal{B}(\theta) + \lambda(\theta-\theta_0)^\top \mathbf{F}(\theta-\theta_0) \tag{5}$$

where λ is the regularization strength that determines how much prior knowledge should be preserved during fine-tuning.

B. Diagonal Approximation of the Hessian

Modern neural networks typically have millions to billions of parameters, thus the Hessian, being at least terabytes, is intractable to compute and store. One practical approximation of the Hessian is the diagonal of the Fisher information matrix, i.e., the expected square of the gradients over the data distribution, known as Elastic Weight Consolidation (EWC) [21]. The loss function of EWC is:

$$\mathcal{L}_{EWC}(\theta) = \mathcal{L}_\mathcal{B}(\theta) + \lambda \mathbf{F}_{EWC}(\theta-\theta_0)^2 \tag{6}$$

where $\mathbf{F}_{EWC}$ is the vectorized expected square of the gradients over the distribution of $\mathcal{D}_\mathcal{A}$.

To estimate $\mathbf{F}_{EWC}$, a small subset of the pre-training data $\mathcal{D}_\mathcal{A}$ is sampled and used to compute the gradients of the training loss function $\mathcal{L}_\mathcal{A}(\theta)$ on task $\mathcal{A}$. The final $\mathbf{F}_{EWC}$ is then the average of the square gradients over the sampled data.

A simplified version of EWC, named L2-SP [25], assigns equal importance to all parameters, which is equivalent to assuming that the Fisher information matrix is an identity matrix. The loss function of L2-SP is:

$$\mathcal{L}_{L2-SP}(\theta) = \mathcal{L}_\mathcal{B}(\theta) + \lambda(\theta-\theta_0)^2 \tag{7}$$

L2-SP can be regarded as an extension of the $\mathcal{L}_2$ regularization: instead of zero, it limits the parameters to be close to the pre-trained values during fine-tuning by assigning a Gaussian prior $\mathcal{N}(\theta_0, \sigma^2\mathbf{I})$. Despite being overly simplified, L2-SP proves to be effective in preventing catastrophic forgetting in transfer learning [25], and is particularly useful when the pre-training data are unavailable since no estimation of the FIM is required.

C. Kronecker-Factored Approximation of the Hessian

While first-order approximations such as EWC and L2-SP are simple and efficient, they are not accurate enough to capture the complete loss landscape since they ignore the off-diagonal elements of the Hessian, i.e., the interactions between parameters. To address this issue, recent advances in second-order optimization [22], [23] utilize block-diagonal approximations of the Hessian: the diagonal blocks of the Hessian, corresponding to the interactions between parameters within a single layer, can be approximated as a Kronecker product of two much smaller matrices. This approximation is known as the Kronecker-factored approximate curvature, usually abbreviated as KFAC.

Following [22], we denote the input, the weight, the pre-activations, the non-linear function, and the output of the l-th layer as a_{l-1}, W_l, s_l, ϕ_l and a_l, respectively. For simplicity, we only consider linear layers with no bias term, thus $s_l = W_l a_{l-1}$ and $a_l = \phi_l(s_l)$. We further define $g_l = \frac{\partial \mathcal{L}}{\partial s_l}$ as the gradient of the loss function $\mathcal{L}$ with respect to the pre-activations s_l. The FIM with respect to the weights W_l can be written as:

$$\mathbf{F}^l_{KFAC} = \frac{\partial^2 \mathcal{L}}{\partial^2 \texttt{vec}(W_l)} = A_l \otimes G_l \tag{8}$$

where $\texttt{vec}(W_l)$ is the vectorized form of W_l, $A_l = a_{l-1}a_{l-1}^\top$, $G_l = g_l g_l^\top$ and $\otimes$ is the Kronecker product operator. To calculate the expectation, the two factors are assumed to be independent, thus the expected Kronecker product is approximated as the Kronecker product of the expected factors. Thanks to a property of the Kronecker product, the quadratic penalty term for each layer can be efficiently calculated:

$$(A_l \otimes G_l)\texttt{vec}(\Delta W_l) = \texttt{vec}(G_l \Delta W_l A_l) \tag{9}$$

where $\Delta W_l = W_l - W_l^0$ is the parameter shift from the pre-trained weight W_l^0 of the l-th layer. The overall loss function of KFAC is:

$$\begin{aligned}\mathcal{L}_{KFAC}(\theta) =& \mathcal{L}_\mathcal{B}(\theta) + \\ & \lambda \sum_{l=1}^{L} \texttt{vec}(\Delta W_l) * \texttt{vec}(G_l \Delta W_l A_l)\end{aligned} \tag{10}$$

Despite KFAC's assumption of independence between layers, the most important in-layer parameter interactions are taken into account. It has been demonstrated that KFAC leads to better prior knowledge preservation in continual learning than using a diagonal approximation of the Hessian [26].

IV. Bayesian PEFT

In this work, we aim to show that Bayesian transfer learning can provide a unifying framework for a variety of PEFT techniques. Such an approach not only retains the parameter efficiency of PEFT but also brings a principled approach to regularization, in turn overcoming catastrophic forgetting.

Looking back on Eq. 5, it is not difficult to see that, as long as the parameter shift ΔW_l of the fine-tuned layers can be expressed in a differentiable way, the Bayesian transfer learning framework can be applied to any PEFT technique in the form of modification to the inherent weight of the pre-trained model. The loss function of Bayesian transfer learning with PEFT is therefore:

$$\begin{aligned}\mathcal{L}_{PEFT}(\theta) =&\mathcal{L}_{\mathcal{B}}(\theta)+\\ &\lambda\sum_{l=1}^{L}\texttt{vec}(\Delta W_l)^\top \mathbf{F}_l \texttt{vec}(\Delta W_l)\end{aligned} \tag{11}$$

The most representative PEFT technique that fits this requirement is the low-rank adaptation (LoRA) family. LoRA [15] aims to optimize the low-rank approximation of the change of the original weight matrices based on the hypothesis that the change of weights during fine-tuning has a low intrinsic rank. It is formulated as adding the matrix product of two low-rank matrices to the original weight matrix, i.e., $W_l = W_l^0 + \gamma A_l B_l^\top$, where $W_l^0 \in \mathbb{R}^{d_o \times d_i}$ is the pre-trained weight matrix, γ is a scaling factor, $A_l \in \mathbb{R}^{d_o \times r}$ and $B_l \in \mathbb{R}^{d_i \times r}$ are two low-rank matrices. Therefore, the weight modification (delta weight) of each layer is simply $\Delta W_l = \gamma A_l B_l^\top$. Following Eq. 11, the loss function of Bayesian transfer learning with LoRA is:

$$\begin{aligned}\mathcal{L}_{LoRA}(\theta) =&\mathcal{L}_{\mathcal{B}}(\theta)+\\ &\lambda\sum_{l=1}^{L}\texttt{vec}(\gamma A_l B_l^\top)^\top \mathbf{F}_l \texttt{vec}(\gamma A_l B_l^\top)\end{aligned} \tag{12}$$

Apart from the original LoRA, there exist several variants of LoRA including AdaLoRA [46], which adaptively assigns the rank to the LoRA matrices in each layer, FedPara (LoHa) [36], [38], of which the delta weight is the Hadamard product of two LoRA delta weights, and KronA (LoKr) [37], [38], which generates the delta weight by the Kronecker product of two low-rank matrices. Thanks to the explicit formulation of the delta weight, the LoRA family fits any aforementioned approximation of the Hessian in the Bayesian transfer learning framework. We also note that other PEFT methods such as $(\texttt{IA})^3$ [47] and Orthogonal Butterfly [48], that do not explicitly calculate the delta weight, also fit in the framework, although regularizing these methods may require extra computation and memory. Given that the original LoRA has achieved sufficiently good performance, e.g., it matches the full fine-tuning performance on the GLUE benchmark [15], and other LoRA variants only offer insubstantial improvements, we only employ the original LoRA and focus on the study of regularization methods in our experiments.

V. Experiments: Language Modeling

A. Tasks

We first apply our methods to fine-tuning pre-trained language models with LoRA on two sets of language modeling tasks: text classification and causal language modeling. The reason for this choice of task is twofold: The first is that language models can be evaluated quantitatively; a clear metric is associated with each task. The second is that it allows objective comparison with the wider literature.

1) Text Classification: We select three sentence-pair classification tasks and one single-sentence classification task from the GLUE benchmark [49]. The sentence-pair tasks are: MNLI [50], a natural language inference task of predicting whether a premise entails, contradicts or is neutral to a hypothesis, QQP [51], a paraphrase detection task of predicting whether a pair of sentences are semantically equivalent, and QNLI [52], a question answering task of predicting whether a sentence answers a question. The single-sentence task is SST-2 [53], a sentiment analysis task of predicting whether a sentence has positive or negative sentiment. For all tasks, the fine-tuning performance is reflected by the accuracy on the validation set. The number of training examples in the four selected datasets are MNLI: 393k, QQP: 363k, QNLI: 105k, and SST-2: 67k.

2) Causal Language Modeling: We experiment on the two subsets, WikiText-2 and WikiText-103, of the WikiText dataset [54], a collection of over 100 million tokens extracted from the set of verified good and featured articles on Wikipedia. The number of tokens in WikiText-2 and WikiText-103 are 2.1M and 103M, respectively. The fine-tuning performance is reflected by the perplexity on the validation set, which is shared by the two subsets.

B. Model: OPT

We select the Open Pre-trained Transformers (OPTs) [55] with 350M and 1.3B parameters as the pre-trained models for our experiments. The OPTs are a suite of decoder-only transformers ranging from 125M to 175B parameters pre-trained on a series of large open-access corpora, including a subset of the Pile [56]. Our choice of model sizes is based on those of state-of-the-art pre-trained TTS models ranging from 100M to 1B parameters [9], [57], [58], so that the findings will hopefully provide useful guidance for our target task.

For text classification, a classification head is added on the last token the model generates and trained along with LoRA. This is purely for the simplicity of the implementation, though it could also be done by instruction tuning. For causal language modeling, the model structure remains unchanged.

C. Experimental Details

Implementation. We base our code on the text classification and the causal language modeling examples of the Hugging Face Transformers library [59]. The Bayesian transfer learning

TABLE I
MAIN RESULTS OF LANGUAGE MODELING EXPERIMENTS.

Model	Method	λ	PT PPL	Classification (ACC↑ / PPL↓)				CLM (PPL↓ / PPL↓)	
				MNLI	QQP	QNLI	SST-2	WikiText-2	WikiText-103
OPT-350M	None	-	15.40	83.33% / 523.7	88.97% / 1234	89.79% / 51.11	93.81% / 19.05	13.48 / 20.35	15.21 / 31.74
	L2-SP	10^{-3}		83.35% / 33.65	88.28% / 19.91	89.84% / 23.69	93.72% / 16.66	13.62 / 18.21	15.95 / 20.61
	EWC	10^{4}		83.67% / 18.67	88.73% / 15.94	89.88% / 16.91	93.78% / 15.60	13.55 / 17.17	15.80 / 16.87
	KFAC	10^{6}		84.21% / 17.24	89.28% / 15.80	90.13% / 16.41	93.76% / 15.56	13.59 / 16.22	15.60 / 16.08
OPT-1.3B	None	-	11.18	87.70% / 23.55	90.97% / 16.28	92.59% / 13.45	95.94% / 11.87	9.81 / 13.08	10.53 / 24.32
	L2-SP	10^{-4}		87.77% / 15.66	90.32% / 15.94	92.51% / 13.33	96.10% / 11.78	9.82 / 12.72	10.71 / 15.93
	EWC	10^{4}		87.78% / 11.72	90.62% / 11.32	92.41% / 11.40	96.08% / 11.23	9.81 / 11.89	10.70 / 13.45
	KFAC	10^{5}		87.76% / 11.45	90.64% / 11.25	92.28% / 11.43	96.17% / 11.20	9.84 / 11.73	10.70 / 11.55

* ACC: accuracy, PPL: perplexity, PT PPL: perplexity of pre-trained model on the sampled test set from the Pile, CLM: causal language modeling.

techniques are implemented with the Hugging Face Parameter-Efficient Fine-Tuning (PEFT) library [60].

Hessian estimation. The Hessian estimates are computed on the pre-training task, i.e., the causal language modeling task, and are shared by all fine-tuning tasks. We randomly sample 20,000 examples from the subset of the Pile used to pre-train the OPTs to compute the Hessian estimates for EWC and KFAC, and another 2,000 examples for the evaluation of the pre-training knowledge preservation.

Training and evaluation. All models are trained using the Adam optimizer [61] on each dataset for 3 epochs without weight decay. The learning rate is set to 5×10^{-4} for the 350M model and 2×10^{-4} for the 1.3B model, both with a linear decay schedule. For the text classification tasks, the batch size for all models is set to 32, while for the causal language modeling tasks, the batch size is set to 16 for the 350M model and 8 for the 1.3B model with a context window of 1024 tokens. LoRA is applied to the linear modules that produce the query and value in every self-attention module. The rank and the scaling factor of LoRA are set to 16 and 2 respectively for all models, resulting in the percentage of trainable parameters of the 350M and 1.3B model being 0.473% and 0.239%, respectively. To evaluate the fine-tuning performance, we calculate the accuracy or the perplexity on the validation set for the text classification tasks and the causal language modeling tasks respectively. For MNLI, the "matched" validation set is used. For the evaluation of the pre-training knowledge preservation, we calculate the perplexity on the sampled test set of the Pile. We run a coarse hyper-parameter sweep on the regularization strength λ with a step size of 10 times for each method on each task. The optimal λ is selected balancing the fine-tuning performance and the preservation of pre-training knowledge, typically the point where fine-tuning performance is going to drop greatly if the regularization further strengthens. All experiments were conducted on machines equipped with one NVIDIA RTX3090. The results are averaged over 5 runs with different random seeds.

D. Results and Analyses

The main results are shown in Table I. Note that the method "None" refers to LoRA without regularization. We elaborate our findings from several perspectives.

Catastrophic forgetting. Compared to the pre-trained models, all models fine-tuned without regularization demonstrated significant forgetting of the pre-training knowledge, e.g., the perplexity on the pre-training data increased from 15.40 to 523.7 when fine-tuned on MNLI. Comparing different tasks, it is obvious that the forgetting is more severe when the model is fine-tuned on more data. In terms of model sizes, we notice that larger models tend to forget the pre-training knowledge less than smaller models, which suggests larger models have better resistance to catastrophic forgetting.

Comparison of regularization methods. All regularization methods significantly reduced the loss of pre-training knowledge. Among them, L2-SP underperforms other methods by a large margin, which is reasonable given its over-simplified assumption of diagonal Hessian with equal importance on all parameters. In general, the Kronecker-based methods outperform EWC especially when there is more fine-tuning data, however, the difference is less significant for larger models. This demonstrates that knowledge preservation does benefit from more accurate Hessian estimations.

TABLE II
COMPARISON OF PERFORMANCE WITH VARYING REGULARIZATION STRENGTH OF OPT-350M ON MNLI.

Method	λ	Accuracy↑	Perplexity↓
Pre-trained	-	-	15.40
None	-	83.33%	523.74
L2-SP	10^{-4}	84.52%	52.51
	$\mathbf{10^{-3}}$	**83.35%**	**33.65**
	10^{-2}	81.51%	34.23
EWC	10^{3}	84.11%	26.84
	$\mathbf{10^{4}}$	**83.67%**	**18.67**
	10^{5}	82.03%	16.88
KFAC	10^{5}	84.32%	19.38
	$\mathbf{10^{6}}$	**84.21%**	**17.24**
	10^{7}	83.12%	17.10

Regularization strength. We provide an example of the regularization strength λ sweep for the 350M model fine-tuned on MNLI, which is shown in Table II. As λ increases, the parameters are more constrained to the pre-trained values, thus the fine-tuning performance drops. We select the optimal λ as the one that achieves a fine-tuning performance better than

that of using the original LoRA and has the lowest perplexity on the pre-training data. It can be seen that, compared to KFAC-based methods, the pre-training knowledge preservation of EWC is worse when achieving the same level of fine-tuning performance. We also observe that the fine-tuning benefits from the regularization when λ is small, which can be attributed to the fact that the Hessian estimation introduces a Gaussian prior that better describes the loss landscape than assuming an isotropic Gaussian prior at zero. This suggests that Bayesian transfer learning can lead to better fine-tuning performance as well as overcoming catastrophic forgetting.

TABLE III
COMPARISON OF HESSIAN ESTIMATES WITH VARYING SAMPLES.

Model	Samples	MNLI		WikiText-103	
		EWC	KFAC	EWC	KFAC
OPT-350M	20000	83.67% / 18.67	84.21% / 17.24	15.80 / 16.87	15.60 / 16.08
	2000	83.66% / 18.77	84.30% / 17.64	15.80 / 16.96	15.57 / 16.22
	200	83.71% / 18.50	84.51% / 17.60	15.83 / 16.84	15.47 / 16.79
	20	83.59% / 18.63	84.47% / 21.39	15.83 / 16.96	15.37 / 18.50
OPT-1.3B	20000	87.78% / 11.72	87.76% / 11.45	10.70 / 13.45	10.70 / 11.55
	2000	87.79% / 11.74	87.70% / 11.46	10.70 / 13.36	10.70 / 11.53
	200	87.74% / 11.70	87.76% / 11.54	10.71 / 13.22	10.66 / 11.68
	20	87.85% / 11.67	87.71% / 11.94	10.70 / 13.49	10.59 / 12.53

Hessian estimates with varying samples. We further experiment on Hessian estimates with a reduced amount of pre-training data to investigate the effect of the sample size on the accuracy of the Hessian estimation. The results are shown in Table III. We observe that EWC is more robust to the sample size than KFAC, showing no degradation in pre-training knowledge preservation with Hessian estimates on fewer samples, whereas KFAC demonstrates significant degradation in perplexity on the pre-training data when the sample size is reduced to 20. This can also be corroborated by the increasing fine-tuning performance of KFAC when sample sizes decrease, which signifies less effective regularization. However, for other larger sample sizes, KFAC always outperforms EWC. Overall, the results suggest that KFAC, while being superior to EWC, requires more data to be estimated accurately than EWC, which is reasonable given its additional off-diagonal elements in the Hessian estimation.

Computational cost and memory usage. We compare the computational cost and memory usage of each regularization method in Table IV. Note that the calculation is based on a linear layer with weight $W_l \in \mathbb{R}^{d_o \times d_i}$ using a single sample. The computational cost has two sources: the estimation stage, where a small subset of the pre-training data is sampled to compute the FIM, and the training stage, where the regularization loss is computed at each iteration.

TABLE IV
COMPARISON OF COMPUTATIONAL COST AND MEMORY USAGE.

Method	Computation		Memory
	Estimation	Regularization	
L2-SP	0	$\mathcal{O}(d_o d_i)$	0
EWC	$\mathcal{O}(d_o d_i)$	$\mathcal{O}(d_o d_i)$	$\mathcal{O}(d_i d_o)$
KFAC	$\mathcal{O}(d_o^2 + d_i^2)$	$\mathcal{O}(d_o d_i (d_o + d_i))$	$\mathcal{O}(d_o^2 + d_i^2)$

VI. EXPERIMENTS: SPEECH SYNTHESIS

A. Tasks

Having verified the efficacy of our methods quantitatively and objectively on language modeling tasks, we further apply them to our target application: the fine-tuning of speech synthesis models. Such models are typically more onerous and subjective to evaluate. Our strategy is to demonstrate that the results from the objective evaluation also apply to the more specific target application.

Specifically, we fine-tune a pre-trained zero-shot speech synthesizer with LoRA to adapt it to an unseen speaker. Next, we evaluate the speaker similarity on both the target speaker and other out-of-domain (OOD) speakers, of which the former represents the fine-tuning performance and the latter indicates how well the model preserves the pre-training knowledge. To amplify the effect of catastrophic forgetting, the target speaker and other OOD speakers should be distinct from the pre-training data, thus we select speakers with particular accents for both fine-tuning and evaluation.

We appreciate that the task of evaluating the pre-training knowledge preservation is perhaps of less practical value since there is more interest in getting a similar voice to the target speaker than maintaining the zero-shot performance on other speakers in such a setting. However, this is a necessary compromise owing to several reasons. Firstly, the current publicly available state-of-the-art speech synthesis models mainly target speaker adaptation and are far from being omnipotent, meaning a good zero-shot performance on other speech characteristics is not guaranteed. Further, both the objective and subjective evaluation methods of speaker similarity are well-established, which is not the case for most of the others. Finally, the multi-speaker speech data are easy to obtain, while in other cases the data are not. Despite the limitation, we believe the results will provide practical guidance not only for speaker adaptation on this model but also for many other models and usages where catastrophic forgetting is detrimental to the model's inherent capabilities.

B. Model: StyleTTS 2

To proceed with the proposed tasks, we need an open-access pre-trained TTS model that has good synthesis quality and zero-shot performance for speaker adaptation. StyleTTS 2 [57] is a recently proposed end-to-end TTS model that utilizes style diffusion and adversarial training with a large speech language model to generate human-level expressive and diverse speech. It also achieves a remarkable zero-shot performance though only trained on limited data of 245 hours from the LibriTTS dataset [62] compared to large-scale models such as VALL-E [10], which is trained on 60k hours of data. Initial experiments on zero-shot synthesis show that despite StyleTTS 2 rendering excellent synthesis quality, the synthesized speech tends to lose the accent traits of the target speaker, which can be attributed to the limited training data. Nevertheless, this could be suitable for our experiments as it makes the improvement brought by fine-tuning or the degradation of zero-shot performance more distinguishable.

StyleTTS 2 has a variety of components, many of which are composed of modules that are not compatible with LoRA or whose Hessian estimation needs extra calculation, such as LSTMs and 1D/2D convolutions. However, we found in our initial experiments that only fine-tuning the linear modules in StyleTTS 2 already achieves reasonably good performance. Therefore, for convenience, we only fine-tune the linear modules in all components that are useful for inference of StyleTTS 2.

C. Experimental Details

Implementation. Our code is based on the official implementation of StyleTTS 2 [2]. The same PEFT library for previous experiments is used for applying Bayesian methods and LoRA to the model.

Hessian estimation. We use the official fine-tuning code to calculate the Hessian estimates, during which all training losses are enabled to ensure the gradients are properly back-propagated to all components. Based on the experience from language modeling experiments, we randomly sample 1,000 utterances from the `train-clean-360` subset of the LibriTTS dataset for Hessian estimation to ensure accuracy.

Data. We select `p248`, a female speaker with an Indian accent in the VCTK dataset [63] as the target speaker and randomly split the data into the training set of 356 utterances (approximately 21 minutes) and the test set of 20 utterances. For OOD speakers, we select another 9 speakers (5 females, 4 males) with different accents from VCTK and randomly choose 20 utterances of each speaker as test sets.

Training and inference. We adopt the official multi-stage fine-tuning strategy of 50 epochs described in the code repository for all models, only reducing the batch size from 8 to 2 due to hardware limits. LoRA is applied to the linear modules in all components except for the discriminators and the text aligner which are fully trained and only used during training. The rank and the scaling factor of LoRA are set to 16 and 2 respectively, resulting in an overall percentage of trainable parameters of 1.639% (2.26M of 138M). The fine-tuning is conducted 3 times with different random seeds. For inference, we synthesize test samples using the test sentences for every speaker using the fine-tuned model. All experiments were conducted on the same hardware as previous experiments.

Evaluation. We conduct both objective and subjective evaluations, focusing exclusively on the speaker similarity. Essentially, we use the objective test results as the guideline for our experiments and corroborate our findings with subjective test results. More details are provided in the following sections.

Regularization. Based on the fact that L2-SP is far inferior to other methods, we only experiment with EWC and KFAC in this section. The optimal regularization strength λ is selected using the same criterion as in the language modeling experiments based on the results of the hyperparameter sweep. It is 10^3 for both EWC and KFAC.

D. Objective Evaluation

For the objective evaluation, we use an ECAPA-TDNN [64] speaker verification model [3] to compute the averaged speaker embedding cosine similarity (SECS) score between the synthesized speech and the ground truth on the test set of each speaker. The averaged results of the three runs are shown in Table V. Note that OOD All/Female/Male are the aggregated scores of all/female/male OOD speakers, "Full" and "Linear" stand for full fine-tuning and linear module-only fine-tuning, respectively. We analyze the results from the following perspectives.

Fine-tuning performance. After fine-tuning, the SECS score of the target speaker p248 increases from 0.216 to above 0.6, which manifests that fine-tuning is essential for improving speaker similarity. Without a doubt, the full fine-tuning achieves the best performance. The linear module only fine-tuning ("Linear") and its LoRA-enabled counterpart ("LoRA") perform similarly, however falling behind by a less than 10% margin. This demonstrates the efficacy of the linear module-only fine-tuning scheme. Applying EWC and KFAC on top of LoRA further degrades the performance slightly, with KFAC performing slightly better than EWC.

Zero-shot performance. The overall scores on all OOD speakers clearly demonstrate the catastrophic forgetting, dropping from 0.293 for the pre-trained model to 0.159 for the fully fine-tuned model. Fine-tuning the linear modules only with or without LoRA slightly mitigates the forgetting, suggesting it is necessary to apply additional regularization. Under optimal λ settings, KFAC (0.280) performs substantially better than EWC (0.224), only showing a slight degradation compared to the pre-trained model. The gender breakdown indicates that the fine-tuned model generally achieves a higher similarity on females than males, which can be attributed to the female fine-tuning data. This is confirmed by our test listening that the male speech synthesized by models without regularization severely deteriorates and resembles female speech more. In the speaker breakdown, despite the pre-trained model performing well on some speakers, the fine-tuning degrades similarities on all OOD speakers. One of the reasons for this could be the distinction between the target speaker and the OOD speakers in terms of the accent and the timbre. Moreover, the similarity drops more on speakers that previously had high similarity before fine-tuning. However, in any case, KFAC successfully preserves the zero-shot performance of the model, exceeding EWC by a large margin.

Regularization strength. We provide the λ sweep results in Table VI. It can be seen that under all λ settings, KFAC always achieves better fine-tuning performance and better zero-shot performance preservation than EWC. When matching a good similarity score above 0.6 on the target, EWC shows a significant degradation on OOD speakers. Furthermore, as λ increases, EWC's fine-tuning performance drops faster than KFAC and its zero-shot performance never surpasses that of KFAC. Overall, the results suggest that KFAC helps maintain the zero-shot synthesis ability of the pre-trained model while achieving good fine-tuning performance, whereas EWC suffers

[2]https://github.com/yl4579/StyleTTS2

[3]https://huggingface.co/speechbrain/spkrec-ecapa-voxceleb

TABLE V
MAIN OBJECTIVE TEST RESULTS OF SPEECH SYNTHESIS EXPERIMENTS.

Speaker	Accent	Model					
		Pre-trained	Full	Linear	LoRA	LoRA+EWC	LoRA+KFAC
p248 (f, target)	Indian	0.216	0.695	0.652	0.654	0.633	0.648
OOD All	-	0.293	0.159	0.204	0.203	0.224	0.280
OOD Female	-	0.325	0.184	0.226	0.227	0.247	0.291
OOD Male	-	0.254	0.127	0.175	0.174	0.196	0.267
p225 (f)	English	0.318	0.167	0.241	0.252	0.296	0.352
p234 (f)	Scottish	0.385	0.221	0.257	0.240	0.274	0.297
p261 (f)	Northern Irish	0.448	0.206	0.288	0.281	0.323	0.374
p294 (f)	American	0.267	0.131	0.173	0.181	0.166	0.241
p335 (f)	New Zealand	0.205	0.195	0.171	0.179	0.176	0.188
p245 (m)	Irish	0.324	0.143	0.189	0.209	0.256	0.319
p302 (m)	Canadian	0.262	0.109	0.169	0.170	0.219	0.308
p326 (m)	Australian	0.165	0.132	0.112	0.105	0.082	0.164
p347 (m)	South African	0.262	0.123	0.232	0.210	0.228	0.276

* A suffix (m/f) is added to the speaker name to indicate the gender. Speakers in bold are selected for subjective evaluation.

TABLE VI
COMPARISON OF EWC AND KFAC WITH VARYING REGULARIZATION STRENGTH.

λ	EWC		KFAC	
	Target	OOD	Target	OOD
10^2	0.641	0.213	0.647	0.261
$\mathbf{10^3}$	**0.633**	**0.224**	**0.648**	**0.280**
10^4	0.575	0.270	0.593	0.283
10^5	0.379	0.271	0.491	0.271

from a significant loss of fine-tuning performance when preserving the pre-training knowledge. This is consistent with the results of language modeling experiments on the smaller 350M model, however here the phenomenon is more pronounced.

E. Subjective Evaluation

Sample selection. Having verified the efficacy with objective tests, we further conduct a subjective evaluation to corroborate our findings. One of the concerns is that the synthesized samples of OOD speakers usually result in a much lower perceptual similarity than those of the target speaker, making it difficult to distinguish the performance of low-performing models. In this regard, we select two OOD speakers that have the highest SECS scores and the most difference among models in each gender for the listening test, which are p225, p261, p245, and p302. 10 samples of the target speaker and 5 samples of each OOD speaker are randomly selected, totaling 10 female samples and 10 male samples of the OOD speakers for each model. We also add a ground truth (GT) group for comparison.

Implementation. We hired 20 native English speakers from the United Kingdom on the Prolific [4] crowd-sourcing platform to rate the speaker similarity between the synthesized speech and the reference on a 5-point scale (5: completely same speaker, 4: mostly similar, 3: equally similar and dissimilar, 2: mostly dissimilar, 1: completely different speaker), using a modified Degradation Category Rating (DCR) method based on the P.808 toolkit [65]. The reference is a random recording of the speaker with spoken content different from that of the test sample and is bound to each test sample. The averaged result is often referred to as the Similarity Mean Opinion Score (SMOS).

[4]https://www.prolific.com

TABLE VII
SUBJECTIVE TEST RESULTS WITH 95% CONFIDENCE INTERVAL.

Model	Target	OOD All	OOD Female	OOD Male
GT	4.46 ± 0.11	4.59 ± 0.07	4.65 ± 0.10	4.52 ± 0.11
Pre-trained	1.90 ± 0.15	2.22 ± 0.13	2.36 ± 0.20	2.08 ± 0.17
Linear	4.06 ± 0.16	1.50 ± 0.10	1.83 ± 0.17	1.18 ± 0.07
LoRA	3.86 ± 0.16	1.48 ± 0.09	1.83 ± 0.17	1.13 ± 0.06
LoRA+EWC	3.60 ± 0.14	1.51 ± 0.10	1.77 ± 0.17	1.26 ± 0.09
LoRA+KFAC	3.81 ± 0.16	2.08 ± 0.13	2.31 ± 0.20	1.85 ± 0.16

Results and analyses. The results are shown in Table VII. In general, the subjective test results corroborated our findings from objective tests, hence we mainly comment on the discrepancies between the two tests. For the target speaker, fine-tuning linear modules ("Linear") achieves an SMOS of 4.06, which is a significant improvement from the pre-trained model of 1.90 and is considerably good given the ground truth of 4.46. Different from the objective test results, the LoRA-only model shows a disadvantage of 0.20 compared to "Linear", meaning fine-tuning a low-rank representation does degrade the fine-tuning performance for this model. The small difference between EWC and KFAC shown by SECS scores is actually perceivable, indicated by a difference of 0.21 in SMOS. In terms of zero-shot performance, EWC's preservation effect is not reflected on SMOS considering all OOD speakers, which is in contrast with KFAC. The gender breakdown shows a slight degradation on male OOD speakers for the LoRA with KFAC model, suggesting KFAC did not perfectly preserve the zero-shot performance of the pre-trained model as the SECS scores showed.

VII. Conclusions

In this work, we explored applying Bayesian learning techniques to parameter-efficient fine-tuning to overcome catastrophic forgetting. We started from the derivation of the Bayesian transfer learning framework and demonstrated that PEFT could be regularized to preserve the pre-training knowledge as long as the parameter shift of the fine-tuned layers could be calculated differentiably. We then conducted experiments with LoRA on both language modeling and speech synthesis tasks to verify the efficacy of the proposed methods and compared the performance of different Laplace approximations. Our results show that catastrophic forgetting can be overcome by our methods without degrading the fine-tuning performance. Furthermore, the results on both tasks suggest using the Kronecker-factored approximations of the Hessian produces more effective preservation of the pre-training knowledge and better fine-tuning performance than the diagonal approximations, even though the former requires more data to be estimated accurately.

Current limitations of this work include that it cannot be applied to PEFT techniques that add new components to the model such as bottleneck adapters; however this is not a serious concern given suitable techniques like LoRA already provide good fine-tuning performance. Further, it is only feasible when at least part of the pre-training data is accessible. Finally, the efficacy on larger (TTS) models has not been verified due to the inaccessibility to these models and hardware constraints. We would like to evaluate our methods on larger TTS models when they become publicly available in the future.

Acknowledgments

This project received funding under NAST: Neural Architectures for Speech Technology, Swiss National Science Foundation grant 185010.

References

[1] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, "Analysis of speaker adaptation algorithms for hmm-based speech synthesis and a constrained smaplr adaptation algorithm," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 1, pp. 66–83, 2009.

[2] S. Ö. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," in *Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, 2018, pp. 10 040–10 050.

[3] O. Perrotin, B. Stephenson, S. Gerber, and G. Bailly, "The Blizzard Challenge 2023," in *Proc. 18th Blizzard Challenge Workshop*, 2023, pp. 1–27.

[4] T. B. Brown *et al.*, "Language models are few-shot learners," in *Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[5] OpenAI, "GPT-4 technical report," *CoRR*, vol. abs/2303.08774, 2023.

[6] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695.

[7] C. Saharia *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," in *Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[8] Z. Borsos *et al.*, "Audiolm: A language modeling approach to audio generation," *IEEE ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2523–2533, 2023.

[9] A. Vyas *et al.*, "Audiobox: Unified audio generation with natural language prompts," *CoRR*, vol. abs/2312.15821, 2023.

[10] C. Wang *et al.*, "Neural codec language models are zero-shot text to speech synthesizers," *CoRR*, vol. abs/2301.02111, 2023.

[11] M. Mosbach, T. Pimentel, S. Ravfogel, D. Klakow, and Y. Elazar, "Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation," in *Findings of the ACL: ACL 2023, Toronto, Canada, July 9-14, 2023*. ACL, 2023, pp. 12 284–12 314.

[12] N. Houlsby *et al.*, "Parameter-efficient transfer learning for NLP," in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, vol. 97. PMLR, 2019, pp. 2790–2799.

[13] X. L. Li and P. Liang, "Prefix-tuning: Optimizing continuous prompts for generation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*. ACL, 2021, pp. 4582–4597.

[14] E. B. Zaken, Y. Goldberg, and S. Ravfogel, "Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*. ACL, 2022, pp. 1–9.

[15] E. J. Hu *et al.*, "Lora: Low-rank adaptation of large language models," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[16] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," ser. Psychology of Learning and Motivation. Academic Press, 1989, vol. 24, pp. 109–165.

[17] R. M. French, "Catastrophic forgetting in connectionist networks," *Trends in Cognitive Sciences*, vol. 3, no. 4, pp. 128–135, 1999.

[18] I. J. Goodfellow, M. Mirza, X. Da, A. C. Courville, and Y. Bengio, "An empirical investigation of catastrophic forgeting in gradient-based neural networks," in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014*, 2014.

[19] Y. Luo, Z. Yang, F. Meng, Y. Li, J. Zhou, and Y. Zhang, "An empirical study of catastrophic forgetting in large language models during continual fine-tuning," *CoRR*, vol. abs/2308.08747, 2023.

[20] D. J. C. MacKay, "A Practical Bayesian Framework for Backpropagation Networks," *Neural Computation*, vol. 4, no. 3, pp. 448–472, 05 1992.

[21] J. Kirkpatrick *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[22] J. Martens and R. B. Grosse, "Optimizing neural networks with kronecker-factored approximate curvature," in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, vol. 37. JMLR.org, 2015, pp. 2408–2417.

[23] A. Botev, H. Ritter, and D. Barber, "Practical Gauss-Newton optimisation for deep learning," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70. PMLR, 06–11 Aug 2017, pp. 557–565.

[24] H. Ritter, A. Botev, and D. Barber, "A scalable laplace approximation for neural networks," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[25] X. Li, Y. Grandvalet, and F. Davoine, "Explicit inductive bias for transfer learning with convolutional networks," in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, vol. 80. PMLR, 2018, pp. 2830–2839.

[26] H. Ritter, A. Botev, and D. Barber, "Online structured laplace approximations for overcoming catastrophic forgetting," in *Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, 2018, pp. 3742–3752.

[27] R. Pascanu and Y. Bengio, "Revisiting natural gradient for deep networks," in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014*, 2014.

[28] T. George, C. Laurent, X. Bouthillier, N. Ballas, and P. Vincent, "Fast approximate natural gradient descent in a kronecker factored eigenbasis," in *Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, 2018, pp. 9573–9583.

[29] A. Kristiadi, M. Hein, and P. Hennig, "Being bayesian, even just a bit, fixes overconfidence in relu networks," in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, vol. 119. PMLR, 2020, pp. 5436–5446.

[30] A. Immer, M. Korzepa, and M. Bauer, "Improving predictions of bayesian neural nets via local linearization," in *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*, vol. 130. PMLR, 2021, pp. 703–711.

[31] E. Daxberger, A. Kristiadi, A. Immer, R. Eschenhagen, M. Bauer, and P. Hennig, "Laplace redux - effortless bayesian deep learning," in *Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, 2021, pp. 20 089–20 103.

[32] T. Kao, K. T. Jensen, G. van de Ven, A. Bernacchia, and G. Hennequin, "Natural continual learning: success is a journey, not (just) a destination," in *Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, 2021, pp. 28 067–28 079.

[33] J. Pfeiffer, I. Vulic, I. Gurevych, and S. Ruder, "MAD-X: an adapter-based framework for multi-task cross-lingual transfer," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. ACL, 2020, pp. 7654–7673.

[34] J. He, C. Zhou, X. Ma, T. Berg-Kirkpatrick, and G. Neubig, "Towards a unified view of parameter-efficient transfer learning," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[35] B. Lester, R. Al-Rfou, and N. Constant, "The power of scale for parameter-efficient prompt tuning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. ACL, 2021, pp. 3045–3059.

[36] N. Hyeon-Woo, M. Ye-Bin, and T. Oh, "Fedpara: Low-rank hadamard product for communication-efficient federated learning," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[37] A. Edalati, M. S. Tahaei, I. Kobyzev, V. P. Nia, J. J. Clark, and M. Rezagholizadeh, "Krona: Parameter efficient tuning with kronecker adapter," in *ENLSP-III NeurIPS Workshop*, 2023.

[38] S. Yeh, Y. Hsieh, Z. Gao, B. B. W. Yang, G. Oh, and Y. Gong, "Navigating text-to-image customization: From lyCORIS fine-tuning to model evaluation," in *The Twelfth International Conference on Learning Representations*, 2024.

[39] Y. Mao *et al.*, "Unipelt: A unified framework for parameter-efficient language model tuning," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*. ACL, 2022, pp. 6253–6264.

[40] G. M. van de Ven, T. Tuytelaars, and A. S. Tolias, "Three types of incremental learning," *Nature Machine Intelligence*, vol. 4, no. 12, pp. 1185–1197, 2022.

[41] Z. Li and D. Hoiem, "Learning without forgetting," in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016*, vol. 9908. Springer, 2016, pp. 614–629.

[42] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, vol. 70. PMLR, 2017, pp. 3987–3995.

[43] J. S. Smith *et al.*, "Continual diffusion: Continual customization of text-to-image diffusion with c-loRA," *Transactions on Machine Learning Research*, 2024.

[44] X. Wang *et al.*, "Orthogonal subspace learning for language model continual learning," in *Findings of the ACL: EMNLP 2023, Singapore, December 6-10, 2023*. ACL, 2023, pp. 10 658–10 671.

[45] J. Xiang *et al.*, "Language models meet world models: Embodied experiences enhance language models," in *Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[46] Q. Zhang *et al.*, "Adaptive budget allocation for parameter-efficient fine-tuning," in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[47] H. Liu *et al.*, "Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning," in *Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[48] W. Liu *et al.*, "Parameter-efficient orthogonal finetuning via butterfly factorization," in *The Twelfth International Conference on Learning Representations*, 2024.

[49] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "GLUE: A multi-task benchmark and analysis platform for natural language understanding," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[50] A. Williams, N. Nangia, and S. R. Bowman, "A broad-coverage challenge corpus for sentence understanding through inference," in *Proceedings of the 2018 Conference of the North American Chapter of the ACL: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*. ACL, 2018, pp. 1112–1122.

[51] S. Iyer, N. Dandekar, and K. Csernai, "Quora question pairs dataset," 2019. [Online]. Available: https://quoradata.quora.com/First-Quora-Dataset-Release-Question-Pairs

[52] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100, 000+ questions for machine comprehension of text," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*. ACL, 2016, pp. 2383–2392.

[53] R. Socher *et al.*, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Seattle, Washington, USA*. ACL, 2013, pp. 1631–1642.

[54] S. Merity, C. Xiong, J. Bradbury, and R. Socher, "Pointer sentinel mixture models," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017*. OpenReview.net, 2017.

[55] S. Zhang *et al.*, "OPT: open pre-trained transformer language models," *CoRR*, vol. abs/2205.01068, 2022.

[56] L. Gao *et al.*, "The pile: An 800gb dataset of diverse text for language modeling," *CoRR*, vol. abs/2101.00027, 2021.

[57] Y. A. Li, C. Han, V. S. Raghavan, G. Mischler, and N. Mesgarani, "Styletts 2: Towards human-level text-to-speech through style diffusion and adversarial training with large speech language models," in *Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[58] M. Lajszczak *et al.*, "BASE TTS: lessons from building a billion-parameter text-to-speech model on 100k hours of data," *CoRR*, vol. abs/2402.08093, 2024.

[59] T. Wolf *et al.*, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: ACL, Oct. 2020, pp. 38–45.

[60] S. Mangrulkar, S. Gugger, L. Debut, Y. Belkada, S. Paul, and B. Bossan, "Peft: State-of-the-art parameter-efficient fine-tuning methods," https://github.com/huggingface/peft, 2022.

[61] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015*, 2015.

[62] H. Zen *et al.*, "Libritts: A corpus derived from librispeech for text-to-speech," in *20th Annual Conference of the International Speech Communication Association, Interspeech 2019, Graz, Austria, September 15-19, 2019*. ISCA, 2019, pp. 1526–1530.

[63] J. Yamagishi, C. Veaux, and K. MacDonald, "CSTR VCTK Corpus: English multi-speaker corpus for CSTR voice cloning toolkit (version 0.92)," 2019. [Online]. Available: https://datashare.ed.ac.uk/handle/10283/3443

[64] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: emphasized channel attention, propagation and aggregation in TDNN based speaker verification," in *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*. ISCA, 2020, pp. 3830–3834.

[65] B. Naderi and R. Cutler, "An open source implementation of ITU-T recommendation P.808 with validation," in *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*. ISCA, 2020, pp. 2862–2866.