TOWARDS CONTINUALLY LEARNING NEW LANGUAGES

Ngoc-Quan Pham[1] *Jan Niehues*[1] *Alex Waibel*[1,2]

[1]Interactive Systems Lab, Karlsruhe Institute of Technology, Karlsruhe, Germany
[2]Carnegie Mellon University, Pittsburgh PA, USA

ABSTRACT

Multilingual speech recognition with neural networks is often implemented with batch-learning, when all of the languages are available before training. An ability to add new languages after the prior training sessions can be economically beneficial, but the main challenge is catastrophic forgetting. In this work, we combine the qualities of weight factorization and elastic weight consolidation in order to counter catastrophic forgetting and facilitate learning new languages quickly. Such combination allowed us to eliminate catastrophic forgetting while still achieving performance for the new languages comparable with having all languages at once, in experiments of learning from an initial 10 languages to achieve 26 languages without catastrophic forgetting and a reasonable performance compared to training all languages from scratch.

Index Terms: speech recognition, multilingual, transformer, continual learning, incremental learning

1. INTRODUCTION

End-to-end speech recognition with neural networks, unlike classical Hidden Markov Model based models, can be easily extended for the multilingual scenario in which the training data consists of multiple languages [1, 2, 3]. The common assumption in these works so far, however, is the datasets for all languages being available before the training process. Supervised learning, therefore, can be applied in straightforward manner.

In practice, it is also possible that only a subset of the languages is available at first, and the data for new languages can be added after the training process. Reversely, the data for previously trained languages might be discarded for storage or privacy reasons. In such case, the typical batch-training scenario often resorts to two options, either to fine-tune the models on the new datasets to obtain new models that are capable of transcribing new languages, or to combine the old and new datasets to construct new models that fit all languages. Naturally, both of these options are not optimal. Fine-tuning on new languages poses a threat for the previously learned languages to be forgotten, known as *catastrophic forgetting* [4] that happened when the parameters of the neural networks are shifted towards optimizing the loss function for the new dataset, and far away from the optimal points with respect to the old ones. On the other hand, training all languages together can potentially obtain the best performance for all languages, but is costly since the training time of neural networks can scale depending on the amount of training data. Furthermore it is not possible when the previous languages are no longer available.

The objective of this paper, therefore, is to find new training strategy for multilingual speech recognition in such continual learning scenario. The desiderata of continual learning involves a number of factors:

- Forward transfer: adding new languages to the current multilingual model can ideally obtain the performance similar to when having them in the initial training.
- Backward transfer: catastrophic forgetting is avoided for the previously learned languages, ideally adding the new languages should not affect the performance for the previously learned ones.
- Optimal training cost: the process of learning new languages should be economically better than re-training all languages from the beginning, in terms of training speed and storage.

Such desiderata can be satisfied by using the combination of two techniques: weight factorization and elastic weight consolidation. The key idea is to organize the weights in the networks into a shared component while off-loading some information into the language-specific components and allocate new weights for new languages in a progressive manner [5]. Weight factorization [3] is one of such methods, facilitating multilingual models by decomposing each matrix in the neural network into a shared component and per-language additive and multiplicative factors. The shared part in weight factorization can be trained elastically using elastic weight consolidation (EWC) that relies on per-weight importance to search for "empty space" in the network capacity.

Applying the techniques in a continual learning scenario involving 26 languages, we showed that it is possible to obtain a network that learn these languages progressively with the ideal performance without catastrophic forgetting. The combination is important such that, EWC by itself is quite ineffective and weight factorization can only prevent forgetting with a heavy loss in learning new languages. Our contribution in continual learning, to the best of our knowledge, the first application in learning to transcribe new languages.

2. RELATED WORKS

Learning tasks consecutively without catastrophic forgetting and using the knowledge of previous tasks to facilitate learning new task is an important topic in machine learning that has been investigated in computer vision or reinforcement learning. There are three common approaches in continual learning: regularization, progressive architecture and replaying from memory. The regularization approach is model agnostic and focuses on designing objective functions that punish weights that tend to be shifted too far from the original positions, where the optimal state with respect to the previous tasks is achieved. The important weights can be identified by importance [6] or memory synapses [7]. Besides, the network can also be designed to to isolate the weights and module of each task, while allocating new weights for new tasks [5, 8]. It is also possible to store examples of previous tasks as memory replaying [9] to ensure that the gradient updates in the new tasks do not have negative effect over the previous datasets.

In Automatic Speech Recognition, continual learning or incremental learning has been explored in a number of monolingual scenarios. The hybrid HMM models were explored in continual learning by learning different datasets such as World Street Journal, Reverb, Librispeech and Chime4 consecutively [10]. In a similar manner, the sequence-to-sequence model can also be trained on different English datasets with the goal of evaluating the performance in each domain after training on another [11]. Recently, the replaying from memory approach has been applied to online continual learning [**?**] without a clear boundary within task.

Compared to the related works, continual learning new languages in multilingual ASR has a clear task separation due to the difference between languages, compared to monolingual setups. The weight factorization method can be classified into the architectural approach, by assigning new network parameter for new 5 languages. In our work, we combine both architectural and regularization approaches to cover forward and backward transfers in the desiderata.

3. CONTINUALLY LEARNING APPROACH

An end-to-end neural model, such as a Transformer model, learns to map the input acoustic features X to a sequence of symbols Y.

$$\begin{aligned} H^E &= Encoder(X, \theta_E) \\ H_t^Y &= Decoder(Y_{t-1}, H_E, \theta_D) \\ P_t &= Softmax(W_{emb} H_t^Y) \end{aligned}$$

in which H_E is the encoded representation from X which is then used by the decoder to auto-regressively generate the hidden states H_t^Y from the previous input Y_{t-1}. The probabilistic output layer P_t is generated by the product between H_t^Y and the word embeddings $W_e mb^1$. Avoiding catastrophic forgetting when adding new languages boils down to how these parameters are used, because they are directly changed when the model is exposed to new languages.

3.1. Weight factorization

A large part of the model parameters θ_E and θ_D are matrices X that linearly project input features X, such as the query-key-value matrices in attention or the weights of the feed-forward neural networks in Transformers. For multilingual representation, these weights can be factorized into the shared component W_S and the language specific parts W_M (multiplicative term) and W_B (bias term):

$$Y = (W_S \odot W_M + W_B)X \tag{1}$$

In order to reduce the number of parameters as well as to encourage the model to share more information between languages instead of partitioning into the exclusive terms, each language-dependent matrix W_M or W_B is further factorized into outer-products of vectors $r \in \mathbb{R}^{D_in}$ and $v \in \mathbb{R}^{D_out}$.

$$W_M = r_m \odot v_m; W_B = r_b \odot v_b \tag{2}$$

We can increase the capacity of each factor by using k different r_m, v_m, r_b, v_b and summing up the outer-products of each pair. With the value of $k << D_{in}$ or D_{out}, the cost specializing each language is $\frac{2k}{D_out}$ number of parameters, assuming $D_{in} = D_{out}{}^2$. Using this method, *new weights* (W_M and W_B specifically can be added to the model for new languages. Freezing the shared weights W_S is the obvious way to prevent catastrophic forgetting, but due to the difference in size between them and the factorized weights, such approach can compromise the performance for the new languages.

3.2. Elastic weight consolidation

A different approach is to allow the shared weights to be elastically updated. EWC [6] is the *regularization* method [12] that punishes the weights from being far from the previously trained state, to avoid deterioration. Assuming after the first training iteration with the initial dataset T_0, we obtain the parameters θ_0 optimized for the training objective in D^0, the next training iteration with the dataset T_1 is regularized with additional loss term:

$$\frac{1}{2} \sum_{j=1}^{d} f_j(\theta_j - \theta_j^0) \tag{3}$$

[1]In the case of end-to-end models using CTC loss, this modeling scheme still applies without the involvement of Y_{t-1}

[2]Its actually much lower than that, because the network may contain layers that do not need to be factorized, such as the output layer, or layer normalization

In which θ denotes the current parameters (initialized with θ_0) and f_j is the importance of the parameter θ_j^0. Minimizing this loss term prevents θ^1 optimized for dataset T_1 to not deviate too far away from the T_0-optimized params θ^0. The importance f is estimated with the diagonal of the Fisher Information matrix, containing the variance of gradients in T_0[3].

Beyond two datasets, EWC can be expanded to the case of m iterations. After iteration $m-1$, the Fisher diagonals for D_{m-1} (estimated with θ^{m-1}) is accumulated to the sum of all previously computed Fishers of $T_{0 \ldots m-2}$ to be used as regularization weights to train the next T_m.

4. EXPERIMENTS AND RESULTS

With such method in mind, we designed the experiments to observe how the models can learn new languages. The research questions are:

- Freezing the shared weights in the weight factorization (WF) scheme completely prevents catastrophic forgetting. Can we obtain the same effect using elastic weight consolidation combined with WF, as minimizing the performance loss of previously learned languages?
- On the other polar, how does the combination of WF and EWC perform compared to the ideal case in which all languages are present at the same time?
- Given a large model size, can EWC solely allow for effective continual learning?

We used the Transformer encoder-decoder model [13] for speech recognition. The decoder weights (including W_emb) are initialized with the MBART50 pretrained model, with the byte-pair encoder covering all 26 languages involved. This way, the presence of potential new output words/characters is not a concern. For better performance and higher model capacity [14], the encoder is initialized with the *xlsr-53* [15] pretrained wav2vec model [16]. All of the weight matrices in the model are factorized with $k = 8$, except for W_emb. The model is trained on 26 languages collected from Commonvoice [17] and Europarl-ST [18] datasets[4] , with the amount of data ranging from 7 to 1050 hours for each language. We used the Large-configuration for both encoder and decoder with the hidden size of 1024 and Dropout 0.3. For EWC training, it is necessary to tune the coefficient of the EWC loss, empirically from 0.00001 to 0.1. Our training strategy is to start from a high value (so that model learns with almost frozen parameters [19] and the factorized weights first), and then relaxing the value over the training course.[5]

[3]This is computed using one single forward pass over the whole dataset T_0 and taking the variance of gradients for all samples

[4]Europarl-ST only covers 8 languages in the 26 language pool

[5]Starting at 0.1 the value is reduced by 10 per $20K$ training steps, the continual learning iteration takes about $60K$ steps, while training the model takes $200K$ steps

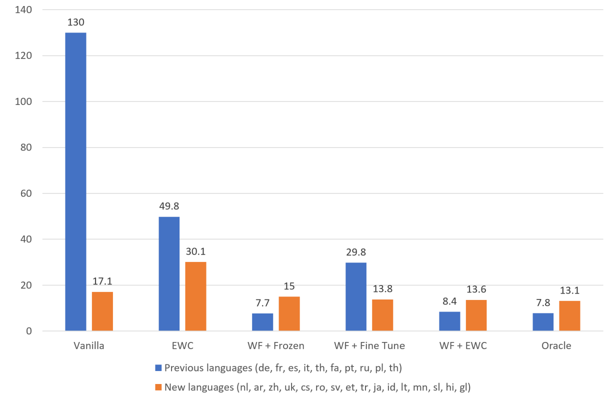

Fig. 1. Comparison between different approaches: Weight factorization (WF) with frozen/fine-tuned/elastic shared weights, elastic weight consolidation (EWC) and a simple fine-tuning (Vanilla). Reported is the average of word error rates (WER) for the languages in the set.

4.1. Can we learn new languages without forgetting?

In a simple scenario, the model is trained in 2 iterations, with the initial one involving 10 languages with the highest amount of data and then the continual learning iteration having 16 languages. Figure 1 that shows the average error rates of the languages shows different learning behaviour of different approaches. A "vanilla" model when fine-tuned on the new languages are quickly shifted so that it cannot retain any previously learned knowledge anymore, even when the encoder and decoder are initialized with pre-trained models covering the new languages. Likewise, the regularization of EWC led the model to a bad state. The deterioration is less severe than complete forgetting, however the performance of the new language is handicapped at 30.1%. The models with weight factorization (WF), however, showed more promising behaviours. Freezing the shared weights keeps the previous learned languages intact, while fine-tuning them increases the error from 7.7% to 29.8%. Surprisingly, combining with EWC maintains the same performance for the new languages compared to fine-tuning at 13.6% and the deterioration for the old ones is limited at 8.4%. The slight improvement over the fine-tuned WF model could be reasoned by the regularization effect of the weights that prevents overfitting for low-resourced languages.

4.2. Continually learning in Multiple iterations

The second scenario involves several iterations, in each of which the model is exposed with a new group of languages. The starting point is the same 10 languages of the previous scenario, we divided the rest of the 16 languages into three groups based on the amount of data.

Table 1 shows the rate of **degradation** over the course of learning with the combination of EWC and WF, compared

Table 1. Combination of EWC and Factorization (WF) vs. WF with frozen shared parameters for three iterations in word error rates (WER). Iteration 0 is the initial learning stage, with 10 languages. The performance of WF with EWC is shown in each iteration, while the performance of WF is always the same across iterations. The languages from the third block (ro, sv, et, tr, ja) are added in Iteration 2 and then treated as "old" languages in the third one.

Lg	Hours	**EWC + WF**			WF
		Iter 1	**Iter 2**	**Iter 3**	-
(de)	1050	7.4	8.7	10.5	7.23
(fr)	800	11.5	13.1	15	11.4
(es)	400	6.7	8.4	9.8	6.74
(it)	325	7.1	9.1	11.1	6.8
(fa)	293	4.1	4.9	6.5	3.7
(ta)	198	19.7	23.3	28.5	18.2
(pt)	120	7.3	9.2	11	7
(ru)	148	6	7.4	9.6	5.3
(pl)	145	8.1	9.4	11.3	7.72
(th)	133	3.4	4	4.6	3.2
Avg	-	**8.1**	**9.8**	**11.7**	**7.7**
(nl)	150	7.19	7.7	9	8
(ar)	85	15.9	16	17.4	19.4
(zh)	63	14.8	15.7	16.9	17.7
(uk)	56	7.9	9.1	12.6	10.4
(cs)	49	9.3	0.3	13.9	10.6
Avg	-	**11**	**11.8**	**14**	**13.2**
(ro)	45	-	11.6	12.8	12
(sv)	35	-	12.1	14.7	14.8
(et)	32	-	12.1	14.7	16.8
(tr)	30	-	7.5	9.5	9.4
(ja)	26	-	7.5	8.3	9.5
Avg	-	-	**10.2**	**12**	**12.5**
(id)	23	-	-	7.9	8
(lt)	16	-	-	28.5	29.3
(mn)	12	-	-	27.7	28
(sl)	9	-	-	11	12.3
(hi)	8	-	-	29.7	30.8
(gl)	7	-	-	12.3	10.9
Avg	-	-	-	**19.5**	**19.9**

to the simple parameter-freezing approach. It is observable that the degradation rate of EWC seems to be faster after the first iteration. For example, in the initial language group (first block), the reduction rate is 5.2% in the first iteration, then 22.2% in the second iteration, and then 19.3% in the third. Similarly, the first new language group (second block) only witnessed a 7.2% reduction rate (from $11\% to 11.8\%$), then 18.6%. In an attempt to explain this problem, we calculated the number of important weights (ranked by the Fisher diagonal values f and the parameters with $f_i \geq 0.25$ can be considered important. After the initial iteration, the network has around 75% of weights being important and 25% weights that can be allocated to the new languages. The first iteration quickly raised this number to 99% and thus the network needed to compromise further in the next step. In exchange, the elastic nature of EWC allowed for the network to learn new languages better than before. Albeit this advantage is somewhat hindered in the third iteration, when the performance between EWC + WF and frozen WF is similar. Probably the reason also lies on the capacity problem above.

The explanation for the ineffectiveness of EWC probably comes from the derivation into the final equation of the regularization loss term. From the theoretical analysis [20], EWC originates from replacing the log posterior $\log p(\theta|T_1)$ with its Taylor expansion form, that requires the optimal value θ^* during optimizing the model for the data T_1. The stochastic gradient descent (SGD) algorithm is not guaranteed to achieve the exact optimal value, for example a typical practice in training Transformer is to average the parameters of several checkpoints[6] showed this trouble of SGD. The approximation is further "approximated" by the fact that the Hessian in the Taylor expansion is approximated by the diagonal of the Fisher Information matrix. Furthermore, the prior is also assumed to be a zero-mean isometric Gaussian [20] which is rather a simple assumption [21]. From such approximation, it is understandable that EWC might be only effective when the new task/data is somewhat close to the original task which is unlikely in language learning.

5. CONCLUSION

Learning new languages without forgetting falls into the intersection between automatic speech recognition and continual learning/incremental learning areas. In this paper, we presented a model for this specific task with the combination of Elastic Weight Consolidation and Weight factorization. This model exhibits an interesting property when learning languages sequentially, that it can learn the new languages very effectively compared to other regularized baselines, while maintaining the performance for the previously learned languages to an acceptable extent. In this work, we focused on tackling the problems in the networks themselves that cause catastrophic forgetting. Further problems in continual learning such as learning to add new symbols dynamically for the new languages will be the target for future works. Alternatively, one can rely on distillation from previous models as a compression format of the data when the latter is no longer available, shown in many concurrent work in image generation [22].

[6]which we applied here for the last 10 checkpoints with the highest unigram development accuracy

6. REFERENCES

[1] Yun Zhu, Parisa Haghani, Anshuman Tripathi, Bhuvana Ramabhadran, Brian Farris, Hainan Xu, Han Lu, Hasim Sak, Isabel Leal, Neeraj Gaur, et al., "Multilingual speech recognition with self-attention structured parameterization.," in *INTERSPEECH*, 2020, pp. 4741–4745.

[2] Vineel Pratap, Anuroop Sriram, Paden Tomasello, Awni Hannun, Vitaliy Liptchinsky, Gabriel Synnaeve, and Ronan Collobert, "Massively multilingual asr: 50 languages, 1 model, 1 billion parameters," *arXiv preprint arXiv:2007.03001*, 2020.

[3] Ngoc-Quan Pham, Tuan-Nam Nguyen, Sebastian Stüker, and Alex Waibel, "Efficient Weight Factorization for Multilingual Speech Recognition," in *Proc. Interspeech 2021*, 2021, pp. 2421–2425.

[4] Robert M French, "Catastrophic forgetting in connectionist networks," *Trends in cognitive sciences*, vol. 3, no. 4, pp. 128–135, 1999.

[5] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell, "Progressive neural networks," 2016.

[6] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al., "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, 2017.

[7] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars, "Memory aware synapses: Learning what (not) to forget," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 139–154.

[8] Xin Guo, Yu Tian, Qinghan Xue, Panos Lampropoulos, Steven Eliuk, Kenneth Barner, and Xiaolong Wang, "Continual learning long short term memory," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020, pp. 1817–1822.

[9] David Lopez-Paz and Marc'Aurelio Ranzato, "Gradient episodic memory for continual learning," *Advances in neural information processing systems*, vol. 30, 2017.

[10] Samik Sadhu and Hynek Hermansky, "Continual learning in automatic speech recognition.," in *Interspeech*, 2020, pp. 1246–1250.

[11] Heng-Jui Chang, Hung yi Lee, and Lin shan Lee, "Towards Lifelong Learning of End-to-End ASR," in *Proc. Interspeech 2021*, 2021, pp. 2551–2555.

[12] German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54–71, 2019.

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017.

[14] Ngoc-Quan Pham, Alex Waibel, and Jan Niehues, "Adaptive multilingual speech recognition with pretrained models," in *Proc. Interspeech 2022*, 2022.

[15] Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli, "Unsupervised Cross-Lingual Representation Learning for Speech Recognition," in *Proc. Interspeech 2021*, 2021, pp. 2426–2430.

[16] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12449–12460, 2020.

[17] Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M Tyers, and Gregor Weber, "Common voice: A massively-multilingual speech corpus," *arXiv preprint arXiv:1912.06670*, 2019.

[18] Javier Iranzo-Sánchez, Joan Albert Silvestre-Cerdà, Javier Jorge, Nahuel Roselló, Adrià Giménez, Albert Sanchis, Jorge Civera, and Alfons Juan, "Europarl-st: A multilingual corpus for speech translation of parliamentary debates," in *ICASSP*, 2020.

[19] Alexey Kutalev and Alisa Lapina, "Stabilizing elastic weight consolidation method in practical ml tasks and using weight importances for neural network pruning," *arXiv preprint arXiv:2109.10021*, 2021.

[20] Ferenc Huszár, "On quadratic penalties in elastic weight consolidation," *arXiv preprint arXiv:1712.03847*, 2017.

[21] Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling, "Improved variational inference with inverse autoregressive flow," *Advances in neural information processing systems*, vol. 29, 2016.

[22] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui, "Open-vocabulary object detection via vision and language knowledge distillation," *arXiv preprint arXiv:2104.13921*, 2021.