

EXPLORING HETEROGENEOUS CHARACTERISTICS OF LAYERS IN ASR MODELS FOR MORE EFFICIENT TRAINING

Lillian Zhou* Dhruv Guliani* Andreas Kabel Giovanni Motta Françoise Beaufays

Google LLC, Mountain View, CA, U.S.A.
{lqz, dguliani, aka}@google.com

ABSTRACT

Transformer-based architectures have been the subject of research aimed at understanding their overparameterization and the non-uniform *importance* of their layers. Applying these approaches to Automatic Speech Recognition, we demonstrate that the state-of-the-art Conformer models generally have multiple *ambient* layers. We study the stability of these layers across runs and model sizes, propose that group normalization may be used without disrupting their formation, and examine their correlation with model weight updates in each layer. Finally, we apply these findings to Federated Learning in order to improve the training procedure, by targeting Federated Dropout to layers by importance. This allows us to reduce the model size optimized by clients without quality degradation, and shows potential for future exploration.

Index Terms— speech recognition, federated learning, ambient layers, deep learning

1. INTRODUCTION

Neural models for Automatic Speech Recognition (ASR) [1] have improved to surpass conventional systems [2], and newer architectures, such as the Conformer [3], have improved quality and latency for mobile applications. These models typically contain well over 100M parameters [4, 5, 6] and often require hundreds of thousands of training steps to converge. This is resource intensive [7] and poses significant practical challenges when training ASR models on edge devices [8].

Research in deep learning has shown that neural networks benefit from being overparameterized [9, 10], and their layers exhibit non-uniform importance to the predictor’s performance [11], with *ambient* layers having little impact, and *critical* layers causing catastrophic changes to model quality if tampered with. Other work has found that, with transformer- and convolution-based architectures, high-quality models can be learned by training only a subset of the full model’s weights [12, 13, 14, 15, 16]. We are motivated by these discoveries to study and extend the concept

of ambient layers to ASR, with the hypothesis that training algorithms can be improved using the ambient characteristics of layers to inform model updates. We apply these findings to an application in Federated Learning (FL) [17].

In doing so, we make the following key contributions:

- Prior work [11] explores ambient properties in convolutional models for image classification. We extend this to State-of-the-Art (SOTA) ASR models [3], demonstrating that ambient properties exist across model sizes, architectures, and domains.
- [11] omits batch normalization at cost of some degradation in model quality in order to better illustrate ambient properties. We show that group normalization [18] can be used while maintaining ambient properties.
- We explore model weight updates of trained models, and how they might relate to ambient properties.
- We show one application to a practical FL domain adaptation task, by targeting Federated Dropout [19] using ambient properties.

2. METHODOLOGY

2.1. Data

We use the Librispeech [20] corpus, containing ~960 hours of transcribed training utterances, and ~21 hours of evaluation audio split over 4 sets, *Dev*, *DevOther*, *Test*, and *TestOther*, with the reporting metric of *Word Error Rate* (WER). In this paper, we primarily report on the *TestOther* set, though we examined the WER for all evaluation sets in our experiments and found performance to be consistent across them.

We then apply our findings to a multi-domain dataset (MD) encompassing domains of search, farfield, telephony, YouTube, etc [21, 22]. Relevant durations are shown in Table 1. A portion of this data, termed Short-Form (SF), with an average duration of 4.6 seconds per utterance, is the subject of our federated learning domain transfer experiments. First, we perform centralized training on the MD dataset with SF withheld (MD-SF), and then follow up with training on this withheld domain (SF) in FL simulation. The resulting model is evaluated on a disjoint test set from the SF domain.

*These authors contributed equally to this work.

Dataset	Hours
Multi-domain (MD)	400k
Short-form domain (SF)	27k
Short-form held out (MD-SF)	373k

Table 1. Durations of MD datasets.

Model	Conf Params	Conf Layers	Total Params
ConformerS	8.1M	$16 \times 0.5\text{M}$	10.3M
ConformerM	25.4M	$16 \times 1.6\text{M}$	30.7M
ConformerL	107.5M	$17 \times 6.3\text{M}$	118.6M

Table 2. Architecture of the Non-streaming Conformers.

2.2. Models

Initial experiments analyzing ambient properties are conducted with the *Non-Streaming* Conformer [3] architecture, which is a transformer augmented with convolution layers. We use the three different sizes of models (ConformerS, ConformerM, and ConformerL) from that paper.

We look for ambient properties within the encoder, which, for these models, comprises 16 or 17 equally-sized Conformer layers (Table 2). These models are trained using the Librispeech corpus.

We expand our findings to larger-scale Multi-domain data using a similar *Streaming* Conformer. This Conformer has modifications to enable latency improvements for a streaming context [4]. Specifically, the order of the convolution and multi-headed self-attention modules is reversed (Figure 1), and a frame-stacking layer is introduced as its encoder layer 4. The additional frame-stacking increases the size of this layer relative to the rest of the encoder layers.

Of note is that in both architectures, there are residual connections around individual modules in the encoder layers, which may help to maintain predictive power when ambient layers are tampered with.

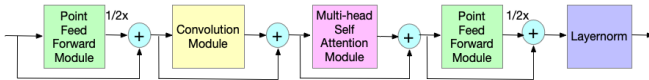


Fig. 1. Streaming Conformer encoder layer; from [4].

2.3. Ambient Properties

We refer to heterogeneous characteristics [11] – layers being ambient or critical [16] – as the *ambient properties* of a network. As in prior work, we identify these ambient properties through ablation studies on each model’s encoder layers. As detailed above, the Conformer [3] architectures we use contain either 16 or 17 repeated conformer layers in their encoders. Although these layers consist of multiple modules, as

illustrated in Figure 1, we examine ambient characteristics at the granularity of the entire layer.

To do so, we study post-training properties of the converged model. Let θ denote a model with D parametric layers $\{\theta_0, \dots, \theta_{D-1}\}$. Similarly, let θ^T represent the state of the model after T steps of training. We wish to determine empirically which of the D layers are most ambient, and which are most critical.

Specifically, we run two sets of experiments to reset each layer, and check the resulting model’s accuracy (also referred to as *quality*). For *re-initialization* (re-init) experiments, we sequentially set one conformer layer at a time to its initial values $\theta_d^T \leftarrow \theta_d^0$. For *re-randomization* (re-rand) studies, we set each layer to random values sampled from the same distribution $\theta_d^0 \sim P_d^0$ that the original initial values were sampled from. In both cases, an evaluation on the held-out dataset is then performed to measure WER. Layers that can be reset without significant penalty are considered ambient layers, while the remainder are critical layers.

We conduct re-initialization and re-randomization studies for different initial random seeds using SOTA architectures [3], reporting both ambient properties and stability across runs.

In [11], batch normalization was omitted, as it disrupts the formation of ambient properties. We run experiments using batch normalization to confirm this finding, and also try replacing it with group normalization [18] as an alternative.

2.4. Norms and Churn Metrics

While empirically resetting layers is one way to discover ambient properties, it would be useful to find numeric properties of the trained model that would reflect which layers are ambient. Similar to [11], we check the norm of the weight matrices of our trained checkpoints across layers, but do so per module. With step t , layer l , and module m , we plot

$$\frac{|\mathbf{W}^{m,l}(t) - \mathbf{W}^{m,l}(0)|_F}{\max_{l'} |\mathbf{W}^{m,l'}(t) - \mathbf{W}^{m,l'}(0)|_F}$$

for each m over l for a fixed t , where the Frobenius norm $|\cdot|_F$ is taken over all indices other than layer and module (e. g.,, attention heads).

We refer to this as the *churn* metric, and compare it across Conformer layers with the ambient properties found empirically.

2.5. Federated Domain Adaptation

Finally, to illustrate one application of ambient properties, we turn to our MD transfer learning setup. We first train a model to convergence on MD-SF, in a centralized setting. Then, we train on the unseen target domain, SF, using a trainer that simulates FL rounds across multiple clients.

To make best use of the limited amount of computational power available on edge devices, we explore using ambient properties to target memory-saving techniques during this federated portion of training. For this paper, we employ Federated Dropout [19], which reduces the number of parameters to train on the client, at the expense of model quality. By first empirically determining the model’s ambient properties, we apply heavier dropout to the ambient layers, in order to increase the computational savings, and check the resulting WER. We compare this with an alternative setting where increased dropout is applied to critical layers.

3. EXPERIMENTAL RESULTS

3.1. Empirical Study of Ambient Properties

As in [11], experiments here showed that batch normalization degrades robustness to re-randomization and re-initialization. However, we find that group normalization [18] gives rise to comparatively stronger ambient properties, as shown in Figure 2. This suggests that group normalization can be used as an alternative to batch normalization, retaining some of the normalization benefits while still being tolerant to layer resets. This works well for FL applications, where group normalization is preferable [23, 24], and reducing training costs is especially desirable. For the rest of the paper, we use group normalization in our experiments.

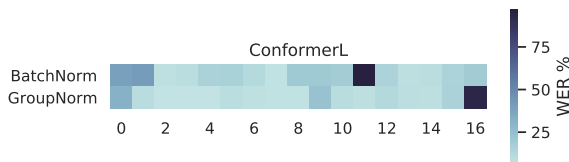


Fig. 2. Comparison of the same ConformerL model trained with batch normalization and group normalization, with each layer re-randomized. The model trained with group normalization was more robust to having its layers reset.

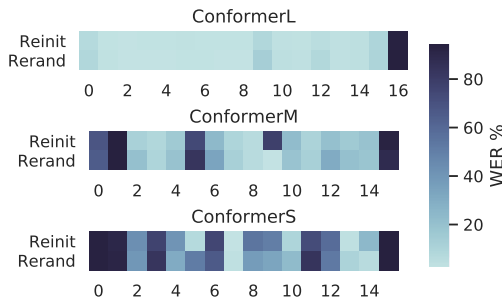


Fig. 3. Comparison of ambient properties following re-initialization and re-randomization across model sizes.

Figure 3 shows an example set of re-init and re-rand experiments. Typically, re-init experiments gave slightly better WER than re-rand. However, as the overall pattern of WER across layers was similar in both, we limit the remainder of our discussion to re-rand. Not only is this the more difficult task, but has deeper applications for FL, where random seeds can be transported instead of layer parameters for the most ambient layers to save on transport cost.

While ambient layers were present in all model sizes, we observed that larger models had more ambient layers, suggesting that ambient properties may be more evident in over-parameterized models.

We also performed re-rand experiments at various points throughout training. During early rounds, the ambient layers were more spread throughout the model; only later during training did the separation between ambient and critical layers become more distinct.

To check if ambient properties are consistent across runs, we trained each model five times, and show the means and range in Figure 4. We observed that larger models have a higher number of stable (illustrated by WER range), and more ambient (illustrated by mean WER) layers. For the largest model, several re-randomized layers show little deviation across runs, and give WER close to the baseline. This strengthens our hypothesis that the stability and extent of ambient properties may be related to overparameterization.

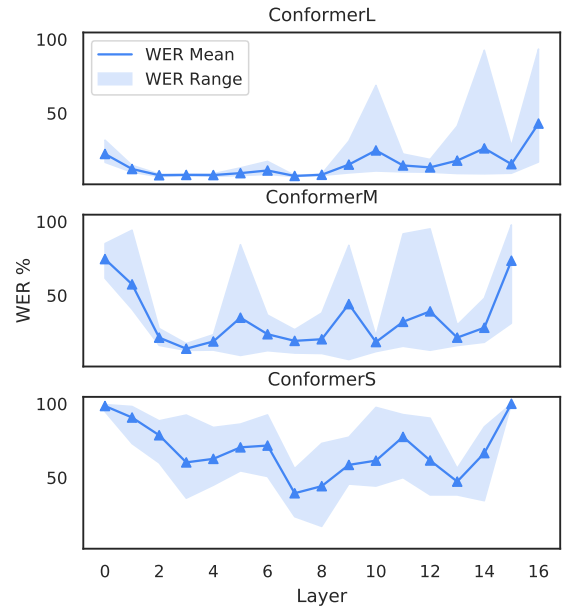
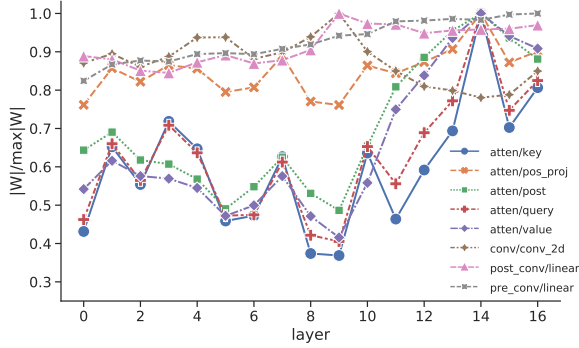


Fig. 4. Stability across 5 runs each of 3 model sizes.

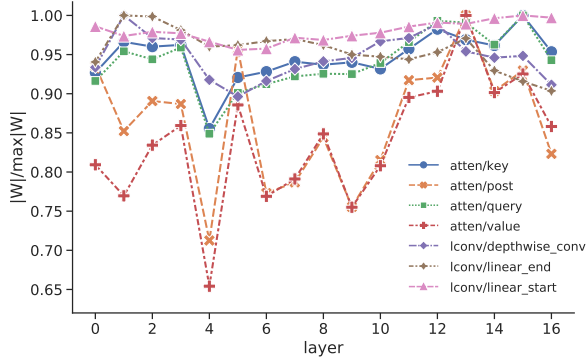
3.2. Numerical Signatures of Ambient Properties

In examining the churn metric across layers in both Non-Streaming (Figure 5(a)) and Streaming (Figure 5(b)) Con-

former models, we observed a distinct geometric pattern in attention-related modules, but not convolution-related modules. These patterns emerged during training—for earlier training phases, they were less distinct or entirely absent, in agreement with our empirical observations from ablation studies during training.



(a) Non-streaming ConformerL trained on LibriSpeech.



(b) Streaming Conformer trained on MD-SF.

Fig. 5. Relative (across layers) magnitude of the change in weight matrices for different modules.

Also similarly, the emerging pattern was less distinct for ConformerM, and absent for ConformerS. Additionally, the attention modules showed greater churn in the higher layers; likewise, we observed that higher layers were more likely to be critical. This suggests that further ablation studies on a per-module basis may yield interesting results.

4. APPLICATIONS

The following experiments were aimed at saving training costs by reducing computations on ambient layers. We explored applications in FL, where transport and edge memory are generally scarce resources [25].

Federated Dropout [19] is a technique for reducing computation and memory costs during edge training. Rather than applying flat dropout across the encoder, experiments were conducted to study whether the dropout budget could be more effectively allocated based on layer criticality.

We found that adding dropout to ambient layers gave significantly better WER results than adding dropout to critical layers. Table 3 shows the results in our domain transfer setup, where we see a 11 – 12% relative WER improvement between applying dropout to ambient versus critical layers, given the same number of parameters dropped. Furthermore, when comparing Amb-4 50% dropout to a Flat 20% dropout across the encoder, dropping from the 4 most ambient layers saves more parameters while giving the same WER. This suggests that ambient properties can find a dropout schema that improves on constant dropout across layers.

Dropout	Params Dropped	WER
Crit-2 50%	8%	6.9
Amb-2 50%	9%	6.3
Crit-3 50%	9%	7.0
Amb-3 50%	10%	6.5
Crit-4 50%	10%	7.3
Flat 20%	11%	6.6
Amb-4 50%	12%	6.6

Table 3. Quality of SF domain transfer task when dropout is either applied to the entire encoder, or targeted to the n most ambient or critical layers (denoted as *Amb- n* or *Crit- n*).

We show one application of ambient properties, but many others likely exist. It may be possible to omit ambient layers entirely during training, make optimizers aware of ambient properties to improve regularization, or use ambient properties to improve neural architecture search.

5. CONCLUSION

In this work, we explored the ambient properties of ASR Conformer models and demonstrated the existence and stability of ambient layers within them. We confirmed that batch normalization is detrimental to ambient properties and illustrated the merits of group normalization instead.

Additionally, we examined weight updates on a per-module basis, showed interesting geometric signatures of attention modules in particular, and proposed a future direction of research in per-module ablation studies.

Finally, we demonstrated how ambient properties can be leveraged to improve computational efficiency during training, and proposed a potential application to Federated Learning, where such efficiency is crucial.

6. ACKNOWLEDGEMENTS

We would like to thank Khe Chai Sim for proof-reading and providing feedback on this paper.

7. REFERENCES

- [1] A. Graves, A. Mohamed, and G. Hinton, “Speech Recognition with Deep Recurrent Neural Networks,” in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 6645–6649.
- [2] T. N. Sainath, Y. He, B. Li *et al.*, “A Streaming On-Device End-To-End Model Surpassing Server-Side Conventional Model Quality and Latency,” 05 2020, pp. 6059–6063.
- [3] A. Gulati, C.-C. Chiu, J. Qin *et al.*, Eds., *Conformer: Convolution-augmented Transformer for Speech Recognition*, 2020.
- [4] B. Li, A. Gulati, J. Yu *et al.*, “A better and faster end-to-end model for streaming asr,” 2021.
- [5] A. Gruenstein, A. Gulati, A. Narayanan *et al.*, “An efficient streaming non-recurrent on-device end-to-end model with improvements to rare-word modeling,” 2021.
- [6] B. Li, R. Pang, T. N. Sainath *et al.*, “Scaling end-to-end models for large-scale multilingual ASR,” *CoRR*, vol. abs/2104.14830, 2021.
- [7] D. Patterson, J. Gonzalez, Q. Le *et al.*, “Carbon emissions and large neural network training,” 2021.
- [8] D. Guliani, F. Beaufays, and G. Motta, “Training Speech Recognition Models with Federated Learning: A Quality/Cost Framework,” in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 3080–3084.
- [9] Z. Allen-Zhu, Y. Li, and Y. Liang, “Learning and generalization in overparameterized neural networks, going beyond two layers,” *CoRR*, vol. abs/1811.04918, 2018.
- [10] B. Neyshabur, Z. Li, S. Bhojanapalli *et al.*, “Towards understanding the role of over-parametrization in generalization of neural networks,” *CoRR*, vol. abs/1805.12076, 2018.
- [11] C. Zhang, S. Bengio, and Y. Singer, “Are all layers created equal?” *CoRR*, vol. abs/1902.01996, 2019.
- [12] A. Rosenfeld and J. K. Tsotsos, “Intriguing properties of randomly weighted networks: Generalizing while learning next to nothing,” *CoRR*, vol. abs/1802.00844, 2018.
- [13] H. Shrivastava, A. Garg, Y. Cao *et al.*, “Echo state speech recognition,” *CoRR*, vol. abs/2102.09114, 2021.
- [14] S. Shen, A. Baevski, A. S. Morcos *et al.*, “Reservoir transformer,” *CoRR*, vol. abs/2012.15045, 2020.
- [15] A. Fan, E. Grave, and A. Joulin, “Reducing transformer depth on demand with structured dropout,” in *International Conference on Learning Representations*, 2020.
- [16] N. S. Chatterji, B. Neyshabur, and H. Sedghi, “The intriguing role of module criticality in the generalization of deep networks,” *CoRR*, vol. abs/1912.00528, 2019.
- [17] H. B. McMahan, E. Moore *et al.*, “Communication-Efficient Learning of Deep Networks from Decentralized Data,” in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, ser. Proceedings of Machine Learning Research, A. Singh and X. J. Zhu, Eds., vol. 54. PMLR, 2017, pp. 1273–1282.
- [18] Y. Wu and K. He, “Group normalization,” *CoRR*, vol. abs/1803.08494, 2018.
- [19] S. Caldas, J. Konečný, B. McMahan *et al.*, “Expanding the Reach of Federated Learning by Reducing Client Resource Requirements,” 2018.
- [20] V. Panayotov, G. Chen, D. Povey *et al.*, “Librispeech: An ASR Corpus Based on Public Domain Audio Books,” in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.
- [21] A. Misra, D. Hwang, Z. Huo *et al.*, “A Comparison of Supervised and Unsupervised Pre-Training of End-to-End Models,” in *Proc. Interspeech 2021*, 2021, pp. 731–735.
- [22] A. Narayanan, R. Prabhavalkar, C.-C. Chiu *et al.*, “Recognizing long-form speech using streaming end-to-end models,” in *IEEE Automatic Speech Recognition and Understanding Workshop, ASRU 2019, Singapore, December 14-18, 2019*. IEEE, 2019, pp. 920–927.
- [23] K. Hsieh, A. Phanishayee, O. Mutlu *et al.*, “The non-iid data quagmire of decentralized machine learning,” *CoRR*, vol. abs/1910.00189, 2019.
- [24] S. J. Reddi, Z. Charles, M. Zaheer *et al.*, “Adaptive federated optimization,” *CoRR*, vol. abs/2003.00295, 2020.
- [25] J. Wang, Z. Charles, Z. Xu *et al.*, “A Field Guide to Federated Optimization,” *CoRR*, vol. abs/2107.06917, 2021.