ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25jun167

Optimizing Speech Models with Freezing

Revanth Reddy Pasula[1]

[1]Department of Computer Science Wichita State University, Wichita, United States

Publication Date: 2025/07/14

Abstract: Adapting speech models to new languages requires an optimization of the trade-off between accuracy and computational cost. In this work, we investigate the optimization of Mozilla's DeepSpeech model when adapted from English to German and Swiss German through selective freezing of layers. Employing a strategy of transfer learning, we analyze the performance impacts of freezing different numbers of network layers during fine-tuning. The experiment reveals that freezing the initial layers achieves significant performance improvements: training time decreases and accuracy increases. This layer-freezing technique hence offers an extensible way to improve automated speech recognition for under-resourced languages.

Keywords: *Automatic Speech Recognition (ASR); Deep Speech; German; Layer Freezing; Low-Resource Languages; Swiss German; Transfer Learning.*

How to Cite: Revanth Reddy Pasula; (2025). Optimizing Speech Models with Freezing. *International Journal of Innovative Science and Research Technology,* (RISEM–2025), 69-73. https://doi.org/10.38124/ijisrt/25jun167

I. INTRODUCTION

ASR systems have improved mostly for the language of English, leading to very well-optimized models for speech tasks (e.g., text-to-speech systems [15]). In contrast, languages with few data sources—like standard German and Swiss German—are under-resourced because they lack large training sets and domain-specific models. The contribution of the current work is to bridge this gap, and we adapt Mozilla's DeepSpeech implementation[1] of Baidu's DeepSpeech architecture [1] to both German and Swiss German. We use transfer learning with a proven pre-trained model in English, and we thoroughly investigate the impact of the freezing of various network layers during fine-tuning.

Previous attempts at deploying DeepSpeech for German [2] and Swiss German [3] have delivered early evidence; nonetheless, differences in data composition and training methods call for further inquiries. In this research, emphasis is put into separating the effects of selective layer freezing and examining the contribution that it makes towards improving the performance of the recognizer while minimizing training time. The research is framed against the backdrop of modern developments in transfer learning methods and the growing interest in ensuring computationally efficient ASR model adaptation towards the use of limited resource environments.

II. TRANSFER LEARNING AND LAYER FREEZING

Transfer learning is now an essential method of deep learning where models are able to recycle knowledge acquired from one task or data set for use in another. Through pre-training of a network over an enormous, varied data set and then initializing with these pre-trained parameters, fine-tuning over a more modest target data set can be rendered more efficient both in terms of time and performance [4]. This exploits the hierarchical representations obtained through training of the network: following exposure to large quantities of data, the layers within the network have extracted helpful features that can be well-transferred to similar tasks without needing to begin from a zero starting point.

It is common practice in computer vision to freeze parts of pre-trained models during fine-tuning of the model for a novel task and keep previously acquired features [5]. The practice has been adapted in end-to-end ASR models such as DeepSpeech [4][6]. The idea is that the lower layers normally extract the basis acoustic patterns (comparable to the low-level visual features), whereas higher layers represent more abstract, language-dependent information. Current assessments of end-to-end ASR models show that, while the feature hierarchy of speech may not always be as apparent as in vision, the higher layers do represent higher order phonetic and linguistic features [7]. Practically, then, the earlier layers capturing common acoustic features can have their parameters frozen while enabling the higher layers to adapt to the new language

III. METHODOLOGY

An experimental framework was formulated to investigate the effects of layer freezing in the scenario of ASR under transfer learning. The methodology is comprised of four principal elements: the DeepSpeech architecture, training procedure with layer freezing settings, hyperparameters and computing environment, and dataset preparation as well as preprocessing pipeline

A. Deep Speech Architecture

The DeepSpeech version of version 0.7 from Mozilla was utilized as the base ASR architecture. The described implementation, deviating minimally from the model proposed originally by Hannun et al. [1], is documented in greater detail in the official documentation[2]. The processing pipeline starts with the MFCC [8] extraction from the raw audio input, followed by a total of six layers with the form of a deep recurrent neural network. The network structure is shown in Table I. Briefly, layers 1–3 are ReLU-activated fully connected layers, layer 4 is an LSTM recurrent layer [11], layer 5 is an additional fully connected but ReLU-activated layer, and layer 6 is the output layer generating character probabilities through the use of softmax. The model is trained with the use of the Connectionist Temporal Classification (CTC) loss [9] and optimization with the use of the Adam optimizer [10].

Table 1 shows the DeepSpeech architecture and data flow, from input audio to feature extraction to output character probabilities (figure adapted from the official documentation).

Table 1 Structure of the DeepSpeech Architecture.

Layer	Description	Activation/Notes
1–3	Fully connected	ReLU
4	Recurrent (LSTM)	Long Short-Term Memory [11]
5	Fully connected	ReLU
6	Output layer	Softmax (character probabilities)

B. Training Procedure and Layer Freezing

We performed a series of training experiments to measure the effect of frozen layers in transfer learning. For weight initialization, we utilized an English pre-trained DeepSpeech model offered by Mozilla. Six training setups were done for both German and Swiss German, which are compiled in Table II. Moreover, we trained one model entirely from scratch with random initialization as our baseline comparison point (labeled the “Reference” condition with no transfer learning).

During fine-tuning, the mentioned layers were frozen by indicating them as non-trainable, while the rest of the layers were trained over the target data. All the transfer learning models' output layer was re-initialized, as the character set (output labels) was different for English compared to the target language. This re-initialization provided compatibility with German or Swiss German transcripts.

Table 2 Training Conditions for Evaluating the impact of layer freezing.

Condition	Description
Reference	Trained from scratch (random initialization, no pre-trained model).
0 Frozen Layers	Initialized from the English model; all layers are fine-tuned on target data.
1 Frozen Layer	Freeze the first layer; fine-tune layers 2–6 on target data.
2 Frozen Layers	Freeze the first two layers; fine-tune layers 3–6 on target data.
3 Frozen Layers	Freeze the first three layers; fine-tune layers 4–6 on target data.
4 Frozen Layers	Freeze the first four layers; fine-tune only the last two layers on target data.

C. Hyperparameters and Computational Environment

The same set of hyperparameters was utilized in all experiments (Table III), with no further tuning aside from these preselected values. Training was carried out under a Linux server with 96 Intel Xeon Platinum 8160 CPU cores. All models (both German and Swiss German) were trained over the same number of epochs under the same conditions to make an unbiased comparison of the different freezing strategies.

Table 3 Hyperparameter Settings Utilized when Training.

Hyperparameter	Value	Notes
Batch Size	24	–
Learning Rate	0.0005	–
Dropout Rate	0.4	–
Training Epochs	30	Per model (each experiment)
Optimizer	Adam	–

D. Datasets and Preprocessing

The data we used for our experiments are tabulated in Table IV. For the German models, we utilized the training data from Mozilla’s German corpus [12]. This data comprises around 315 hours of speech, provided by about 4,823 speakers, with utterances lasting around 3 to 5 seconds. For the Swiss German models, we drew upon an even smaller dataset of 70 hours of Swiss German speech derived from Bernese parliamentary debates [13]. The Swiss German dataset covers formal speaking with relatively few speakers (around 191), and its size is considerably lower compared to the German corpus.

The initial model for the English DeepSpeech model was trained with a much larger dataset (over 6500 speech audio hours of English data) aggregated from heterogeneous sources such as LibriSpeech and the English part of the Common Voice dataset (more information can be found in footnote 5). Before training, all data sets underwent common preprocessing steps, for example, audio-normalization and cleaning of transcript text (e.g. lowercasing, punctuation removal), to make them consistent. Table V contains an itemized description of each component of the data set, as well as the preprocessing pipeline.

Along with the acoustic data, we used an external language model at time of inference to enhance the accuracy of the recognizer. To do this, we trained a tri-gram language model with the KenLM toolkit [14] over a large corpus of text consisting of public domain German-language text from Wikipedia articles and Europarl parliamentary debates. This language model we incorporated into the DeepSpeech decoder for both German and Swiss German trials, helping the system to make more accurate transcripts through language contextualization.

Table 4 Summary of the Datasets used for Training.

Dataset	Language	Hours of Audio	Number of Speakers
Pre-training	English	> 6500	—
Training	German	315	4,823
Training	Swiss German	70	191

Table 5 Description of each Dataset and key Preprocessing Details.

Component	Description
German Dataset	Collected from Mozilla Common Voice; crowd-sourced speech with diverse speakers; average utterance length ~3–5 seconds.
Swiss German Dataset	Collected from Bernese Parliament speeches; formal register, fewer speakers; significantly lower volume of data compared to the German set.
English Pretraining	Combined from large-scale English corpora (LibriSpeech + Common Voice English); provides broad acoustic coverage for subsequent adaptation.

IV. RESULTS AND DISCUSSION

We compared the performance of the six training schemes in terms of word error rate (WER) and character error rate (CER) on test sets for both German and Swiss German. Table VI shows the WER and CER seen by each model configuration for German, while Table VII shows the WER and CER for Swiss German. “Reference” in these tables indicates the model trained from scratch without any transfer learning, and the “Improvement” column shows the percentage point improvement in WER with respect to that baseline.

For the German ASR task, the baseline model of training without any transfer learning obtained a WER of 70.0% with CER of 42.0%. Employing the pre-trained model for English with no frozen layers (0 frozen, full fine-tuning) reduced the WER to 63.0% (CER 37.0%), which is only a modest improvement of 7.0 points. However, partial freezing of the initial layers produced much greater improvements. Simply freezing the first layer improved the WER to 48.0% (CER 26.0%), which is a 22-point WER improvement over the baseline. Freezing the first two layers improved the WER further to 44.0% (CER 22.0%), which is the best performance and an improvement of 26 points over the baseline. Significantly, two or three frozen layers showed the same WER (44.0%), which means that there would have been no additional improvement from the third layer over the first two. With four frozen layers, performance actually decreased slightly with an increase in WER to 46.0% and CER to 25.0%, though still significantly better than the baseline. These are all indications that, for German, retaining the lower-level layers of the acoustic features (up to two or three layers) gives the best result, significantly outperforming the baseline and the full fine-tuned model.

For Swiss German, we see the same pattern with differing magnitude. The baseline Swiss German model (no-transfer) achieved a WER of 74.0% (CER 52.0%). Fine-tuning all model layers on Swiss German data (0 frozen) caused the WER to worsen slightly to 76.0%, which shows that such indiscriminate fine-tuning with no freezing can overfit or mis-adapt to the small Swiss German corpus. Freezing the early layers, in contrast, worked: with one frozen, the WER improved to 69.0% (CER 48.0%), about a 5-point improvement over the baseline, and with two frozen, the WER further improved to 67.0% (CER 45.0%), which was the best performance for Swiss German (a 7-point improvement over baseline). Freezing three or four layers showed no additional improvements (WER ~68.0% in each case, ~6 points improvement over baseline). So, for Swiss German, the first two layers of pre-trained model freezation provided the greatest improvement, with freezation beyond two not bringing an additional advantage and retaining approximately the same performance.

In total, selective freezing of layers resulted in notably improved accuracy for both languages over training from scratch. The advantage was particularly dramatic for German, with the larger dataset; the method of transfer learning reduced the WER by more than 26 absolute points. Swiss German, with the much smaller dataset and higher dialectal variation, also showed improved performance with transfer learning, though the relative improvement fell short. These results show that the lowest layers of the deep model encode general acoustic representations that are relevant for many languages. By holding these layers constant, the fine-tuning procedure can concentrate on adapting higher-level layers to the target language's idiosyncrasies. But freezing too many layers starts to restrict the model's flexibility: the modest decline in performance when four layers were frozen indicates that the model required some of the later layers to adapt to language-specific features.

Interestingly, we found that models with varying numbers of frozen layers showed very comparable training convergence patterns. This suggests that retaining the pre-trained low-level feature extractors didn't impede training; the models all trained at around the same speed, just they achieved different ending accuracy levels depending upon the number of layers updated. This result indicates that much of the key learning of the new language happens higher up in the model after an effective set of building block features is established.

Table 6 below presents the performance of the different training strategies for German, and Table VII presents the respective results for Swiss German. Table VIII presents a high-level comparison of each language's optimal freezing configurations and the resultant error rates.

Table 6 German ASR Performance with Various Layer-Freezing Strategies (WER = Word Error Rate, CER = Character Error Rate). The Improvement Column Indicates WER Improvement Compared to the Baseline (Reference) Model.

Training Strategy	WER (%)	CER (%)	WER Improvement
Reference (No Transfer; Random Init.)	70.0	42.0	—
0 Frozen Layers (Full fine-tuning)	63.0	37.0	+7.0
1 Frozen Layer	48.0	26.0	+22.0
2 Frozen Layers	44.0	22.0	+26.0
3 Frozen Layers	44.0	22.0	+26.0
4 Frozen Layers	46.0	25.0	+24.0

Table 7 ASR Performance for Swiss German under Different Layer-Freezing Strategies.

Training Strategy	WER (%)	CER (%)	WER Improvement
Reference (No Transfer; Random Init.)	74.0	52.0	—
0 Frozen Layers (Full fine-tuning)	76.0	54.0	–2.0
1 Frozen Layer	69.0	48.0	+5.0
2 Frozen Layers	67.0	45.0	+7.0
3 Frozen Layers	68.0	47.0	+6.0
4 Frozen Layers	68.0	46.0	+6.0

Table 8 Summary of Optimal Performance Results Across Languages.

Language	Optimal # of Frozen Layers	Best WER (%)	Best CER (%)
German	2–3	44.0	22.0
Swiss German	2	67.0	45.0

V. CONCLUSION

Finally, we have shown in this work that transfer learning along with selective layer freezing can be an affordable approach to enhance ASR systems for low-resource languages. We experimented heavily with Mozilla's DeepSpeech setup on German and Swiss German and could confirm that by initializing the network from a pre-trained English model and freezing the initial layers, recognition performance can be improved significantly. The best gains were found in models with two to three frozen layers, suggesting that low-level phonetic features that English ASR systems learned are highly transferable. By preserving them with a frozen model, the fine-tuning can more quickly specialize the higher layers of the model to the target language. On the other hand, models trained from scratch (i.e., no pre-training and no transfer learning) performed significantly worse, demonstrating the utility of abundant English data in low-resource settings.

Our investigation demonstrated that higher layer wise freezing for Swiss German (after the second layer) and for German (after the third layer) leads to no further improvements in accuracy, but the selective freezing of higher dense layers is still very advantageous. It not only increases the accuracy but also makes the fine-tuning more easily by decreasing the trainable parameters. There seems to be a trade-off between keeping pre-learned representations and enough flexibility for language-specific adaptation. Freezing more layers (even four) harms the performance slightly, and thus the higher layers still need some retraining to handle the nuances of the target language. This trade-off probably depends on the amount and quality of the training data available in the target language, and deserves further exploration.

Overall, selective layer freezing transfer learning is a powerful technique for closing the performance gap between high and low resource languages in speech recognition. The results also motivate additional research into adaptive freezing strategies (e.g., deciding at runtime which layers to freeze), and demonstrate the potential of such methodology towards building scalable, robust, and computationally efficient multilingual ASR systems. Further studies in this area are likely to result in ASR technology that is more accessible across languages and dialects, and thus increase the inclusivity of ASR systems globally.

FUTURE WORK

Further, future work should focus on improving this layer-freezing method and extending it to other models and languages. Another direction is to study how to optimize the selective layer freezing strategies by taking more adaptive or dynamic ways. For instance, optimal number of frozen layers can be modified according to target dataset size and quality, as the trade-off of preserving prelearned features to adapt can be different. It remains for future work whether techniques for being able to automatically determine or gradually unfreeze the weights of layers could also benefit performance. It would be interesting to try other pretrained model or models as the base. Testing the freezing strategy on other state-of-the-art ASR models would reveal whether the gains achieved are consistent across different network designs and potentially use richer pretrained representations for improving performance. In addition, an interesting direction is to extend the proposed transfer- learning to multiple languages or more complex datasets to examine its generality. Finally, it will be important to apply the presented method to languages outside of German and Swiss German (other language families, and also languages with phonetic characteristics quite different from what was considered here) in order to see whether the low-level acoustic features learnt from English can generally be successfully employed, or whether fine-tuning at the language specific level is required. Likewise, generalizing to more challenging and/or more diverse datasets (such as larger speech corpora with more speakers, dialectal variability and noisier audio), is important to evaluate the robustness of the method in real-world settings. Such experiments would help confirming the effectiveness of approach in multilingual setting and also provide practical optimizations for scalable and efficient speech model adaptation for low resource scenarios.

REFERENCES

[1]. A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Y. Ng, “Deep speech: Scaling up end-to-end speech recognition,” 2014.

[2]. A. Agarwal and T. Zesch, “German end-to-end speech recognition based on DeepSpeech,” *Proc. of the 15th Conf. on Natural Language Processing (KONVENS 2019): Long Papers*, Erlangen, Germany: German Society for Computational Linguistics & Language Technology, 2019, pp. 111–119.

[3]. “LTL-UDE at low-resource speech-to-text shared task: Investigating Mozilla DeepSpeech in a low-resource setting,” 2020.

[4]. J. Kunze, L. Kirsch, I. Kurenkov, A. Krug, J. Johannsmeier, and S. Stober, “Transfer learning for speech recognition on a budget,” in *Proc. of the 2nd Workshop on Representation Learning for NLP (RepL4NLP@ACL 2017)*, Vancouver, Canada, Aug. 2017, Association for Computational Linguistics, pp. 168–177.

[5]. M. Huh, P. Agrawal, and A. A. Efros, “What makes ImageNet good for transfer learning?” 2016.

[6]. B. Li, X. Wang, and H. S. M. Beigi, “Cantonese automatic speech recognition using transfer learning from Mandarin,” *CoRR*, 2019.

[7]. Y. Belinkov and J. Glass, “Analyzing hidden representations in end-to-end automatic speech recognition systems,” in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 2441–2451.

[8]. S. Imai, “Cepstral analysis synthesis on the mel frequency scale,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP’83)*, vol. 8, 1983, pp. 93–96.

[9]. A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks,” in *Proc. of the 23rd International Conference on Machine Learning*, 2006, pp. 369–376.

[10]. D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014.

[11]. S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[12]. R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, “Common Voice: A massively-multilingual speech corpus,” in *Proc. of The 12th Language Resources and Evaluation Conference (LREC 2020)*, Marseille, France, May 2020, European Language Resources Association, pp. 4218–4222.

[13]. M. Plüss, L. Neukom, and M. Vogel, “GermEval 2020 Task 4: Low-resource speech-to-text,” 2020.

[14]. K. Heafield, “KenLM: Faster and smaller language model queries,” in *Proc. of the 6th Workshop on Statistical Machine Translation*, Association for Computational Linguistics, 2011, pp. 187–197.

[15]. M. Schröder and J. Trouvain, “The German text-to-speech synthesis system MARY: A tool for research, development and teaching,” *International Journal of Speech Technology*, vol. 6, no. 4, pp. 365–377, 2003.