INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

On the Learning Dynamics of Semi-Supervised Training for ASR

Electra Wallington, Benji Kershenbaum, Ondřej Klejch, Peter Bell

Centre for Speech Technology Research, University of Edinburgh, United Kingdom
{electra.wallington, s1709221, o.klejch, peter.bell}@ed.ac.uk

Abstract

The use of semi-supervised training (SST) has become an increasingly popular way of increasing the performance of ASR acoustic models without the need for further transcribed speech data. However, the performance of the technique can be very sensitive to the quality of the initial ASR system. This paper undertakes a comprehensive study of the improvements gained with respect to variation in the initial systems, the quantity of untranscribed data used, and the learning schedules. We postulate that the reason SST can be effective even when the initial model is poor is because it enables utterance-level information to be propagated to the frame level, and hence hypothesise that the quality of the language model plays a much larger role than the quality of the acoustic model. In experiments on Tagalog data from the IARPA MATERIAL programme, we find that indeed this is the case, and show that with an appropriately chosen recipe it is possible to achieve over 50% relative WER reductions from SST, even when the WER of the initial system is more than 80%.

Index Terms: speech recognition, semi-supervised training

1. Introduction

Traditional approaches to acoustic model (AM) training for automatic speech recognition (ASR) rely on large quantities of hand-transcribed acoustic training data. However, this presents a significant barrier to the development of systems for the vast majority of the world's languages, where such resources are minimal or non-existent due to the high cost of transcription. Even in well-resourced languages, many situations require new models tailored to a particular domain; transcribing such new data too is often infeasible.

In semi-supervised training (SST) an initial 'seed' ASR system – typically trained on small amounts of transcribed data, possibly mismatched to the domain of interest – is used to provide 'pseudo-labels' for an untranscribed data-set (henceforth termed 'semi-supervised' data). In the classic formulation [1], a new AM is then trained (in the traditional manner) using these pseudo-labels as transcription.

Whilst useful for low-resource settings, a noted difficulty with SST is finding ways to avoid performance degradation due to training on erroneous pseudo-labels. To this end, most approaches have performed some form of confidence filtering of the semi-supervised data [2, 3, 4]. However, in so doing, we note that there is a risk of selecting only the easiest utterances from the new data, which may simply reinforce the seed model's initial predictions and decision boundaries. The problem is particularly acute in the case when the seed model's quality is poorer; here, very few utterances may be selected.

The work of [5] solves this conundrum by using lattices to encode uncertainty in the SST labels. Differently to earlier work [6] on lattice-based SST, the use of the LF-MMI criterion in [5] enabled for the first time a correct calibration of confidence levels between word-based lattices from decoding with the seed model, and state-based confidences from the AM alone. Importantly, this approach allows utilization of a much larger pool of training utterances, including those more challenging utterances constituting the most useful training examples.

Turning to our work, we believe that the success of [5] illuminates the following hypothesis: that the key to SST's effectiveness for ASR is allowing the AM to utilize information contained at the sequence level – typically utterance level – to reduce the frame level labelling uncertainty naturally obtained by the initial model. This utilizing of external sequence-level information differentiates SST for ASR from examples in other fields such as [7], and also calls for more explicit consideration of what such sequence/frame-level distinction means for SST learning dynamics. Note that SST has recently been applied to end-to-end ASR models [8, 9, 10] but, in accordance with our hypothesis, these works obtained improvements only when starting from relatively good initial models: pseudo-labels were filtered one-best transcripts from first-pass decoding.

It is therefore important to consider how sequence level information is introduced into the SST method: via the language model (LM), implicitly incorporated by decoding or rescoring. The LM's role and importance has to our knowledge not been explicitly considered in the SST literature, nor has a systematic study of precise AM-LM relationship dynamics (as independent components) been conducted. Yet, whilst it is intuitive – and has been explored in e.g. [11, 12] – that initial system quality directly affects subsequent SST success, we here hypothesize that optimally-performed SST may specifically be much more sensitive to the quality of the LM than the AM or system as a whole: for a sufficient LM may be able to counteract the weaknesses of a poor AM (hence poor frame-level predictions), allowing for SST to make gains in situations otherwise lost to error-propagation (c.f. [13]). By establishing our own semi-supervised pipeline and comparing how varying-quality seed AMs, varying-quality LMs, and, importantly, combinations of such AMs and LMs, affect final Word Error Rate (WER) achieved, we explore how seed acoustic model (AM) quality and, *independently*, language model (LM) quality impact upon SST success.

We also consider incremental semi-supervised training ('iSST'). Previous works [11, 4, 12] have shown iSST to benefit scenarios where supervised AM data is lacking: more iterations of training/decoding lessens reliance on seed model predictions and can result in final WERs lower than those achieved by 'one-shot' approaches. Not explicitly considered in the literature however is how iSST interacts with LM quality. For whilst iSST benefits poor seed AM scenarios, we believe iSST may, in contrast, be particularly harmful for poor LM scenarios: more increments means more decoding iterations, and hence, if using a poor LM, the more times low quality LM information enters the system and impacts upon semi-supervised predictions.

In sum, our paper's contribution is as follows. We develop

http://dx.doi.org/10.21437/Interspeech.2021-1777

a systematic understanding of the limits of, and optimal conditions for, SST. Under a Hybrid HMM-DNN framework, with a suite of experiments run on Tagalog speech, we contribute such a systematic examination through a novel lens, assessing how SST is sensitive to, and impacted by, quality of seed AM and, *independently*, LM. A second series of experiments extends this to iSST, asking: can iSST recover from a low-quality LM as it can from a low-quality AM?

2. Data and Methods

We carry out SST experiments on a Tagalog task from the IARPA MATERIAL programme [14]. The ASR task comprises of diverse acoustic data drawn from news and topical broadcast genres (including vlog-style content). Comparatively, the seed model acoustic training data consists purely of telephone conversations and is therefore significantly mismatched to the target domain. Note that the MATERIAL programme mirrors this setup across many languages: though we focus purely on Tagalog here due the intensive nature of our experiments, we have also found our approach successful on languages including Somali [15], Pashto, Kazakh and more.

Our experiments aim to assess the extent to which the WER improvements from SST depend on the quality of the initial system and, specifically, how this behaviour is affected by the quality of the AM and LM components. To do this, we artificially degrade both models by training them on progressively reduced subsets of the original data. Following initial analysis, we use these same varying-quality models to investigate iSST.

2.1. Language Models

The LM training data consists of the Tagalog CommonCrawl data-set[1] and four smaller Tagalog sets provided by our project partners from Columbia University. We train a full 3-gram LM with Kneser-Ney smoothing using the SRILM toolkit [16] on all five data-sets. We then train seven LMs on randomly selected subsets of the CommonCrawl data, each subset being four times smaller than the previous one. Since the CommonCrawl data-set contains information about URL sources for each sentence, we perform random subsampling at the URL level, as we believe this method better emulates how LM-quality would degrade in a low-resource scenario. All LMs use a maximum vocabulary size of 300k words and a pruning threshold of 1e-9. We report perplexity of these LMs and WER when decoding with the full AM in Table 1.

2.2. Seed Acoustic Models

The seed AM training data consists of 85 hours of transcribed narrow-band (8kHz) telephone conversation data from 966 speakers from the Babel Tagalog build pack [17]. Mirroring the LM setup, we first train a full seed model on this complete supervised data-set and then, to train 6 further increasingly-degraded AMs, successively reduce training data by half by random sampling at the speaker level. The seed models are trained following a standard LF-MMI recipe [18] using a CNN-TDNNF neural network architecture, with 40-dimensional MFCC features as input together with i-vectors for speaker adaptation [19, 20]. The models are trained with natural gradient [21] for 6 epochs and use Dropout [22] and SpecAugment [23] for regularization. Table 2 reports these seed models' WERs when decoding with the full LM.

Table 1: *LMs trained on decreasing amounts of training data and evaluated with full AM.*

% data	# Tokens	# OOV	PPL	% WER
full	388.1M	0.5k	578.9	35.3
4^{-1}	97.1M	8.8k	664.1	38.4
4^{-2}	24.2M	8.9k	674.0	39.7
4^{-3}	6.1M	9.5k	700.3	42.3
4^{-4}	1.5M	10.2k	717.8	45.3
4^{-5}	378.1k	11.7k	731.4	51.1
4^{-6}	95.1k	13.6k	751.8	54.3
4^{-7}	23.5k	17.2k	770.6	61.3

Table 2: *Seed AMs trained on decreasing amounts of training data and evaluated with full LM.*

% data	# Hours	# Speakers	% WER
full	85.1	966	35.3
2^{-1}	42.5	480	38.5
2^{-2}	21.3	244	43.1
2^{-3}	10.6	118	51.1
2^{-4}	5.3	62	60.7
2^{-5}	2.6	32	73.6
2^{-6}	1.3	16	83.4

2.3. Semi-supervised data collection and pre-processing

To obtain further acoustic Tagalog data for SST, we scrape YouTube videos by querying the most common Tagalog trigrams in the full LM. Because this data is likely to be noisy - containing audio other than speech, and languages other than Tagalog - we employ filtering at the video level. To do this we decode the data using the full seed AM and LM and discard videos with resulting mean confidence falling below 0.7. We also discard videos where average speaking rate (of speech identified by VAD) falls below 1.25 words per second: we find this helps to filter out non-speech data such as music videos where confidence levels can be erroneously high. The thresholds were set by comparing the distributions of these parameters' values for our raw scraped data against those for our development set, which we know to be of good quality, and with an out-of-language data-set. See Figure 1.

After pre-processing, we are left with a wide-band (16kHz) semi-supervised data-set of 400 hours. For initial lattice generation (when applying initial seed models), this data is downsampled to 8kHz: as mentioned, the seed AMs are trained purely on 8kHz data. However, we train all subsequent models on the original wide-band features. Whilst this choice does mean – unlike much other work – we are unable to incorporate *any* supervised data into the final models, in [15] we found that when the lattice-based SST recipe is tuned correctly, this is a beneficial trade-off to make to allow the wide-band features to be used.

2.4. Incremental semi-supervised training

In the standard SST recipe using LF-MMI [5], all unlabelled data is decoded at once with the seed model; the subsequent SST model is then trained on all newly-transcribed data. In our incremental training (iSST) setup comparatively we split the data into n equally-sized chunks. When processing the i-th

[1] http://data.statmt.org/ngrams/raw/

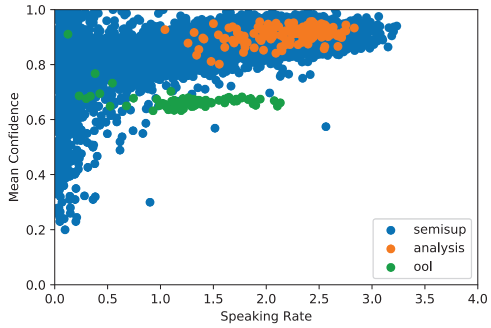

Figure 1: *Plotting speaking rate against mean lattice confidence for each utterance in: our raw Youtube data ('semisup'); our analysis data-set; and out-of-language data ('ool').*

chunk we use the model produced by training on the previous chunk to decode the current chunk and then continue training that model with just that chunk. In contrast to approaches such as [11], we never train twice on the same chunk as we find this can quickly lead to over-fitting. As discussed above, the seed model cannot be used so we commence SST with a randomly-initialized model. We use the exponential decay training schedule [4] for the continued training. The initial learning rate when processing the i-th chunk, lr_i, is computed from the global initial learning rate lr_0 and the global final learning rate lr_n as:

$$lr_i = lr_0 * exp\left(\frac{i}{n} log\frac{lr_n}{lr_0}\right). \tag{1}$$

The final learning rate for the i-th chunk is equal to lr_{i+1}. In the experiments we set $lr_0 = 1.5 \times 10^{-4}$ and $lr_n = 1.5 \times 10^{-5}$. This ensures that the iSST learning rate schedule is identical to our standard SST setup's learning rate schedule.

3. Results

We evaluate all models (seed and post-SST) on a fixed 9.5 hour, wide-band test set provided by MATERIAL (downsampled for seed evaluation). All figures in this section plot the WER performance of a final system following SST against WER of the corresponding initial seed system, a format which we find best illustrates the gains to be made from the technique. All plots include the $y = x$ line, to mark the case where there is no gain from SST; points below the line indicate a gain. Importantly, because SST is applied to the AM only, we always use a consistent LM for decoding with the initial and final model, so as to be able to directly assess the benefits of SST.

3.1. SST sensitivity to AM vs. LM quality

We first run our semi-supervised pipeline with each of Table 2's seed AMs, keeping LM and amount of semi-supervised data constant (full LM; 200 hours of data). Thus we ask: how sensitive is SST to quality of seed AM? Importantly, can we rely on a good-quality LM to compensate for even poor AMs? Or are AMs below some threshold so poor that their erroneous predictions cannot be mitigated even with external LM information? Figure 2a shows that SST, when decoding with full LM, enabled 'recovery' from all qualities of seed AM tested. Hence, this figure supports our hypothesis that a reliable LM can very often compensate for a poor seed model, lessening overall SST sensitivity to AM quality. Even when employing our poorest seed AM, SST still led to WER gains of up to 50% relative, though note the graph does suggest that SST will eventually fail at exceptionally high WERs (even with a high-quality LM). Interestingly an extrapolation of the curve in the lower WER range implies that attempting SST under our recipe with too high-quality a seed AM may actually also be detrimental. Possibly this is because such a high-quality seed would have had to have been trained on significant quantities of well-matched transcribed data and hence retraining such model from scratch with SST would downgrade the model's diversity/robustness. Overall, these experiments demonstrate SST can be very beneficial even when starting with a particularly poor AM, provided the LM is of sufficient quality.

Next we ask: how sensitive is SST to LM quality? In a set of experiments mirroring those above, we run our semi-supervised pipeline with each of Table 1's LMs, keeping the full seed AM and amount of semi-supervised data constant (200 hours). Figure 2a shows the resulting 'varying LM' curve's shape is radically different to that when the AM is varied. Importantly, this suggests high LM-quality sensitivity. First, a lack of plateau in the lower WER range indicates improving LM quality leads to increasingly greater pay-off in terms of SST leverage (apparently suggested is that WER could continue to be reduced even to zero with improving LM quality, though bear in mind that in practice this movement along the curve for a fixed AM would require exponential increases in LM training data quantity). Second, this curve's crossing of $y = x$ at a comparatively low initial WER indicates that a poor LM is detrimental for SST.

We run two further sets of experiments, again systematically varying AM and/or LM, but this time holding the worst AM/LM constant. Results in Figure 2b show that moving from consistently decoding with our full LM to our worst LM is directly paralleled by a decrease in SST's ability to leverage the semi-supervised data to make gains. That this is true across all qualities of seed AM employed, plus the 2b curve's faster plateauing, further emphasizes how reliant SST is on a good LM. Comparatively, Figure 2b's 'Varying LM' curve is similar in trajectory to its Figure 2a counterpart despite switching full for worst AM, further evidence that a poor AM does not place the same ceiling on SST-gains as a poor LM.

Practically, these findings suggest SST to be most valuable to low-resource settings where sufficient LMs can be built. Improving a LM is likely to be the greatest factor for SST success in such scenarios. Without access to adequate text data however, what can be achieved with SST may be fundamentally limited.

3.2. iSST sensitivity to AM vs. LM quality

To assess iSST's sensitivity to AM versus LM, we partition our 400 hour semi-supervised data-set into series of finer-grained data-sets (to evaluate effect of increment size): 1x400 hours; 2x200 hours; 4x100 hours; 8x50 hours and 16x25 hours. We then compare these increment sizes with systems employing varying seed AMs (whilst utilizing the full LM). In parallel, we assess how these increment sizes vary in utility when differing LMs are employed (with seed AM constant).

Note we first validated our use of the iSST learning rate (LR) schedule detailed in Section 2.4 for these experiments by comparing to a schedule which resets LR after each increment. Indeed when evaluated with 2x200 increments, the latter achieves 29.0 WER; the former 27.4.

We report results in Figures 3a and 3b. In line with

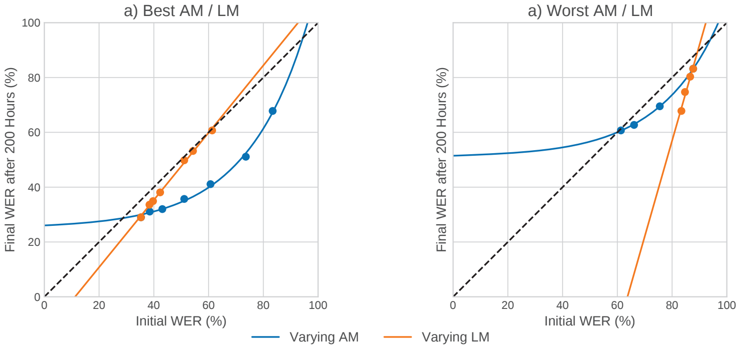

Figure 2: *How varying the seed AM (blue) and LM (orange) affects WER gains made during SST.*

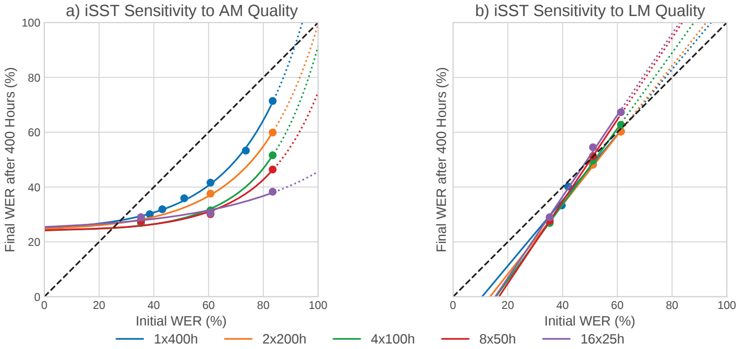

Figure 3: *How varying the iSST regime affects WER gains made during SST, for systems with different seed AMs (a) and LMs (b).*

[4, 11, 12], Figure 3a indicates that iSST with increasingly finer-grained increments benefits limited supervised acoustic data settings (though note as seed AM quality increases this advantage lessens): finer increments entail more iterations of increasing lattice accuracy. Importantly, when varying LM quality (Figure 3b), the opposite is true. When employing our worst LM for decoding, iSST with increments smaller than 200 hours actually degrades system performance. This makes sense: the more times a poor LM is incorporated into the pipeline, the more opportunities for incorrect information to be integrated into labelling predictions/decisions. Overall then, in investigating iSST utility explicitly in relation to AM *and* LM quality, we emphasize again how integral it is to consider LM and AM separately when deciding whether to conduct SST or when establishing a SST pipeline. For though recent studies have shown iSST to lead to gains, it is evident that iSST should not be incorporated into SST pipelines alongside a poor LM.

4. Future Work and Conclusions

To conclude, our experiments provide multiple sources of evidence for the notion that LM quality can be instrumental *or* detrimental to SST. This has important practical implications: for settings in which a good LM can be utilized, initial seed AM quality becomes far less important (an intuition less easily reached without explicit consideration of the LM's role independently of the seed), thus providing a fruitful avenue for building ASR systems in low-resource scenarios. Although our experiments have focused solely on traditional hybrid-HMM systems, we believe our results have important implications for understanding how to achieve good improvements from SST on end-to-end systems. Here, external LM data is notably harder to incorporate directly into the model: this could explain why, to date, SST gains on these systems have been found only when initial system is relatively good.

In the future, we wish to conduct further systematic charting of AM-LM interactions: this would facilitate increasingly reliable predictions of how SST would react in real-world, low-resource settings. The LM's role in relation to lattice rescoring could also be considered. Also worth exploring is precisely how WER changes as a function of the amount of semi-supervised data supplied. Is there a monotonically positive relationship between amount of unlabelled data and final performance? Or do SST gains lessen or converge as data is added? Finally, we consider it worth exploring how self-supervised representation learning interacts with SST, particularly whether utilizing such learning to provide more robust speech representations could be a method of compensating for poor-LM-quality scenarios.

5. Acknowledgements

This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via Air Force Research Laboratory (AFRL) contract #FA8650-17-C-9117. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, AFRL or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein. This work was also partially supported by EPSRC Project EP/T024976/1 (Unmute).

6. References

[1] L. Lamel, J.-L. Gauvain, and G. Adda, “Unsupervised acoustic model training,” in *ICASSP*, 2002.

[2] K. Veselý, M. Hannemann, and L. Burget, “Semi-supervised training of deep neural networks,” in *ASRU*, 2013.

[3] T. Drugman, J. Pylkkonen, and R. Kneser, “Active and semi-supervised learning in ASR: Benefits on the acoustic and language models,” *arXiv preprint arXiv:1903.02852*, 2016.

[4] S. H. K. Parthasarathi and N. Strom, “Lessons from building acoustic models with a million hours of speech,” in *ICASSP*, 2019.

[5] V. Manohar, H. Hadian, D. Povey, and S. Khudanpur, “Semi-supervised training of acoustic models using lattice-free MMI,” in *ICASSP*, 2018.

[6] T. Fraga-Silva, J.-L. Gauvain, and L. Lamel, “Lattice-based unsupervised acoustic model training,” in *ICASSP*, 2011.

[7] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, “Self-training with noisy student improves imagenet classification,” in *CVPR*, 2020.

[8] J. Kahn, A. Lee, and A. Hannun, “Self-training for end-to-end speech recognition,” in *ICASSP*, 2020.

[9] G. Synnaeve, Q. Xu, J. Kahn, E. Grave, T. Likhomanenko, V. Pratap, A. Sriram, V. Liptchinsky, and R. Collobert, “End-to-end ASR: from supervised to semi-supervised learning with modern architectures,” *arXiv preprint arXiv:1911.08460*, 2019.

[10] D. S. Park, Y. Zhang, Y. Jia, W. Han, C.-C. Chiu, B. Li, Y. Wu, and Q. V. Le, “Improved noisy student training for automatic speech recognition,” in *Interspeech*, 2020.

[11] B. Khonglah, S. Madikeri, S. Dey, H. Bourlard, P. Motlicek, and J. Billa, “Incremental semi-supervised learning for multi-genre speech recognition,” in *ICASSP*, 2020.

[12] Q. Xu, T. Likhomanenko, J. Kahn, A. Hannun, G. Synnaeve, and R. Collobert, “Iterative pseudo-labeling for speech recognition,” *arXiv preprint arXiv:2005.09267*, 2020.

[13] A. Srinivasamurthy, P. Motlicek, M. Singh, Y. Oualil, M. Kleinert, H. Ehr, and H. Helmke, “Iterative learning of speech recognition models for air traffic control.” in *Interspeech*, 2018.

[14] C. Rubino, “IARPA Material program,” *http://www.iarpa.gov/Programs/ia/Babel/babel.html*.

[15] A. Carmantini, P. Bell, and S. Renals, “Untranscribed web audio for low resource speech recognition,” in *Interspeech*, 2019.

[16] A. Stolcke, “SRILM-an extensible language modeling toolkit,” in *Seventh international conference on spoken language processing*, 2002.

[17] M. Harper, “IARPA Babel program,” *http://www.iarpa.gov/Programs/ia/Babel/babel.html*.

[18] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, “Purely sequence-trained neural networks for asr based on lattice-free MMI,” in *Interspeech*, 2016.

[19] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, “Front-end factor analysis for speaker verification,” *IEEE TASLP*, vol. 19, no. 4, pp. 788–798, 2010.

[20] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, “Speaker adaptation of neural network acoustic models using i-vectors,” in *ASRU*, 2013.

[21] D. Povey, X. Zhang, and S. Khudanpur, “Parallel training of DNNs with natural gradient and parameter averaging,” *arXiv preprint arXiv:1410.7455*, 2014.

[22] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *JMLR*, vol. 15, no. 1, pp. 1929–1958, 2014.

[23] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “SpecAugment: A simple data augmentation method for automatic speech recognition,” *Interspeech 2019*, 2019.