INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Transliteration Based Data Augmentation for Training Multilingual ASR Acoustic Models in Low Resource Settings

Samuel Thomas, Kartik Audhkhasi, Brian Kingsbury*

IBM Research AI, Yorktown Heights, USA

Abstract

Multilingual acoustic models are often used to build automatic speech recognition (ASR) systems for low-resource languages. We propose a novel data augmentation technique to improve the performance of an end-to-end (E2E) multilingual acoustic model by transliterating data into the various languages that are part of the multilingual training set. Along with two metrics for data selection, this technique can also improve recognition performance of the model on unsupervised and cross-lingual data. On a set of four low-resource languages, we show that word error rates (WER) can be reduced by up to 12% and 5% relative compared to monolingual and multilingual baselines respectively. We also demonstrate how a multilingual network constructed within this framework can be extended to a new training language. With the proposed methods, the new model has WER reductions of up to 24% and 13% respectively compared to monolingual and multilingual baselines.

Index Terms: speech recognition, data augmentation, multilingual data, end-to-end systems.

1. Introduction

Acoustic models for state-of-the-art speech recognition systems are typically trained on several hundred hours of task specific training data, or more. However, in low resource scenarios often only a few tens of hours of annotated training data are available. In these settings, it is possible to take advantage of transcribed data from other languages to build multilingual acoustic models [1, 2]. These multilingual acoustic models are then used to either extract multilingual bottleneck features for subsequent processing or are directly used as acoustic models after a fine-tuning step on the low resource language [3–15].

Data augmentation is another method for improving the performance of models trained on small amounts of data by extending the dataset with artificially perturbed copies of the training data. Augmentation techniques include adding noise to clean speech [16, 17], vocal tract length perturbation (VTLP) [18, 19], audio speed and tempo perturbation [20], SpecAugment [21], and various combinations of these methods [22]. Another form of data augmentation uses untranscribed data from the same language after semi-supervised labels have been created by processing the data automatically with a trained model. Data created in this fashion is often filtered using various confidence scores before being added to the training data pool [23–25]. More recently, methods to build language-agnostic multilingual ASR systems by transforming the various training languages into one writing system through a many-to-one transliteration transducer have been proposed. With these techniques, it has been shown that multilingual acoustic models become more robust to issues like code-switching [26].

*Work performed while at IBM

In this paper we combine various themes described above: multilingual processing, augmentation using unsupervised data, and transliteration across languages, to develop a transliteration based data augmentation technique to create better acoustic models in low resource settings. The speed-tempo and SpecAugment data augmentation techniques described above can be considered as data augmentation strategies applied at the input audio/feature level. In contrast, the proposed transliteration method in this paper is a data augmentation method at the output label level. With this technique a speech utterance originally transcribed in a particular language can now also be represented in terms of output symbols of other languages which are part of the multilingual training. In other words, a multilingual network is now used as a tool to transcribe audio into various languages. The newly transcribed data is then added back to the training data pool along with the original training data to retrain more accurate multilingual models.

The remainder of the paper is organized as follows. In Section 2, we describe the baseline multilingual acoustic model trained using the Connectionist Temporal Classification (CTC) loss function [27]. Section 3 describes how such a model can be used to process various kinds of data: the original training data used to build the models, untranscribed data from the training languages, and also crosslingual data from a language outside the multilingual training set. We also show how the proposed multilingual models can be ported to new languages. Section 4 describes experiments and results using the multilingual model and proposed transliteration scheme on a set of Babel [28] languages. The paper concludes with a discussion in Section 5.

2. Multilingual CTC Acoustic Models

Similar to acoustic models developed previously, we train a CTC-based multilingual acoustic model with shared recurrent layers on a pooled data set drawn from several languages. The shared recurrent layers are then connected to language specific output layers as shown in Figure 1. In contrast to similar previous models, we do not have a bottleneck layer or any language specific layers except for the the final output layer of each language. Instead of using a common symbol set to cover all the languages, the language specific output layers model the grapheme sets of each particular language separately. With this proposed configuration, we encourage the recurrent layers to first learn a common language representation covering the acoustic space of the various languages used in training. This shared representation, available at the output of the last recurrent layer, is then projected to language graphemic targets via the fully connected language specific layers.

To effectively construct the multilingual CTC model, the model is trained in several stages. In the first step, the model is trained for a fixed number of epochs, over whole utterances to minimize an aggregated CTC loss using stochastic gradient

Copyright © 2020 ISCA

http://dx.doi.org/10.21437/Interspeech.2020-2593

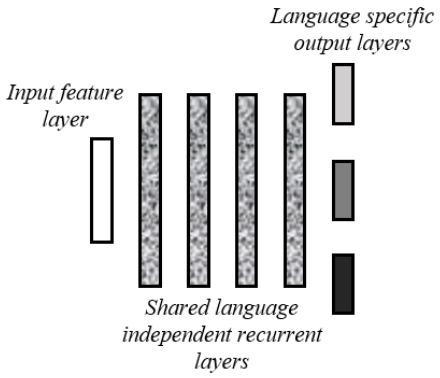

Figure 1: *Schematic of the proposed multilingual AM*

descent. This loss is a sum of the individual CTC losses at the output of each of the language specific layers. In the next training stage, the model is further improved with "soft forgetting" where an additional mean-squared error term ("twin loss") is incorporated to the training loss function [29]. In a third training step, the final multilingual model is constructed by integrating a "guided loss" term with the CTC loss term [30].

For the training procedure outlined above to be effective, it is important to augment the training data appropriately. Data augmentation has traditionally been done at two levels in the proposed multilingual training framework. In an offline setting, multiple copies of the multilingual data are first created using speed and tempo perturbation ($0.9\times$ and $1.1\times$). Additionally, SpecAugment [21] is employed on the fly to further increase the diversity of the training data.

3. Transliterating Multilingual Data

As described earlier, we now use the trained multilingual network to transcribe multilingual data into various languages.

3.1. Transliterating training data

Multilingual data is transliterated by passing acoustic features for each utterance through a trained network and collecting outputs at each of the language specific outputs. When a multilingual network is trained on N languages, for each utterance forward passed through the network, N language specific outputs can be derived. If an utterance being processed is from the set of training languages, the outputs at the corresponding language specific layer will be a transcription of the utterance in terms of the original language, while the outputs at the other $(N-1)$ language specific layers will be transliterations of the utterance into the other training languages. To produce these transliterations, we perform greedy decoding without a language model, removing only symbol repetitions and the blank symbol. This relaxes any language specific constraints during the transliteration process and allows for grapheme sequences not present in a language to be produced when transliterating across languages. Given that for an utterance, $(N-1)$ new label sequences can be generated, this method has the potential to generate an $N \times (N-1)$ fold data augmentation.

3.2. Measuring the quality of transliterated data

Finding a suitable measure to select reliable data from a potentially large pool of transliterated data is crucial to the use of transliterated data for training. Unlike automatic transcriptions, whose quality can be measured against reference transcripts, there is no straightforward measure of transliteration quality, given that the output labels for an utterance are now in a different language. In our training framework, each utterance in the pool of multilingual data has a transcript only in the original spoken language and not in the other languages used to train the multilingual network. It is therefore necessary to develop a measure that is indicative of whether a transliterated utterance is being well represented with the symbol set of the language into which it has been transliterated, without any reference transcript for the utterance in the transliterated language.

We propose two metrics that can be used to measure the usefulness of a transliterated outputs:

1. Symbol count of transliterated output (**SC**): This measure is a simple count of the number of symbols in the transliterated output of an utterance. When an utterance is being transliterated into a language different from the spoken language, it is possible that the acoustic signal cannot be very well represented with symbols from the target language. We hypothesize that for utterances which have been effectively transliterated, more symbols will likely be produced by the network, compared to utterances which are poorly transcribed because of language mismatch.
2. Ratio of symbol count in a transliterated language to symbol count of the reference transcript (**SR**): This measure compares the number of symbols in the transliterated output to the number of symbols in the utterance's original reference transcript. Although it is unlikely that every symbol in the original spoken language can be mapped to a symbol in a transliterated language, a higher ratio value can be indicative of a better transliterated output.

An immediate application of creating and selecting transliterated output for data augmentation is with transliterations of the training data itself. In this case, both the measures described above are suitable for use. However, as will be detailed in the following sections, it is possible to also transliterate untranscribed audio from the training languages or even data from languages that are not part of training. In such cases, since a reference transcript will not be available, the ratio based metric cannot be used. The symbol count metric is however still useful in those settings. Both of the metrics described above do not use traditional posterior word or symbol confidences for the reason that CTC posteriors are often peaky, and hence not reliable confidence measures for use as a selection metric.

3.3. Transliterating untranscribed data

When multilingual training data is being transliterated, the network is in essence reprocessing data that it has already seen, but now with different output labels. With the proposed data augmentation, we hypothesize that the network is probably now able to learn better associations between languages, as a single data set is being presented in multiple languages during training. In a second application, the transliteration scheme can be used to process untranscribed data, hence introducing novel data not yet seen by the network into training. Untranscribed data belonging to the languages used to train the network can be forward passed through the network. The output at the output layer corresponding to the spoken language is a semi-supervised transcription of the utterance. The other network outputs are transliterations of an utterance into the other training languages. If untranscribed data sets are available in all

the N training languages, this data augmentation scheme can produce an $N \times N$ fold data augmentation.

Given that an actual reference is not available, the SR metric described earlier cannot be readily used to select novel data. It can however be modified to be the ratio of the output symbol count in a transliterated language to the semi-supervised output symbol count in the source language, instead of the reference symbol count in the source language. The SC metric on the other hand can be used directly.

3.4. Dealing with crosslingual data

In both the data settings described above, only data from the languages used to train the multilingual network have been used. It is however also possible to process data from completely disjoint languages and use the transliterated data for augmentation. In this setting utterances from outside the N languages used to train the multilingual network are used. Before the data can be added to the training data pool, the data is transliterated into the N training languages. If data from M languages is being used, with this scheme it is possible to create an $M \times N$ fold data augmentation. For data selection, given that neither reference nor semi-supervised transcripts are available, only the symbol count (SC) based metric can be used.

3.5. Extending to new languages

Multilingual networks have been shown to be useful as pretrained acoustic models in low-resource settings. A low resource acoustic model is constructed with layers initialized from the multilingual model and a new output layer corresponding to the language. The model is then further fine tuned with data just from the low resource language. In the current framework, to port the proposed multilingual model to a new language and allow for data augmentation using transliteration, an extended multilingual model with the new language as an additional data set is instead trained.

Once a low resource language is added to an existing pool of N languages, a new multilingual model is trained with $(N+1)$ language specific layers as described earlier. Given that the new language is part of the new model, the multilingual data pool can now be transliterated to the new language as well. An improved multilingual model is then trained with the additional transliterated data. This procedure can be extended to include multiple new languages.

4. Experiments and Results

To demonstrate the efficacy of our proposed transliteration based data augmentation scheme we build CTC based end-to-end ASR systems on 4 low resource languages. Resources for these languages — Mongolian (401), Javanese (402), Dholuo (403) and Georgian (404) — were created as part of the IARPA Babel program [28]. The full language packs of each of these languages contain about $\sim$40 hours of transcribed data. An additional $\sim$4 hours of data is used as a heldout data set. We present results on $\sim$10 hour test sets for each of these languages.

Four additional copies of each training data set are first created using speed and tempo perturbation, producing $\sim$210 hours of data. LSTM based acoustic models are then trained on these augmented data sets. Each of the monolingual acoustic models has 4 bidirectional LSTM layers with 512 units per direction and a final fully connected layer corresponding to a language specific output grapheme symbol set. We use 63, 38, 49, and 40 units for 401, 402, 403, and 404 respectively. After 20 epochs of SGD based training to minimize the CTC loss, the models are trained for 20 more epochs with an additional soft-forgetting loss. The final models are created after 20 more epochs of guided training. SpecAugment is also applied to the input log-mel spectral features to provide on-the-fly data augmentation. We use hyperparameter settings from [29, 30] in these training steps. The language models (LM) used in our experiments are all Kneser-Ney smoothed bigram models.

Table 1: *Model performance (WER%) with transliterated training data*

Condition	**401**	**402**	**403**	**404**	**Hrs.**
[A1] Mono	52.0	56.2	41.8	44.4	210
[B1] Mult	48.4	54.3	40.7	44.0	850
[C1] Mult+TL-ALL	47.9	54.0	40.5	43.7	1350
[D1] Mult+TL-FL1	46.5	52.8	39.3	42.6	1320
[E1] Mult+TL-FL2	46.6	52.6	39.1	42.6	1050
[F1] Mult+TL-FL3	46.0	52.7	39.1	42.4	1230

Experiment [A1] in Table 1 shows the WER results we obtain for each language when we train separate monolingual models. We then pool the different language sets and train a single multilingual model with shared LSTM layers and language specific output layers. Given that more data is available to train the multilingual model, we increase the number of LSTM layers to 6 keeping all other training parameters the same. Experiment [B1] shows the WER results with the multilingual model. The model already shows significant improvement over the monolingual systems. We now use the trained multilingual system to transliterate the training data. The 401 training data, for example, is now forward passed through the network and outputs at the 402, 403 and 404 language specific output layers are used as transliterations of the 401 data into those languages. We limit the data we process to just the original $\sim$40 hours of data in each language. This results in $\sim$480 (40$\times$4$\times$3) hours of transliterated data. Experiment [C1] shows the result of adding all the transliterated data to the training data pool. Although there is a slight improvement we hypothesize that there could be *noisy* data that needs to be filtered out.

In the next set of experiments we explore the use of the two metrics proposed to measure the usefulness of transliterated outputs. Experiment [D1] shows the result of using the symbol count (**SC**) metric. With this metric we filter out all transliterated utterances with less than 3 symbols in the transliterated output. We also evaluate the usefulness of the symbol ratio (**SR**) metric. For experiments [E1] and [F1], we sort all utterances by their symbol ratios, i.e. the ratio of the symbol count in a transliterated language to the symbol count of the utterance's reference in the original spoken language. As described earlier, a high ratio value suggests that more symbols were transliterated, suggesting that the symbol set of the language into which the utterance was transliterated can adequately represent the utterance. In experiment [E1], we select 50 hours of transliterated data with the highest SR scores, for each language. For experiment [F1] we increase the amount to 100 hours per language. Both these experiments show the usefulness of the proposed metric as we now obtain more gains by selecting the appropriate data. The final multilingual system [F1] with the transliteration based augmentation and filtering, has a relative improvement of up to 5% over the baseline multilingual system [B1] and 12% over the monolingual system performance in [A1]. All these

gains are realized without introducing novel training data and on top of other input level data augmentation strategies.

Table 2: *Model performance (WER%) with untranscribed training data after transliteration*

Condition	401	402	403	404	Hrs.
[A2] Mult+ST	46.9	53.5	40.1	43.9	1000
[B2] Mult+ST+TL	45.8	52.3	39.1	42.6	1460

As described earlier, the proposed transliteration method can not only be used to transform multilingual training data, but can also be applied on untranscribed data. For each of the language packs, 40 hours of additional untranscribed audio is available. This data is forward passed through the baseline multilingual network developed for experiment [B1]. Because reference transcripts are not available, we select transliterated outputs using the SC metric, filtering out all transliterated utterances with fewer than 3 symbols in the transliterated output. We consider transcripts collected at the network outputs corresponding to the same underlying spoken language as semi-supervised transcripts. In a first set of experiments we add $\sim$150 hours of semi-supervised data (40$\times$4) into the data pool to train a new multilingual model. Experiment [A2] in Table 2 shows that adding novel data with semi-supervised labels is indeed useful as the new multilingual network with the additional data performs better than the baseline multilingual system [B1]. We then add all the transliterated outputs available at other output layers to the training pool. Experiment [B2] shows that using the transliterated outputs along with the semi-supervised outputs further improves the performance of the new multilingual system by up to 5% relative over the baseline system.

Table 3: *Model performances (WER%) with untranscribed cross-lingual data after transliteration*

Condition	401	402	403	404	Hrs.
[A3] Mult+CRS1	46.8	53.5	39.7	43.2	1390
[B3] Mult+CRS2	47.8	54.1	40.7	44.0	1430

In a next set of experiments, we investigate if multilingual data from outside the set of languages employed to train the multilingual network can be used. We use two different sets of Babel languages, the first set of cross-lingual languages (CSR1) include 4 languages from the OP1 phase of the program: Cebuano (301), Kazakh (302), Telugu (303), and Lithuanian (304). The second set of languages (CRS2) includes 4 languages from the OP2 phase: Pashto (104), Paraguayan Guarani (305), Igbo (306), and Amharic (307). We use $\sim$40 hours of data from each of the languages for these experiments and forward pass both language sets through the multilingual baseline used in [B1]. This produces $\sim$640 hours (40$\times$4$\times$4) hours of transliterated data. After SC filtering we use $\sim$540 and $\sim$580 additional hours of transliterated data to train two separate multilingual networks. Experiments [A3] and [B3] in Table 3 show the results of these cross-lingual experiments. In both cases we observe improvements of up to 3% relative over the baseline multilingual system, showing that the proposed transliteration of multilingual data is also useful for languages outside the training set of the base multilingual network.

In our final set of experiments we investigate how the proposed multilingual model and data augmentation scheme can be extended for use with a new language. Similar to the data settings used for the Babel languages, we select $\sim$40 hours of transcribed Italian from an internal data collection. As with earlier experiments, the data set is then expanded with speed and tempo augmentation to $\sim$200 hours. In experiments [A4] and [B4], we train two different monolingual systems: one using randomly initialized weights and the second using LSTM weights from the multilingual model trained in experiment [B1]. For the latter experiment, the final language specific layers of the multilingual model are replaced with a randomly initialized language specific layer with 35 outputs corresponding to the Italian symbol set. Although both the systems are trained with the same amount of data, the monolingual system in [B4], which is initialized from a trained multilingual system, performs better. Instead of training a separate monolingual system, in the next experiment we integrate an additional Italian output layer to the original multilingual network with 4 Babel languages and train a new multilingual system. This system benefits from the multilingual training and enjoys considerably better performance as can be seen in the results of system [C4]. Similar to previous experiments, we now transliterate the training pool of 5 languages. The transliterated data is processed with the SC metric to select utterances with SC greater than 3. An additional $\sim$750 hours of transformed training data is added to the data pool and used to retrain a new multilingual system in [D4]. The proposed data augmentation scheme allows the network to improve performance by 14% relative over the previous multilingual system. Compared to the original Italian monolingual system, this is a very significant 24% relative improvement, without the need for any additional transcribed Italian data.

Table 4: *Model performance (WER%) after porting the model to include a new language*

Training condition	IT	Hrs.
[A4] IT-Mono.(rand)	32.8	200
[B4] IT-Mono.(mult)	31.8	200
[C4] IT-Mult	28.9	1050
[D4] IT-Mult+TL	25.0	1800

5. Conclusion

In this paper we have proposed a new data augmentation technique useful for building multilingual acoustic models. Coupled with two metrics for data selection, we have demonstrated how transliterated data generated using these multilingual models can be used to further improve performance. The efficacy of our proposed method has been shown on several kinds of data sets: the original training data used to build the models, untranscribed data from the training languages, and also crosslingual data from outside the set of original training languages. We have also shown how the proposed multilingual models can be ported to new languages and how improved models with significantly lower word error rates can be constructed.

6. References

[1] A. Waibel, H. Soltau, T. Schultz, T. Schaaf, and F. Metze, "Multilingual speech recognition," in *Verbmobil: Foundations of Speech-to-Speech Translation*. Springer, 2000, pp. 33–45.

[2] H. Lin, L. Deng, D. Yu, Y.-f. Gong, A. Acero, and C.-H. Lee, "A study on multilingual acoustic modeling for large vocabulary asr,"

in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, pp. 4333–4336.

[3] S. Thomas, S. Ganapathy, and H. Hermansky, "Cross-lingual and Multi-stream Posterior Features For Low Resource LVCSR Systems," in *ISCA Interspeech*, 2010.

[4] D. Imseng, H. Bourlard, and P. Garner, "Using KL-divergence And Multilingual Information To Improve ASR For Under-resourced Languages," in *IEEE ICASSP*, 2012.

[5] Z. Tuske, R. Schluter, and H. Ney, "Multilingual Hierarchical MRASTA Features For ASR," in *ISCA Interspeech*, 2013.

[6] Huang, J. and Li, J. and Yu, D. and Deng, L. and Gong, Y., "Cross-language Knowledge Transfer Using Multilingual Deep Neural Network With Shared Hidden Layers," in *IEEE ICASSP*, 2013.

[7] G. Heigold, V. Vanhoucke, A. Senior, P. Nguyen, M. Ranzato, M. Devin, and J. Dean, "Multilingual Acoustic Models Using Distributed Deep Neural Networks," in *IEEE ICASSP*, 2013.

[8] A. Ghoshal, P. Swietojanski, and S. Renals, "Multilingual Training Of Deep Neural Networks," in *IEEE ICASSP*, 2013.

[9] S. Thomas, M. L. Seltzer, K. Church, and H. Hermansky, "Deep Neural Network Features And Semi-supervised Training For Low Resource Speech Recognition," in *IEEE ICASSP*, 2013.

[10] F. Grézl and M. Karafiát, "Combination of Multilingual And Semi-Supervised Training For Under-Resourced Languages," in *ISCA Interspeech*, 2014.

[11] N. Thang, B. Wojtek, F. Metze, and T. Schultz, "Initialization Schemes For Multilayer Perceptron Training and Their Impact On ASR Performance Using Multilingual Data," in *ISCA Interspeech*, 2012.

[12] Y. Qian and J. Liu, "Cross-Lingual and Ensemble MLPs - Strategies for Low-Resource Speech Recognition," in *ISCA Interspeech*, 2012.

[13] A. Ragni, M.J.F. Gales and K.M. Knill, "A Language Space Representation For Speech Recognition," in *IEEE ICASSP*, 2015.

[14] Y. Zhang, E. Chuangsuwanich, J. Glass, "Language ID-based Training Of Multilingual Stacked Bottleneck Features," in *ISCA Interspeech*, 2014.

[15] J. Cui, B. Kingsbury, B. Ramabhadran, A. Sethy, K. Audhkhasi, X. Cui, E. Kislal, L. Mangu, M. Nussbaum-Thom, M. Picheny, Z. Tuske, P. Golik, R. Schluter, H. Ney, M.J.F. Gales, K.M. Knill, A. Ragni, H. Wang and P. Woodland, "Multilingual Representations For Low Resource Speech Recognition And Keyword Search," in *IEEE ASRU*, 2015.

[16] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[17] M. J. Gales, A. Ragni, H. AlDamarki, and C. Gautier, "Support vector machines for noise robust asr," in *2009 IEEE Workshop on Automatic Speech Recognition & Understanding*. IEEE, 2009, pp. 205–210.

[18] N. Jaitly and G. E. Hinton, "Vocal tract length perturbation (vtlp) improves speech recognition," in *Proc. ICML Workshop on Deep Learning for Audio, Speech and Language*, vol. 117, 2013.

[19] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep neural network acoustic modeling," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 9, pp. 1469–1477, 2015.

[20] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[21] 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," *arXiv preprint arXiv:1904.08779*, 2019.

[22] A. Ragni, K. Knill, S. P. Rath, and M. Gales, "Data augmentation for low resource languages," in *Proc. Interspeech*, 2014.

[23] G. Zavaliagkos and T. Colthurst, "Utilizing untranscribed training data to improve performance," in *DARPA Broadcast News Transcription and Understanding Workshop, Landsdowne*. Citeseer, 1998.

[24] G. Evermann and P. C. Woodland, "Large vocabulary decoding and confidence estimation using word posterior probabilities," in *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 00CH37100)*, vol. 3. IEEE, 2000, pp. 1655–1658.

[25] L. Lamel, J.-L. Gauvain, and G. Adda, "Lightly supervised and unsupervised acoustic model training," *Computer Speech & Language*, vol. 16, no. 1, pp. 115–129, 2002.

[26] A. Datta, B. Ramabhadran, J. Emond, A. Kannan, and B. Roark, "Language-agnostic multilingual modeling," *ArXiv*, vol. abs/2004.09571, 2020.

[27] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[28] "The IARPA Babel Program," http://www.iarpa.gov/index.php/research-programs/babel, [Online; accessed 2020-05-05].

[29] K. Audhkhasi, G. Saon, Z. Tüske, B. Kingsbury, and M. Picheny, "Forget a Bit to Learn Better: Soft Forgetting for CTC-Based Automatic Speech Recognition," *Proc. Interspeech 2019*, pp. 2618–2622, 2019.

[30] G. Kurata and K. Audhkhasi, "Guiding CTC Posterior Spike Timings for Improved Posterior Fusion and Knowledge Distillation," *arXiv preprint arXiv:1904.08311*, 2019.