Pretraining by Backtranslation for End-to-end ASR in Low-Resource Settings

Matthew Wiesner[†], *Adithya Renduchintala*[†], *Shinji Watanabe*[†],
Chunxi Liu[†], *Najim Dehak*[†], *Sanjeev Khudanpur*[†‡]

[†]Center for Language and Speech Processing, The Johns Hopkins University, USA
[‡]Human Language Technology Center of Excellence, The Johns Hopkins University, USA
{wiesner,adi.r,shinjiw,cliu77,ndehak3,khudanpur}@jhu.edu

Abstract

We explore training attention-based encoder-decoder ASR in low-resource settings. These models perform poorly when trained on small amounts of transcribed speech, in part because they depend on having sufficient target-side text to train the attention and decoder networks. In this paper we address this shortcoming by pretraining our network parameters using only unpaired target-side text and transcribed speech from other languages. We pretrain the decoder and attention networks by using an *augmenting* encoder that shares the attention and decoder networks and encodes synthetically "back-translated" symbolic input. To pretrain the encoder we explore two alternatives: we either feed the output of the augmenting encoder directly to the primary encoder, or we simply train on transcribed speech from other languages. We compare our approach to shallow fusion with an external language model. Across 3 test languages, pretraining on back-translated symbolic input resulted in a 20% average relative improvement in character error rate (CER) over the same technique without pretraining, which in turn outperforms shallow fusion with an external language model language. Using transcribed speech from nearby languages gives a further 20-30% relative reduction in character error rate.

Index Terms: Multi-modal data augmentation, pretraining, multilingual ASR, encoder-decoder, low-resource

1. Introduction

Attention-based encoder-decoder networks have achieved state-of-the art performance in ASR when trained on over 12k hours of transcribed speech [1], but their performance lags behind conventional systems in more moderate resource conditions and has only just begun to be studied in low-resource conditions [2, 3]. One way to improve ASR performance without access to more transcribed speech is to leverage linguistic resources from other languages and modalities. Bolstering the decoder with a language model (LM) trained on supplemental text data is one such method that improves end-to-end ASR performance [4, 5]; however, more significant gains can be obtained by training on additional synthetically perturbed speech [6, 7, 8], or by *multilingual training*, which augments the training data with transcribed speech from other languages [9, 10, 11, 12, 13, 14]

Using an LM in decoding is appealing as it requires only text data, but provides only modest improvements in performance. And while multilingual training often provides more significant improvements in performance, this approach also requires additional transcribed speech, preferably from similar languages [11, 15]. Our aim is to achieve performance improvements similar to multilingual training, but obtained solely from text data.

As a starting point we consider multi-modal data augmentation (MMDA): a data augmentation scheme for encoder-decoder based ASR which only requires text data [16] (see figure 1. (a)). The approach, inspired by "back-translation" in neural machine translation (NMT) [17], involves using an additional *augmenting* encoder (in addition to the traditional acoustic encoder), which accepts a sequence of features derived from text as input and learns to predict the original text. Other work uses a text-to-speech (TTS) system to generate the augmenting features, however, training a reasonable TTS requires more single speaker data than we have available in many low-resource situations [18, 19].

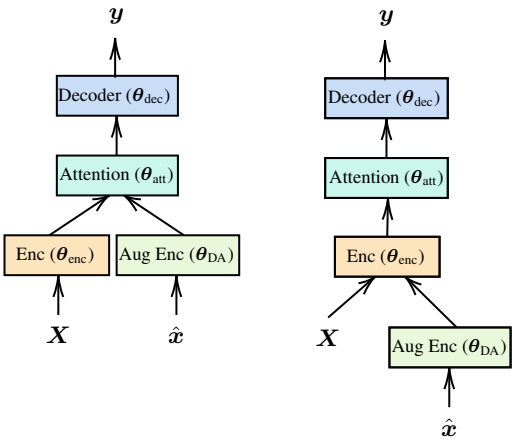
We extend MMDA to work in low resource contexts by again borrowing from techniques in NMT. We adapt a technique proposed in [20], which uses an unsupervised language modeling task on both the source and target languages to pretrain the encoder and decoder model parameters respectively in order to reduce over-fitting and improve generalization. This approach uses the insight that in NMT both the encoder and decoder act as language models. In ASR, however, only the decoder clearly exhibits this behavior. Unlike in NMT, however, ASR exhibits monotonic attention, which is relatively easy to initialize. For this reason we instead pretrain the decoder and attention parameters using synthetically "back-translated" training examples with MMDA.

To pretrain the encoder we propose a modified architecture that feeds the output of the augmenting encoder to the acoustic encoder. The augmenting data can then be viewed as *pseudo-speech* from some language that we add to our training data. We refer to this as pseudo-speech data augmentation (PSDA) as the augmenting encoder is implicitly tasked with learning representations of the augmenting data that resemble the original acoustic features. We also compare pretraining the network with PSDA and synthetic inputs to training instead with actual speech from other languages in order to study the usefulness of the synthetic data in pretraining the encoder.

2. Related Work

The most similar work is [16], which uses categorical data in addition to transcribed speech when training the attention and decoder networks in end-to-end ASR. Related work on how to best integrate language models into end-to-end ASR includes deep and shallow fusion [21], cold fusion [22], or transfer fusion [23]. While [20] proposes pretraining both the encoder and decoder on unpaired monolingual data using an *unsupervised* language modeling objective, we propose using back-translated data as in [16], to pretrain using a *supervised* objective.

This work was supported by DARPA LORELEI Grant No HR0011-15-2-0024 and partially carried out during the 2018 Jelinek Memorial Summer Workshop on Speech and Language Technologies, supported by gifts from Microsoft, Amazon, Google, Facebook, and MERL/Mitsubishi Electric.



(a) *MMDA Architecture* (b) *PSDA Architecture*

Figure 1: *MMDA and PSDA. $\boldsymbol{X}$ is the original speech, $\hat{\boldsymbol{x}}$ is the text-based augmenting input, and $\boldsymbol{y}$ is an output character sequence. $\boldsymbol{\theta}_{enc}$, $\boldsymbol{\theta}_{att}$, $\boldsymbol{\theta}_{dec}$, $\boldsymbol{\theta}_{DA}$, are the parameters corresponding to the encoder, attention, decoder, and data-augmenting encoder respectively*

Work in multilingual (pre)training has the same objective as [20] of increasing the generalizability of the encoder. [11] and [14] have both investigated multilingual training of encoder-decoder architectures. [15] showed that pretraining the encoder using data from other, preferably nearby, languages can result in large performance gains in encoder-decoder ASR.

In our work we pretrain both the encoder and decoder, as in [20]. For the decoder we pretrain only on augmenting data using MMDA [16] instead of a language modeling objective. For the encoder we explore training using speech in other languages as well as a novel architecture, PSDA, to enable joint pretraining of the encoder and decoder on back-translated text.

3. Data Augmenting Architecture

Our architecture follows the encoder-decoder model which maximizes the log-likelihood:

$$\mathcal{L}(\boldsymbol{\theta}) = \log P(\boldsymbol{y} \mid \boldsymbol{X}\ ; \boldsymbol{\theta}_{\text{enc}}, \boldsymbol{\theta}_{\text{att}}, \boldsymbol{\theta}_{\text{dec}}) \tag{1}$$

$\boldsymbol{y}$ denotes the desired output character sequence and $\boldsymbol{X} \in \mathbb{R}^{L \times D}$ a tensor of speech frames of length L and feature dimension D. We denote the entire set of network parameters by $\boldsymbol{\theta}$, which is composed of acoustic encoder parameters $\boldsymbol{\theta}_{\text{enc}}$, attention mechanism parameters $\boldsymbol{\theta}_{\text{att}}$ and decoder parameters $\boldsymbol{\theta}_{\text{dec}}$. The encoder consists of a projection-biLSTM with a pyramidal structure for the acoustic encoder [24], the decoder is a single-layer LSTM and location-aware attention completes the entire end-to-end network.

3.1. Multi-Modal Data Augmentation

The MMDA technique (fig. 1a) transforms the objective into a multi-task objective:

$$\mathcal{L}(\boldsymbol{\theta}) = \begin{cases} \log P(\boldsymbol{y} \mid \boldsymbol{X}\ ; \boldsymbol{\theta}_{\text{enc}}, \boldsymbol{\theta}_{\text{att}}, \boldsymbol{\theta}_{\text{dec}}) \text{ speech} \\ \log P(\boldsymbol{y} \mid \hat{\boldsymbol{x}}\ ; \boldsymbol{\theta}_{\text{DA}}, \boldsymbol{\theta}_{\text{att}}, \boldsymbol{\theta}_{\text{dec}}) \text{ text-based} \end{cases} \tag{2}$$

When the inputs are acoustic features, $\boldsymbol{X}$, MMDA uses the standard encoder-decoder network to maximize the original ASR objective and the primary task. When the inputs are text-based features, $\hat{\boldsymbol{x}}$, MMDA uses a *data-augmenting encoder* instead of the acoustic encoder and maximizes the probability of the output sequence paired with the text-based representation (secondary task). In Eq 2, $\boldsymbol{\theta}_{\text{DA}}$ denotes the parameters of the data-augmenting encoder which is composed of an embeddings layer and a single-layer projection-biLSTM.

3.2. Pseudo-Speech Data Augmentation

We propose a variation of MMDA which changes the architecture during the secondary task (fig. 1b). In this setup, we cascade the data-augmenting and acoustic encoders and force the data-augmenting encoder's output to match the dimensionality of acoustic frames (which are the input in the primary task). Thus, in PSDA the entire encoder-decoder network is part of the computation graph in both tasks.

$$\mathcal{L}(\boldsymbol{\theta}) = \begin{cases} \log P(\boldsymbol{y} \mid \boldsymbol{X}\ ; \boldsymbol{\theta}_{\text{enc}}, \boldsymbol{\theta}_{\text{att}}, \boldsymbol{\theta}_{\text{dec}}) \text{ speech} \\ \log P(\boldsymbol{y} \mid \hat{\boldsymbol{x}}\ ; \boldsymbol{\theta}_{\text{DA}}, \boldsymbol{\theta}_{\text{enc}}, \boldsymbol{\theta}_{\text{att}}, \boldsymbol{\theta}_{\text{dec}}) \text{ text-based} \end{cases} \tag{3}$$

PSDA can be viewed as a proxy multilingual training method, where the pseudo-speech generated by the data-augmenting encoder (which is fed into the acoustic encoder) is a cheap approximation of real acoustic features of some new, but related language. We use the same structure for the data-augmenting encoder as in the MMDA case.

3.3. Multi-task Training & Pretraining

We pretrain our encoder and decoder with augmenting data using both MMDA and PSDA architectures and show that it significantly improves ASR performance. It is important to occasionally update the network using augmenting data after pretraining in order to prevent catastrophic forgetting [25].

In [16] we proposed training the MMDA network by alternating between audio-data and augmenting-data minibatches. We now allow for more flexibility by using a hyper-parameter $\rho \in (0, 1)$ that decides if the model should be trained on speech data or text-based data. In this way we can tune the number of augmenting updates needed to prevent catastrophic forgetting.

4. Shallow Fusion

We compare MMDA, PSDA, and our pretrained variants to a shallow fusion baseline. Shallow fusion [21], is a simple, effective and commonly used technique for external language model integration in sequence to sequence learning for ASR [26]. In shallow fusion, a list of partial hypothesis and corresponding scores is produced by the ASR decoder. Each partial hypothesis is then also scored by an external language model. A composite score for the partial hypothesis is given by

$$\text{score}(\boldsymbol{y}) = \log P_{ASR}(\boldsymbol{y}|\boldsymbol{x}) + \lambda \log P_{LM}(\boldsymbol{y}). \tag{4}$$

$\log P_{ASR}(\boldsymbol{y}|\boldsymbol{x})$ is the ASR score for a hypothesis sequence $\boldsymbol{y}$ given an input utterance $\boldsymbol{x}$, $\log P_{LM}(\boldsymbol{y})$ is the corresponding language model score, and λ is a tunable parameter. The list of hypotheses is reordered prior to prediction of the subsequent output and only the top scoring hypotheses are retained.

5. Experiments

We conducted experiments on 4 languages from the Voxforge corpus: Catalan, Portuguese, Italian, and French. We chose

these data sets because they have small amounts of relatively clean training data (0.5-30h) and are closely related to Spanish which we used in multilingual training (see 5.2). This allows us to study the effect of small training data on end-to-end ASR in isolation, without worrying about confounding factors such as language relatedness or the noisiness of the training data.

For Catalan, Portuguese, and Italian, we created 5 baseline systems: 1. A baseline monolingual model (Monolingual) 2. A monolingual model with decoded using shallow fusion (LM) 3. The same baseline model trained as in [16] using an augmenting encoder and augmenting data scraped from the web (MMDA). 4. A model that was trained on transcribed speech from other languages in addition to the monolingual data (ML). 5. The multilingual model decoded with shallow fusion (ML+LM). All of the augmenting data was used to train the RNNLM for each language to enable a fair comparison between shallow fusion and MMDA.

5.1. Monolingual Systems

We trained monolingual systems for Catalan, Portuguese, and Italian. The training, development and evaluation sets are constructed by randomly sampling 80%, 10%, and 10% of the data for each set respectively, ensuring that no prompt in the development or test sets is duplicated in the training set. The Catalan and Portuguese systems were trained on the entire 30 min and 3 hour extracted training sets respectively. For Italian we trained only on a 4 hour subset of the full 16 hour training set in order to more closely mimic the training conditions of the two other languages. All systems were trained using ESPnet [27]. We trained encoder-decoder networks as described in 3, but without the augmenting encoder. We used the same configurations as in [11], except for we used 4 encoder layers for all experiments.

5.2. Multilingual Systems

For Catalan and Portuguese we augmented the training data with all 30h of the Hub-4 Spanish Broadcast news corpus training set and all 16h of the Italian Voxforge training set. For Italian we only added the Hub-4 Spanish to training. All systems were trained using the same network configurations and training parameters as the monolingual systems. We followed [14, 11] and use as output symbols the union of all graphemes seen in training such that the network was capable of outputting any of the languages seen in training.

5.3. MMDA & PSDA

We trained monolingual MMDA and PSDA systems as well as systems with pretraining (MMDA+P, PSDA+P) as described in section 3 using the same data splits as described in section 5.1. We also trained multilingual (ML) MMDA and PSDA systems which we compared to an ML baseline with RNNLM shallow fusion (ML+LM).

Augmenting Data: The augmenting data were generated by first scraping Wikipedia for text in the language of interest. We then filtered out tokens with characters that did not appear in the audio training data as well as long and short sentences, resulting in 2.2, 3.2, 3.8, and 4.2 million training examples for Catalan, Portuguese, Italian, and French respectively. As in [16] we converted this text into sequences of phonemes which was shown to give better performance than simply using only the graphemes. First, we created pronunciation lexicons for each language, by scraping Wiktionary for pronunciations of all words seen in the augmenting text data. For each language we then trained a grapheme-to-phoneme transducer using `Phonetisaurus` [28] on the corresponding scraped lexicons, which we then used to recover pronunciations for all words in the augmenting data absent from the lexicon.

[16] also found that modeling phoneme duration was important, and use the frame level phoneme alignments from TIMIT to learn this model. Transferring the duration model to a new language required manually mapping the new phoneme

Table 1: *Summary of **monolingual** experiments. We see that our proposed pretraining (indicated with **+P**) improves performance dramatically. Both MMDA+P and PSDA+P show strong and consistent improvement over Monolingual, LM and MMDA baselines, reducing CER by 20% to 26%.*

Task	CA (0.5h) dev, eval	PT (3h) dev, eval	IT (4h) dev, eval
Monolingual	85.2, 82.3	76.9, 80.1	31.2, 31.4
LM	79.7, 76.9	77.6, 79.9	32.1, 32.1
MMDA	79.1, 76.5	73.7, 72.3	27.9, 28.2
PSDA	86.3, 81.4	80.0, 76.9	29.2, 29.4
MMDA + P	73.8, 75.3	55.4, 56.1	**23.9, 24.1**
PSDA + P	**71.2, 72.2**	**47.4, 50.2**	25.0, 26.0

Table 2: *Summary of **multilingual** experiments (indicated with ML). MMDA+P and PSDA+P yield performance gains beyond multilingual training and RNNLM fusion for both PT and IT.*

Task	CA (0.5h) dev, eval	PT (3h) dev, eval	IT (4h) dev, eval
ML	33.1, 37.2	34.5, 38.4	20.1, 21.0
ML+LM	**31.1**, 36.4	33.3, 37.7	18.7, 19.6
MMDA+P+ML+LM	34.2, **36.2**	**32.4**, 35.9	17.2, 17.8
PSDA+P+ML+LM	34.9, 38.7	33.8, **35.3**	**17.1, 17.6**

Table 3: *Example VoxForge Italian sentence (criptogenetico-criptogenetico-20081224-jmd-it-0801) decoded using 4 ASR systems trained on 4h of speech. LM is the baseline system decoding using langauge model shallow fusion. PSDA+P refers to PSDA with pretraining. ML+LM is multilingual training and language model shallow fusion. COMB uses all techniques above combined. The development set word-error-rate (WER) of each model is shown. Results on the corresponding evaluation sets are always 2-3% worse. Word errors are **bold**.*

System	WER	**Sentence**
LM	74.9	QUESTE **SEI VERSO NON** NOSTRE **TORNATO** I QUINDI **NOSTRI PER SI ERANO DI BANDERE A** TUTTI
PSDA+P	70.1	QUESTE **SE IL VESO NON MOSTRE** TORNATE QUINDI **E** VOSTRI **PENSI E** NOI **DI** MANGEREMO **A** TUTTI
ML+LM	60.7	QUESTE SELVE **SON A** NOSTRE TORNATE **QUINDIE NOSTRI** PAESI **E** NOI **DI** MANGEREMO TUTTI
COMB	56.2	QUESTE SELVE SONO NOSTRE TORNATE QUINDI **E** VOSTRI **PESI E** NOI **DI** MANGEREMO TUTTI
Reference		QUESTE SELVE SONO NOSTRE TORNATE QUINDI AI VOSTRI PAESI O NOI VI MANGEREMO TUTTI

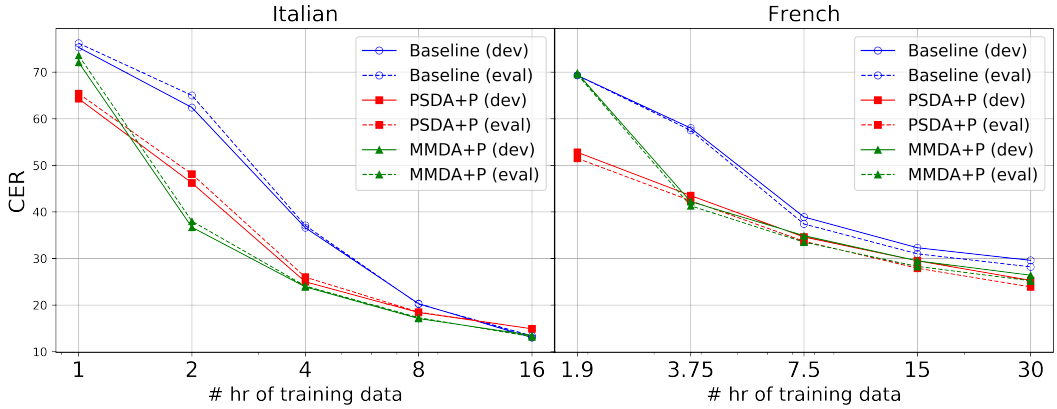


Figure 2: *CER of the baseline system, MMDA+P, and PSDA+P on the Voxforge Italian and French Corpora across varying training set sizes*

inventory to TIMIT phonemes. We instead model phoneme duration with a shared Gaussian distribution across all phonemes, whose mean is the average ratio of input frames to output symbols in the audio training data. We then repeated each phoneme by a duration sampled from this distribution. The variance ensured that any duplicate or similar sentences resulted in unique training pairs. This technique eliminated the need for frame-level phoneme alignments, and still resulted in considerable improvement.

Hyper-parameter Optimization: We randomly sampled hyper-parameters from the possible configurations of # pretraining batches and augmenting ratio: $\{2000, 5000, 8000\} \times \{0.1, 0.2, 0.5\}$. We selected the parameters that performed best on the development set for each experiment. For the monolingual French and Italian experiments, however, we simply used 2000 pretraining batches and 0.1 and 0.5 augmenting ratios for PSDA and MMDA respectively as we found these values worked well for Portuguese and Catalan.

6. Results

Tables 1, 2 show the performance (CER) of data augmentation across different languages with similar data sizes. We note that vanilla MMDA outperformed shallow fusion (LM) in all 3 languages. The pretrained variants resulted in a further 20% relative improvement. PSDA+P outperformed pretrained MMDA+P for both Catalan and Portuguese, which have extremely limited training sets, but MMDA+P was the best system on Italian. This corroborated our intuition that PSDA should help more when fewer data are available, as it allows for encoder pretraining, though its utility may only be in extremely data constrained situations.

We also studied data augmentation on a single language across various amounts of training data. To this end we created 4 smaller Italian and French training sets by successively randomly removing half of the training examples from the original 16 and 30 hour training sets respectively. We then trained the baseline monolingual, MMDA+P, and PSDA+P systems on each resulting dataset using the same network and training parameters as before. We used the same hyperparameters as in the monolingual 4h Italian experiments. Both MMDA+P and PSDA+P performed similarly to each other across all training data sizes, except when training on just a few hours of speech (see fig.2). They both outperformed the baseline by a wide margin, with greater improvements when data were more scarce.

Finally, comparing the use of pseudo-speech features (PSDA+P) to multilingual training we see that PSDA+P gives about 50% of the improvement of multilingual training on extremely close languages. Furthermore, the {MMDA,PSDA}+P+ML+LM systems were our best performing on the evaluation set in every language tested. Using these techniques together on only 1/4 of the full Italian training data gives performance similar to the baseline model trained on the full data set.

Since pretrained PSDA (PSDA+P) did not outperform pretrained MMDA (MMDA+P) we conclude that most of the gain likely comes from pretraining the decoder and attention parameters. However, since training on other languages seems to help the encoder, we conclude that it is likely the synthetic data itself, and not necessarily PSDA, which is of limited use for pretraining the encoder.

Finally table 3. shows the WER of Italian ASR systems and a sample decoded sentence. Appropriate pretraining of the encoder and decoder reduced the WER by 20% absolute in the 4h Italian set, to 56.2%. This performance has been shown to still be usable for some downstream tasks such as topic identification in low-resource settings [29].

7. Conclusion & Future Work

We have presented a new data augmentation scheme, PSDA, and demonstrated that pretraining on augmenting data for both MMDA and PSDA outperforms the monolingual, vanilla MMDA and RNNLM shallow fusion baselines. We have shown that without using any additional transcribed speech in any language we can achieve performance improvements approaching those of multilingual training on related languages. Furthermore, our MMDA and PSDA variants improve upon multilingual systems for Portuguese and Italian. Future work should expand upon PSDA by attempting to more explicitly generate speech like features from text, possibly using a generative adversarial network to encourage the augmenting encoder in PSDA to act as a light-weight TTS engine.

8. References

[1] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, E. Gonina *et al.*, "State-of-the-art speech recognition with sequence-to-sequence models,"

in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4774–4778.

[2] A. Rosenberg, K. Audhkhasi, A. Sethy, B. Ramabhadran, and M. Picheny, "End-to-end speech recognition and keyword search on low-resource languages," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5280–5284.

[3] J. Cho, M. K. Baskar, R. Li, M. Wiesner, S. H. Mallidi, N. Yalta, M. Karafiat, S. Watanabe, and T. Hori, "Multilingual sequence-to-sequence speech recognition: architecture, transfer learning, and language modeling," in *IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 521–527.

[4] J. Chorowski and N. Jaitly, "Towards better decoding and language model integration in sequence to sequence models," in *Interspeech*, 2017, pp. 523–527.

[5] T. Hori, S. Watanabe, Y. Zhang, and W. Chan, "Advances in joint CTC-attention based end-to-end speech recognition with a deep CNN encoder and RNN-LM," in *Interspeech*, 2017, pp. 949–953.

[6] A. Ragni, K. M. Knill, S. P. Rath, and M. J. Gales, "Data augmentation for low resource languages," in *Interspeech*, 2014, pp. 810–814.

[7] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Interspeech*, 2015, pp. 3586–3590.

[8] 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.

[9] 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 *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 4333–4336.

[10] A. Ghoshal, P. Swietojanski, and S. Renals, "Multilingual training of deep neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7319–7323.

[11] S. Watanabe, T. Hori, and J. R. Hershey, "Language independent end-to-end architecture for joint language identification and speech recognition," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2017, pp. 265–271.

[12] B. Li, T. N. Sainath, K. C. Sim, M. Bacchiani, E. Weinstein, P. Nguyen, Z. Chen, Y. Wu, and K. Rao, "Multi-dialect speech recognition with a single sequence-to-sequence model," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4749–4753.

[13] S. Kim and M. L. Seltzer, "Towards language-universal end-to-end speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4914–4918.

[14] S. Toshniwal, T. Sainath, R. Weiss, B. Li, P. Moreno, E. Weinstein, and K. Rao, "End-to-end multilingual speech recognition using encoder-decoder models," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4904–4908.

[15] O. Adams, M. Wiesner, S. Watanabe, and D. Yarowsky, "Massively multilingual adversarial speech recognition," in *North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, 2019, (accepted).

[16] A. Renduchintala, S. Ding, M. Wiesner, and S. Watanabe, "Multi-modal data augmentation for end-to-end ASR," in *Interspeech*, 2018, pp. 2394–2398.

[17] R. Sennrich, B. Haddow, and A. Birch, "Improving Neural Machine Translation Models with Monolingual Data," in *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2016, pp. 86–96.

[18] T. Hayashi, S. Watanabe, Y. Zhang, T. Toda, T. Hori, R. Astudillo, and K. Takeda, "Back-translation-style data augmentation for end-to-end ASR," in *IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 426–433.

[19] M. Mimura, S. Ueno, H. Inaguma, S. Sakai, and T. Kawahara, "Leveraging sequence-to-sequence speech synthesis for enhancing acoustic-to-word speech recognition," in *IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 477–484.

[20] P. Ramachandran, P. Liu, and Q. Le, "Unsupervised pretraining for sequence to sequence learning," in *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2017, pp. 383–391.

[21] C. Gulcehre, O. Firat, K. Xu, K. Cho, L. Barrault, H.-C. Lin, F. Bougares, H. Schwenk, and Y. Bengio, "On using monolingual corpora in neural machine translation," *arXiv preprint arXiv:1503.03535*, 2015.

[22] A. Sriram, H. Jun, S. Satheesh, and A. Coates, "Cold fusion: Training seq2seq models together with language models," in *Interspeech*, 2018, pp. 387–391.

[23] H. Inaguma, J. Cho, M. K. Baskar, T. Kawahara, and S. Watanabe, "Transfer learning of language-independent end-to-end asr with language model fusion," *arXiv preprint arXiv:1811.02134*, 2018.

[24] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 4960–4964.

[25] I. J. Goodfellow, M. Mirza, D. Xiao, A. Courville, and Y. Bengio, "An empirical investigation of catastrophic forgetting in gradient-based neural networks," *arXiv preprint arXiv:1312.6211*, 2013.

[26] S. Toshniwal, A. Kannan, C.-C. Chiu, Y. Wu, T. N. Sainath, and K. Livescu, "A comparison of techniques for language model integration in encoder-decoder speech recognition," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 369–375.

[27] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N.-E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen *et al.*, "Espnet: End-to-end speech processing toolkit," in *Interspeech*, 2018, pp. 2207–2211.

[28] J. R. Novak, N. Minematsu, and K. Hirose, "WFST-based grapheme-to-phoneme conversion: Open source tools for alignment, model-building and decoding." in *FSMNLP*, 2012, pp. 45–49.

[29] M. Wiesner, C. Liu, L. Ondel, C. Harman, V. Manohar, J. Trmal, Z. Huang, N. Dehak, and S. Khudanpur, "Automatic speech recognition and topic identification from speech for almost-zero-resource languages," in *Interspeech*, 2018, pp. 2052–2056.