MULTILINGUAL SEQUENCE-TO-SEQUENCE SPEECH RECOGNITION: ARCHITECTURE, TRANSFER LEARNING, AND LANGUAGE MODELING

Jaejin Cho[1,‡] , *Murali Karthick Baskar*[2,‡], *Ruizhi Li*[1,‡], *Matthew Wiesner*[1],
Sri Harish Mallidi[3], *Nelson Yalta*[4], *Martin Karafiát*[2], *Shinji Watanabe*[1], *Takaaki Hori*[5]

[1]Johns Hopkins University, [2]Brno University of Technology, [3]Amazon, [4]Waseda University,
[5]Mitsubishi Electric Research Laboratories (MERL)

{ruizhili,jcho52,shinjiw}@jhu.edu,{baskar,karafiat}@fit.vutbr.cz,thori@merl.com

ABSTRACT

Sequence-to-sequence (seq2seq) approach for low-resource ASR is a relatively new direction in speech research. The approach benefits by performing model training without using lexicon and alignments. However, this poses a new problem of requiring more data compared to conventional DNN-HMM systems. In this work, we attempt to use data from 10 BABEL languages to build a multilingual seq2seq model as a prior model, and then port them towards 4 other BABEL languages using transfer learning approach. We also explore different architectures for improving the prior multilingual seq2seq model. The paper also discusses the effect of integrating a recurrent neural network language model (RNNLM) with a seq2seq model during decoding. Experimental results show that the transfer learning approach from the multilingual model shows substantial gains over monolingual models across all 4 BABEL languages. Incorporating an RNNLM also brings significant improvements in terms of %WER, and achieves recognition performance comparable to the models trained with twice more training data.

Index Terms: Automatic speech recognition (ASR), sequence to sequence, multilingual setup, transfer learning, language modeling

1. INTRODUCTION

The sequence-to-sequence (seq2seq) model proposed in [1, 2, 3] is a neural architecture for performing sequence classification and later adopted to perform speech recognition in [4, 5, 6]. The model allows to integrate the main blocks of ASR such as acoustic model, alignment model and language model into a single framework. The recent ASR advancements in connectionist temporal classification (CTC) [6, 5] and attention [4, 7] based approaches has created larger interest in speech community to use seq2seq models. To leverage performance gains from this model as similar or better to conventional hybrid RNN/DNN-HMM models requires a huge amount of data [8]. Intuitively, this is due to the wide-range role of the model in performing alignment and language modeling along with acoustic to character label mapping at each iteration.

In this paper, we explore the multilingual training approaches [9, 10, 11] used in hybrid DNN/RNN-HMMs to incorporate them into the seq2seq models. In a context of applications of multilingual approaches towards seq2seq model, CTC is mainly used instead of the attention models. A multilingual CTC is proposed in [12], which uses a universal phoneset, FST decoder and language model. The authors also use linear hidden unit contribution (LHUC) [13] technique to rescale the hidden unit outputs for each language as a way to adapt to a particular language. Another work [14] on multilingual CTC shows the importance of language adaptive vectors as auxiliary input to the encoder in multilingual CTC model. The decoder used here is a simple $argmax$ decoder. An extensive analysis on multilingual CTC mainly focusing on improving under limited data condition is performed in [15]. Here, the authors use a word level FST decoder integrated with CTC during decoding.

On a similar front, attention models are explored within a multilingual setup in [16, 17] based on attention-based seq2seq to build a model from multiple languages. The data is just combined together assuming the target languages are seen during the training. And, hence no special transfer learning techniques were used here to address the unseen languages during training. The main motivation and contribution behind this work is as follows:

- To incorporate the existing multilingual approaches in a joint CTC-attention [18] (seq2seq) framework, which uses a simple beam-search decoder as described in sections 2 and 4
- Investigate the effectiveness of transferring a multilingual model to a target language under various data sizes. This is explained in section 4.3.
- Tackle the low-resource data condition with both transfer learning and including a character-based RNNLM trained with multiple languages. Section 4.4 explains this in detail.

2. SEQUENCE-TO-SEQUENCE MODEL

In this work, we use the attention based approach [2] as it provides an effective methodology to perform sequence-to-sequence (seq2seq) training. Considering the limitations of attention in performing monotonic alignment [19, 20], we choose to use CTC loss function to aid the attention mechanism in both training and decoding. The basic network architecture is shown in Fig. 1.

Let $X = (\mathbf{x}_t | t = 1, \ldots, T)$ be a T-length speech feature sequence and $C = (c_l | l = 1, \ldots, L)$ be a L-length grapheme sequence. A multi-objective learning framework $\mathcal{L}_{\text{mol}}$ proposed in [18] is used in this work to unify attention loss $p_{\text{att}}(C|X)$ and CTC loss $p_{\text{ctc}}(C|X)$ with a linear interpolation weight λ, as follows:

$$\mathcal{L}_{\text{mod}} = \lambda \log p_{\text{ctc}}(C|X) + (1 - \lambda) \log p^{*}_{\text{att}}(C|X) \quad (1)$$

The unified model allows to obtain both monotonicity and effective sequence level training.

‡ All three authors share equal contribution

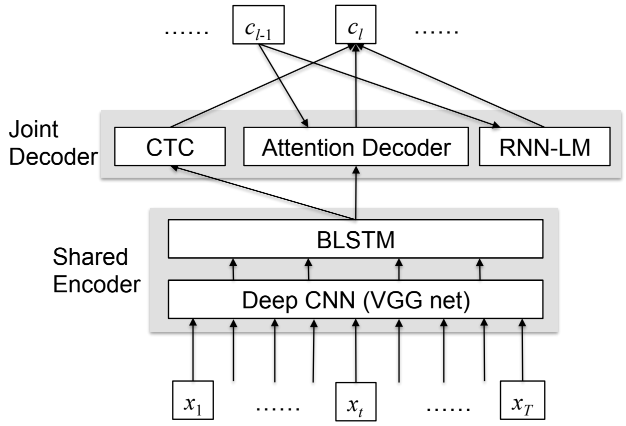

Fig. 1: Hybrid attention/CTC network with LM extension: the shared encoder is trained by both CTC and attention model objectives simultaneously. The joint decoder predicts an output label sequence by the CTC, attention decoder and RNN-LM.

$p_{\text{att}}\left(C|X\right)$ represents the posterior probability of character label sequence C w.r.t input sequence X based on the attention approach, which is decomposed with the probabilistic chain rule, as follows:

$$p^{*}_{\text{att}}\left(C|X\right) \approx \prod_{l=1}^{L} p\left(c_l|c_1^{*}, ..., c_{l-1}^{*},\ X\right), \tag{2}$$

where c_l^{*} denotes the ground truth history. Detailed explanations about the attention mechanism is described later.

Similarly, $p_{\text{ctc}}\left(C|X\right)$ represents the posterior probability based on the CTC approach.

$$p_{\text{ctc}}\left(C|X\right) \approx \sum_{Z\in\mathcal{Z}(C)} p(Z|X), \tag{3}$$

where $Z = (\mathbf{z}_t|t = 1, \ldots, T)$ is a CTC state sequence composed of the original grapheme set and the additional blank symbol. $\mathcal{Z}(C)$ is a set of all possible sequences given the character sequence C.

The following paragraphs explain the encoder, attention decoder, CTC, and joint decoding used in our approach.

Encoder

In our approach, both CTC and attention use the same encoder function, as follows:

$$\mathbf{h}_t = \text{Encoder}(X), \tag{4}$$

where $\mathbf{h}_t$ is an encoder output state at t. As an encoder function Encoder$(\cdot)$, we use bidirectional LSTM (BLSTM) or deep CNN followed by BLSTMs. Convolutional neural networks (CNN) has achieved great success in image recognition [21]. Previous studies applying CNN in seq2seq speech recognition [22] also showed that incorporating a deep CNNs in the encoder could further boost the performance.

In this work, we investigate the effect of convolutional layers in joint CTC-attention framework for multilingual setting. We use the initial 6 layers of VGG net architecture [21] in table 2. For each speech feature image, one feature map is formed initially. VGG net then extracts 128 feature maps, where each feature map is downsampled to $(1/4 \times 1/4)$ images along time-frequency axis via the two maxpooling layers with $stride = 2$.

Attention Decoder:

Location aware attention mechanism [23] is used in this work. Equation (5) denotes the output of location aware attention, where a_{lt} acts as an attention weight.

$$a_{lt} = \text{LocationAttention}\left(\{a_{l-1}\}_{t=1}^{T}, \mathbf{q}_{l-1}, \mathbf{h}_t\right). \tag{5}$$

Here, $\mathbf{q}_{l-1}$ denotes the decoder hidden state, $\mathbf{h}_t$ is the encoder output state as shown in equation (4). The location attention function represents a convolution function * as in equation (6). It maps the attention weight of the previous label a_{l-1} to a multi channel view $\mathbf{f}_t$ for better representation.

$$\mathbf{f}_t = \mathbf{K} * \mathbf{a}_{l-1}, \tag{6}$$

$$e_{lt} = \mathbf{g}^T \tanh(\text{Lin}(\mathbf{q}_{l-1}) + \text{Lin}(\mathbf{h}_t) + \text{LinB}(\mathbf{f}_t)), \tag{7}$$

$$a_{lt} = \text{Softmax}(\{e_{lt}\}_{t=1}^{T}) \tag{8}$$

Equation (7) provides the unnormalized attention vectors computed with the learnable vector $\mathbf{g}$, linear transformation Lin$(\cdot)$, and affine transformation LinB$(\cdot)$. Equation (8) computes a normalized attention weight based on the softmax operation Softmax$(\cdot)$. Finally, the context vector $\mathbf{r}_l$ is obtained by the weighted summation of the encoder output state $\mathbf{h}_t$ over entire frames with the attention weight as follows:

$$\mathbf{r}_l = \sum_{t=1}^{T} a_{lt}\mathbf{h}_t. \tag{9}$$

The decoder function is an LSTM layer which decodes the next character output label c_l from their previous label c_{l-1}, hidden state of the decoder q_{l-1} and attention output $\mathbf{r}_l$, as follows:

$$p\left(c_l|c_1, ..., c_{l-1},\ X\right) = \text{Decoder}(\mathbf{r}_l, q_{l-1}, c_{l-1}) \tag{10}$$

This equation is incrementally applied to form p^{*}_{att} in equation (2).

Connectionist temporal classification (CTC):

Unlike the attention approach, CTC do not use any specific decoder. Instead it invokes two important components to perform character level training and decoding. First component, is an RNN based encoding module $p(Z|X)$. The second component contains a language model and state transition module. The CTC formalism is a special case [6, 24] of hybrid DNN-HMM framework with an inclusion of Bayes rule to obtain $p(C|X)$.

Joint decoding:

Once we have both CTC and attention-based seq2seq models trained, both are jointly used for decoding as below:

$$\begin{aligned}\log p_{\text{hyp}}(c_l|c_1, ..., c_{l-1}, X) =& \\ \alpha \log p_{\text{ctc}}(c_l|c_1, ..., c_{l-1}, X)& \\ +(1-\alpha)\log p_{\text{att}}(c_l|c_1, ..., c_{l-1}, X)&\end{aligned} \tag{11}$$

Here $\log p_{\text{hyp}}$ is a final score used during beam search. α controls the weight between attention and CTC models. α and multi-task learning weight λ in equation (1) are set differently in our experiments.

Table 1: Details of the BABEL data used for performing the multilingual experiments

Usage	Language	Train		Eval		# of characters
		# spkrs.	# hours	# spkrs.	# hours	
Train	Cantonese	952	126.73	120	17.71	3302
	Bengali	720	55.18	121	9.79	66
	Pashto	959	70.26	121	9.95	49
	Turkish	963	68.98	121	9.76	66
	Vietnamese	954	78.62	120	10.9	131
	Haitian	724	60.11	120	10.63	60
	Tamil	724	62.11	121	11.61	49
	Kurdish	502	37.69	120	10.21	64
	Tokpisin	482	35.32	120	9.88	55
	Georgian	490	45.35	120	12.30	35
Target	Assamese	720	54.35	120	9.58	66
	Tagalog	966	44.0	120	10.60	56
	Swahili	491	40.0	120	10.58	56
	Lao	733	58.79	119	10.50	54

3. DATA DETAILS AND EXPERIMENTAL SETUP

In this work, the experiments are conducted using the BABEL speech corpus collected from the IARPA babel program. The corpus is mainly composed of conversational telephone speech (CTS) but some scripted recordings and far field recordings are presented as well. Table 1 presents the details of the languages used in this work for training and evaluation.

80 dimensional Mel-filterbank (fbank) features are then extracted from the speech samples using a sliding window of size 25 ms with 10ms stride. KALDI toolkit [25] is used to perform the feature processing. The fbank features are then fed to a seq2seq model with the following configuration:

The Bi-RNN [26] models mentioned above uses a LSTM [27] cell followed by a projection layer (BLSTMP). In our experiments below, we use only a character-level seq2seq model trained by CTC and attention decoder. Thus in the following experiments we intend to use character error rate (% CER) as a suitable measure to analyze the model performance. However, in section 4.4 we integrate a character-level RNNLM [28] with seq2seq model externally and showcase the performance in terms of word error rate (% WER). In this case the words are obtained by concatenating the characters and the space together for scoring with reference words. All experiments are implemented in ESPnet, end-to-end speech processing toolkit [29].

4. MULTILINGUAL EXPERIMENTS

Multilingual approaches used in hybrid RNN/DNN-HMM systems [11] have been used for for tackling the problem of low-resource data condition. Some of these approaches include language adaptive training and shared layer retraining [30]. Among them, the most benefited method is the parameter sharing technique [11]. To incorporate the former approach into encoder, CTC and attention decoder model, we performed the following experiments:

- Stage 0 - Naive training combining all languages
- Stage 1 - Retraining the decoder (both CTC and attention) after initializing with the multilingual model from stage-0

Table 2: Experiment details

Model Configuration	
Encoder	Bi-RNN
# encoder layers	5
# encoder units	320
# projection units	320
Decoder	Bi-RNN
# decoder layers	1
# decoder units	300
# projection units	300
Attention	Location-aware
# feature maps	10
# window size	100
Training Configuration	
MOL	$5e^{-1}$
Optimizer	AdaDelta
Initial learning rate	1.0
AdaDelta ϵ	$1e^{-8}$
AdaDelta ϵ decay	$1e^{-2}$
Batch size	30
Optimizer	AdaDelta
Decoding Configuration	
Beam size	20
ctc-weight	$3e^{-1}$

(a) Convolutional layers in joint CTC-attention

CNN Model Configuration -2 components	
Component 1	2 convolution layers
Convolution 2D	in = 1, out = 64, filter = 3× 3
Convolution 2D	in = 64, out = 64, filter = 3× 3
Maxpool 2D	patch = 2×2, stride = 2×2
Component 2	2 convolution layers
Convolution 2D	in = 64, out = 128, filter = 3× 3
Convolution 2D	in = 128, out = 128, filter = 3× 3
Maxpool 2D	patch = 2×2, stride = 2×2

- Stage 2 - The resulting model obtained from stage-1 is further retrained across both encoder and decoder

Table 4: Comparison of naive approach and training only the last layer performed using the Assamese language

Model type	Retraining	% CER	% Absolute gain
Monolingual	-	45.6	-
Multi. (after 4^{th} epoch)	Stage 1	61.3	-15.7
Multi. (after 4^{th} epoch)	Stage 2	44.0	1.6
Multi. (after 15^{th} epoch)	Stage 2	41.3	4.3

4.1. Stage 0 - Naive approach

In this approach, the model is first trained with 10 multiple languages as denoted in table 1 approximating to 600 hours of training data. data from all languages available during training is used to build a single seq2seq model. The model is trained with a character label set composed of characters from all languages including both train and target set as mentioned in table 1. The model provides better generalization across languages. Languages with limited data when

Table 3: Recognition performance of naive multilingual approach for eval set of 10 BABEL training languages trained with the train set of same languages

%CER on Eval set	Target languages									
for	Bengali	Cantonese	Georgian	Haitian	Kurmanji	Pashto	Tamil	Turkish	Tokpisin	Vietnamese
Monolingual - BLSTMP	43.4	37.4	35.4	39.7	55.0	37.3	55.3	50.3	32.7	54.3
Multilingual - BLSTMP	42.9	36.3	38.9	38.5	52.1	39.0	48.5	36.4	31.7	41.0
+ VGG	39.6	34.3	36.0	34.5	49.9	34.7	45.5	28.7	33.7	37.4

trained with other languages allows them to be robust and helps in improving the recognition performance. In spite of being simple, the model has limitations in keeping the target language data unseen during training.

Comparison of VGG-BLSTM and BLSTMP

Table 3 shows the recognition performance of naive multilingual approach using BLSTMP and VGG model against a monolingual model trained with BLSTMP. The results clearly indicate that having a better architecture such as VGG-BLSTM helps in improving multilingual performance. Except Pashto, Georgian and Tokpisin, the multilingual VGG-BLSTM model gave 8.8 % absolute gain in average over monolingual model. In case of multilingual BLSTMP, except Pashto and Georgian an absolute gain of 5.0 % in average is observed over monolingual model. Even though the VGG-BLSTM gave improvements, we were not able to perform stage-1 and stage-2 retraining with it due to time constraints. Thus, we proceed further with multilingual BLSTMP model for retraining experiments tabulated below.

4.2. Stage 1 - Retraining decoder only

To alleviate the limitation in the previous approach, the final layer of the seq2seq model which is mainly responsible for classification is retrained to the target language.

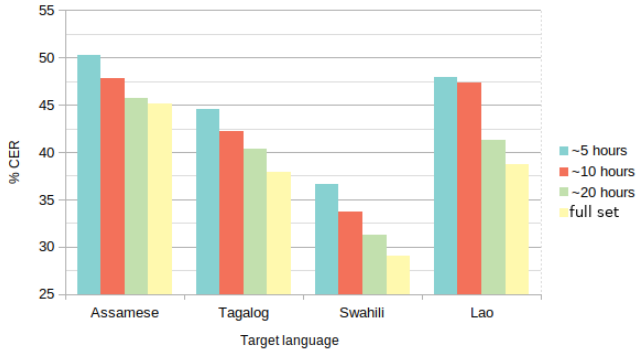

Fig. 2: Difference in performance for 5 hours, 10 hours, 20 hours and full set of target language data used to retrain a multilingual model from stage-1

In previous works [11, 30] related to hybrid DNN/RNN models and CTC based models [12, 15] the softmax layer is only adapted. However in our case, the attention decoder and CTC decoder both have to be retrained to the target language. This means the CTC and attention layers are only updated for gradients during this stage. We found using SGD optimizer with initial learning rate of $1e^{-4}$ works better for retraining compared to AdaDelta.

The learning rate is decayed in this training at a factor of $1e^{-1}$ if there is a drop in validation accuracy. Table 4 shows the performance of simply retraining the last layer using a single target language Assamese.

4.3. Stage 2 - Finetuning both encoder and decoder

Based on the observations from stage-1 model in section 4.2, we found that simply retraining the decoder towards a target language resulted in degrading %CER the performance from 45.6 to 61.3. This is mainly due to the difference in distribution across encoder and decoder. So, to alleviate this difference the encoder and decoder is once again retrained or fine-tuned using the model from stage-1. The optimizer used here is SGD as in stage-1, but the initial learning rate is kept to $1e^{-2}$ and decayed based on validation performance. The resulting model gave an absolute gain of 1.6% when finetuned a multilingual model after 4th epoch. Also, finetuning a model after 15th epoch gave an absolute gain of 4.3%.

Table 5: Stage-2 retraining across all languages with full set of target language data

% CER on	Target Languages			
eval set	Assamese	Tagalog	Swahili	Lao
Monolingual	45.6	43.1	33.1	42.1
Stage-2 retraining	41.3	37.9	29.1	38.7

To further investigate the performance of this approach across different target data sizes, we split the train set into $\sim$5 hours, $\sim$10 hours, $\sim$20 hours and $\sim$full set. Since, in this approach the model is only finetuned by initializing from stage-1 model, the model architecture is fixed for all data sizes. Figure 2 shows the effectiveness of finetuning both encoder and decoder. The gains from 5 to 10 hours was more compared to 20 hours to full set.

Table 5 tabulates the % CER obtained by retraining the stage-1 model with $\sim$full set of target language data. An absolute gain is observed using stage-2 retraining across all languages compared to monolingual model.

4.4. Multilingual RNNLM

In an ASR system, a language model (LM) takes an important role by incorporating external knowledge into the system. Conventional ASR systems combine an LM with an acoustic model by FST giving a huge performance gain. This trend is also shown in general including hybrid ASR systems and neural network-based sequence-to-sequence ASR systems.

The following experiments show a benefit of using a language model in decoding with the previous stage-2 transferred models. Al-though the performance gains in %CER are also generally observed over all target languages, the improvement in %WER was more distinctive. The results shown in the following Fig. 3 are in %WER. “whole” in each figure means we used all the available data for the target language as full set explained before.

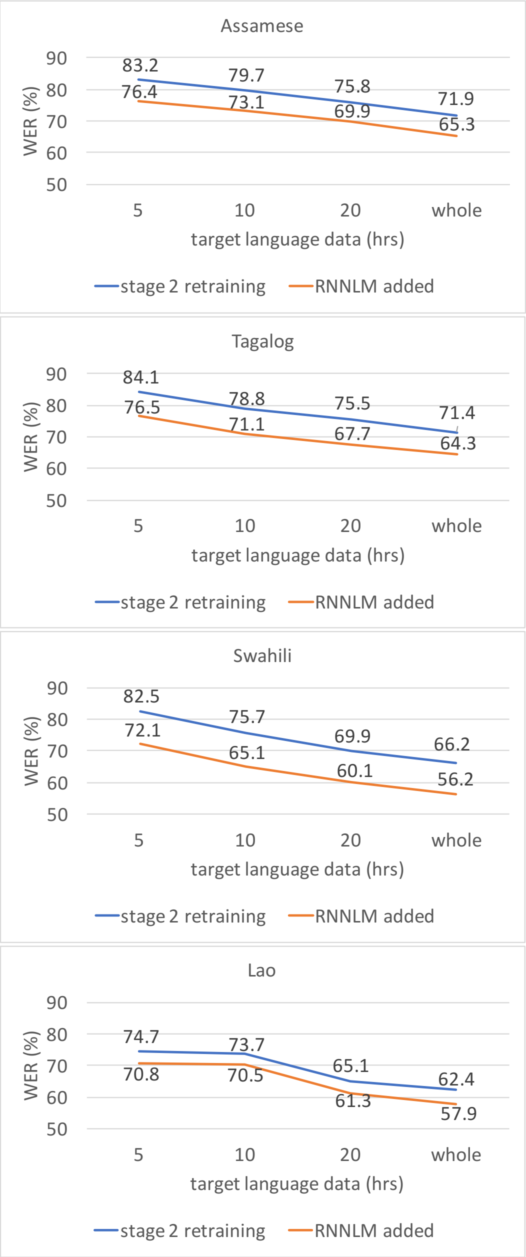

Fig. 3: Recognition performance after integrating RNNLM during decoding in %WER for different amounts of target data

We used a character-level RNNLM, which was trained with 2-layer LSTM on character sequences. We use all available paired text in the corresponding target language to train the LM for the language. No external text data were used. All language models are trained separately from the seq2seq models. When building dictionary, we combined all the characters over all 15 languages mentioned in table 1 to make them work with transferred models. Regardless of the amount of data used for transfer learning, the RNNLM provides consistent gains across all languages over different data sizes.

Table 6: Recognition performance in %WER using stage-2 retraining and multilingual RNNLM

Model type	%WER on target languages			
	Assamese	Tagalog	Swahili	Lao
Stage-2 retraining	71.9	71.4	66.2	62.4
+ Multi. RNNLM	65.3	64.3	56.2	57.9

As explained already, language models were trained separately and used to decode jointly with seq2seq models. The intuition behind it is to use the separately trained language model as a complementary component that works with a implicit language model within a seq2seq decoder. The way of RNNLM assisting decoding follows the equation below:

$$\begin{aligned}\log p(c_l|c_{1:l-1}, X) = &\log p_{\text{hyp}}(c_l|c_{1:l-1}, X) \\ &+ \beta \log p_{\text{lm}}(c_l|c_{1:l-1}, X)\end{aligned} \tag{12}$$

β is a scaling factor that combines the scores from a joint decoding eq.(11) with RNN-LM, denoted as p_{lm}. This approach is called shallow fusion.

Our experiments for target languages show that the gains from adding RNNLM are consistent regardless of the amount of data used for transfer learning. In other words, in Figure 3, the gap between two lines are almost consistent over all languages.

Also, we observe the gain we get by adding RNN-LM in decoding is large. For example, in the case of assamese, the gain by RNN-LM in decoding with a model retrained on 5 hours of the target language data is almost comparable with the model stage-2 retrained with 20 hours of target language data. On average, absolute gain $\sim$6% is obtained across all target languages as noted in table 6.

5. CONCLUSION

In this work, we have shown the importance of transfer learning approach such as stage-2 multilingual retraining in a seq2seq model setting. Also, careful selection of train and target languages from BABEL provide a wide variety in recognition performance (%CER) and helps in understanding the efficacy of seq2seq model. The experiments using character-based RNNLM showed the importance of language model in boosting recognition performance (%WER) over all different hours of target data available for transfer learning.

Table 5 and 6 summarizes, the effect of these techniques in terms of %CER and %WER. These methods also show their flexibility in incorporating it in attention and CTC based seq2seq model without compromising loss in performance.

6. FUTURE WORK

We could use better architectures such as VGG-BLSTM as a multilingual prior model before transferring them to a new target language by performing stage-2 retraining. The naive multilingual approach can be improved by including language vectors as input or target during training to reduce the confusions. Also, investigation of multilingual bottleneck features [31] for seq2seq model can provide better performance. Apart from using the character level language model as in this work, a word level RNNLM can be connected during decoding to further improve %WER. The attention based decoder can be aided with the help of RNNLM using cold fusion approach during training to attain a better-trained model. In near future, we will incorporate all the above techniques to get comparable performance with the state-of-the-art hybrid DNN/RNN-HMM systems.

7. REFERENCES

[1] Ilya Sutskever, Oriol Vinyals, and Quoc V Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[3] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[4] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio, "Attention-based models for speech recognition," in *Advances in neural information processing systems*, 2015, pp. 577–585.

[5] Alex Graves and Navdeep Jaitly, "Towards end-to-end speech recognition with recurrent neural networks.," in *ICML*, 2014, vol. 14, pp. 1764–1772.

[6] Alex Graves, "Supervised sequence labelling," in *Supervised sequence labelling with recurrent neural networks*, pp. 5–13. Springer, 2012.

[7] William Chan, Navdeep Jaitly, Quoc Le, and Oriol 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)*. IEEE, 2016, pp. 4960–4964.

[8] Andrew Rosenberg, Kartik Audhkhasi, Abhinav Sethy, Bhuvana Ramabhadran, and Michael Picheny, "End-to-end speech recognition and keyword search on low-resource languages," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5280–5284.

[9] Laurent Besacier, Etienne Barnard, Alexey Karpov, and Tanja Schultz, "Automatic speech recognition for under-resourced languages: A survey," *Speech Communication*, vol. 56, pp. 85–100, 2014.

[10] Zoltan Tuske, David Nolden, Ralf Schluter, and Hermann Ney, "Multilingual mrasta features for low-resource keyword search and speech recognition systems," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 7854–7858.

[11] Martin Karafiát, Murali Karthick Baskar, Pavel Matějka, Karel Veselỳ, František Grézl, and Jan Černocky, "Multilingual blstm and speaker-specific vector adaptation in 2016 but babel system," in *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, 2016, pp. 637–643.

[12] Sibo Tong, Philip N Garner, and Hervé Bourlard, "Multilingual training and cross-lingual adaptation on CTC-based acoustic model," *arXiv preprint arXiv:1711.10025*, 2017.

[13] Pawel Swietojanski and Steve Renals, "Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014, pp. 171–176.

[14] Markus Müller, Sebastian Stüker, and Alex Waibel, "Language adaptive multilingual CTC speech recognition," in *International Conference on Speech and Computer*. Springer, 2017, pp. 473–482.

[15] Siddharth Dalmia, Ramon Sanabria, Florian Metze, and Alan W Black, "Sequence-based multi-lingual low resource speech recognition," *arXiv preprint arXiv:1802.07420*, 2018.

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

[17] Shubham Toshniwal, Tara N Sainath, Ron J Weiss, Bo Li, Pedro Moreno, Eugene Weinstein, and Kanishka Rao, "Multilingual speech recognition with a single end-to-end model," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[18] Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R Hershey, and Tomoki Hayashi, "Hybrid CTC/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.

[19] Matthias Sperber, Jan Niehues, Graham Neubig, Sebastian StÃŒker, and Alex Waibel, "Self-attentional acoustic models," in *19th Annual Conference of the International Speech Communication Association (InterSpeech 2018)*, 2018.

[20] Chung-Cheng Chiu and Colin Raffel, "Monotonic chunkwise attention," *CoRR*, vol. abs/1712.05382, 2017.

[21] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," 09 2014.

[22] Ying Zhang, Mohammad Pezeshki, Philémon Brakel, Saizheng Zhang, César Laurent, Y Bengio, and Aaron Courville, "Towards end-to-end speech recognition with deep convolutional neural networks," September 2016.

[23] Jan Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems*. 2015, vol. 2015-January, pp. 577–585, Neural information processing systems foundation.

[24] Takaaki Hori, Shinji Watanabe, Yu Zhang, and William Chan, "Advances in joint CTC-attention based end-to-end speech recognition with a deep cnn encoder and RNN-LM," *arXiv preprint arXiv:1706.02737*, 2017.

[25] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," in *Automatic Speech Recognition and Understanding, 2011 IEEE Workshop on*. IEEE, 2011, pp. 1–4.

[26] Mike Schuster and Kuldip K Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[27] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[28] Tomas Mikolov, Stefan Kombrink, Anoop Deoras, Lukar Burget, and Jan Cernocky, "RNNLM-recurrent neural network language modeling toolkit," in *Proc. of the 2011 ASRU Workshop*, 2011, pp. 196–201.

[29] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, et al., “Espnet: End-to-end speech processing toolkit,” *arXiv preprint arXiv:1804.00015*, 2018.

[30] Sibo Tong, Philip N Garner, and Hervé Bourlard, “An investigation of deep neural networks for multilingual speech recognition training and adaptation,” Tech. Rep., 2017.

[31] Frantisek Grézl, Martin Karafiát, and Karel Veselỳ, “Adaptation of multilingual stacked bottle-neck neural network structure for new language,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014.