INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

End-to-end Accented Speech Recognition

Thibault Viglino[1], *Petr Motlicek*[2], *Milos Cernak*[3]
[1]École Polytechnique Fédérale de Lausanne
[2]Idiap Research Institute
[3]Logitech Europe S.A.
thibault.viglino@alumni.epfl.ch, petr.motlicek@idiap.ch, mcernak@logitech.com

Abstract

Correct pronunciation is known to be the most difficult part to acquire for (native or non-native) language learners. The accented speech is thus more variable, and standard Automatic Speech Recognition (ASR) training approaches that rely on intermediate phone alignment might introduce errors during the ASR training. With end-to-end training we could alleviate this problem. In this work, we explore the use of multi-task training and accent embedding in the context of end-to-end ASR trained with the connectionist temporal classification loss. Comparing to the baseline developed using conventional ASR framework exploiting time-delay neural networks trained on accented English, we show significant relative improvement of about 25% in word error rate. Additional evaluation on unseen accent data yields relative improvements of of 31% and 2% for New Zealand English and Indian English, respectively.

Index Terms: Speech recognition, accented speech, accent embedding, multi-task, end-to-end.

1. Introduction

Automatic speech recognition (ASR) is an established research area of speech processing. Current techniques focus mainly on neural networks and are therefore sensitive to the generalization potential of the data. There are many ways to speak, even within the same language. The absence of given accents in the training set often means that a system may operate well for seen accents, and become unusable for unseen accents.

With the advent of voice assistants and other voice user interfaces, ASR systems are still more ubiquitous in our lives and good performance is necessary to get satisfied user experience. ASR systems in some benchmarks surpassed human speech recognition abilities, and their performances are continuously getting better [1]. These good results encourage companies and public services to use ASR systems in their products.

The issue arises when the technology does not help to bridge the gap anymore, but rather digs it deeper. When considering the performance of ASR systems, we often neglect an important fact, which is that the way people talk vary widely across the world. It is even more the case for international languages such as English, which is not only spoken by about 380 million native, but also by close to 740 million non-native speakers [2]. All those speakers, many using English as a linuga franca, have a very wide variety of accents usually influenced by their native language. Pronunciation is considered by some linguists to be the most difficult part to acquire for learners [3]. Consequently, it seems rather hopeful to think that a system well tuned for a certain group of people would work well for everyone.

The accent problem we are facing can be thought of a domain adaptation. That is, how to take the incomplete knowledge we have of a task and expand it without starting from scratch. There have been many works already done in this topic. Multilingual training approach was tackled in [4], applying the idea that the acoustics of languages of different origins are not totally alien to each other and that one could leverage the resources of one to help for the other. The same issue was explored for low-resource ASR through transfer learning [5]. A wise choice of the source language for language adaptation can significantly impact the results as well [6]. The report investigates the idea of first transforming back an accent to a standard form as a processing step using a pair-based accent conversion system [7]. Closer to our work, a recent work on data augmentation for foreign accent ASR [8] states that "Overall, we find speed modification to be a remarkably reliable data augmentation technique for improving recognition of foreign accented speech.". Siamese networks also give a promising avenue of research [9]. These methods use the fact that similar tasks may lead to better generalization by sharing different parts of their network. Other works propose multi-task learning for ASR, applied on both the target language and the source language at the same time [10].

Recently, additional improvements have been reported by investigating two techniques that address the issue of robustness to accented voice: (i) multi-task learning with a model sharing layers for accent and phone classification tasks, respectively, and (ii) accent embeddings, where a standalone model is used to create accent embeddings, further employed as additional input feature for the phone classifier [11].

In our work, aiming to develop a robust accent ASR system, we hypothesize that end-to-end training used on the top of current best-performing models can yield further improvements. As stated above that the pronunciation is heavily impacted by accents, standard ASR training approaches relying on intermediate phone alignment can introduce training errors. With end-to-end training, we aim to alleviate this problem. The proposed solution is implemented as an open-source project[1] and the repository includes all the experiments performed in this paper. The experiments can be reproduced by modifying the `config.py` file and running the `run_experiment.py` script.

The rest of the paper is composed as follows: section 2 proposes the end-to-end multi-task accent training and section 3 describes an experimental setup and evaluation of the proposed system. Finally, section 4 concludes the paper and outlines future work.

2. Multi-task accent network

2.1. Conventional end-to-end with the CTC loss

In our work, the conventional end-to-end ASR system is based on the Deepspeech 2 architecture [12] that uses the Connectionist Temporal Classification (CTC) loss [13]. The network is visually represented in figure 1.

The number and size of the layers is obtained experimentally using cross-validation. Batch normalization, applied on all

[1] `https://github.com/SilvrDuck/AccentedSpeechRecognition`

http://dx.doi.org/10.21437/Interspeech.2019-2122

layers is a good way to reduce overfitting and to speed up the training. In essence, "[It] allows each layer of a network to learn by itself a little bit more independently of other layers." [14].

The input features, in our work we used 40 dimensional high resolution Mel-Frequency Cepstral Coefficients (MFCCs) computed by the Kaldi framework [15], are first processed by the 2 convolutional layers with 32 channels, a stride of 2 and a big kernel of first 41×11 and then 21×11. The task of the convolutional layers is to extract the features of the short-time frame in a way that is "meaningful" to the rest of the network.

After these convolutions, the data flows through 5 recurrent layers. A recurrent network allows us to operate with variable input length and to learn the temporal relationship between the frames. We use bidirectional Gated Recurrent Units (GRU) [16] in this step. GRU layers were found to outperform Long Short-Term Memory (LSTM) layers on smaller dataset, while having less parameters (and are thus quicker to train) [17].

After the recurrent layers, 2 simple fully connected layers with the Rectified Linear Units (ReLU) follow. The ReLU activation function is known to perform well in the context of speech recognition [18]. They have the advantage of sparse activation and efficient computation over other activations like *tanh* or *sigmoid* for example.

The final layer uses a *softmax* activation function. The function produces a probability distribution over the set of characters, and implements the CTC loss. The CTC loss is a powerful tool in the realm of sequence to sequence prediction where the relative lengths might differ. The CTC allows to work without any label alignment of the input and the output data.

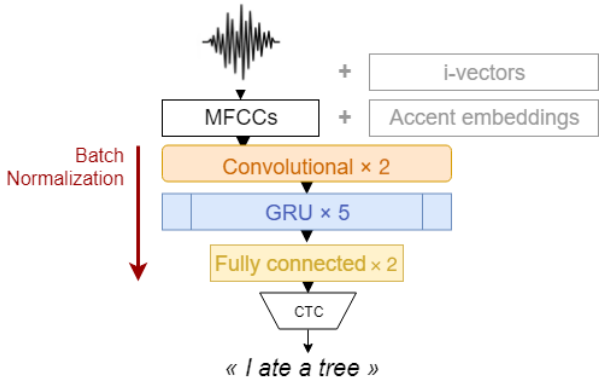

Figure 1: *The conventional end-to-end network we used to benchmark the other variants of our models. We also used the same architecture when exploring the use of i-vectors and accent embeddings.*

The Adam optimization algorithm [19] was used during the backpropagation. It is an efficient algorithm based on the Stochastic Gradient Descent (SGD). It has the advantage to use an evolving learning rate, and is known to be robust and fast, while in certain cases also getting better results than SGD [20].

2.2. Multi-task

Multi-task learning is an active area of machine learning in which it is assumed that, by analogy to the human brain, a model performing multiple tasks will generalize better in any given task. In practice, the network has multiple outputs, and therefore multi-component loss computed in order to do the backpropagation. It was shown, for example, that this method can help a model perform better on low resources task, while not necessarily getting better or worse on the main task [21]. As we deal with accented data, we hypothesized that by modeling jointly the class of the accent as well as the sequence of symbols in the utterance, the model will get more robust to unseen accents. In our case, we would not expect any major improvement on ASR task on the test set, but rather to see an improvement over the unseen accents (e.g. on Test-NZ test subset described later in section 3.2.1).

The multi-task model was built upon the base architecture described in section 2.1, by adding a secondary accent classification network after the first recurrent layers. This can be seen in figure 2. The network was then trained based on a combined loss with the following formula:

$$Loss(y) = \lambda \cdot CTC(y) + (1 - \lambda) \cdot CrossEntropy(y),$$

where λ is a hyper-parameter called the *mixing factor*.

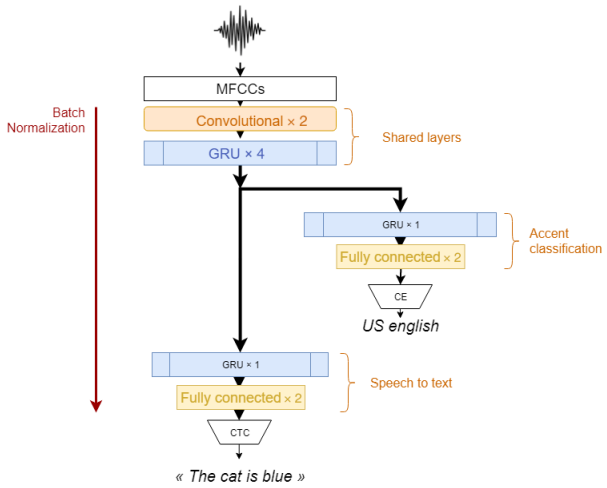

Figure 2: *The multitask architecture. Accent classification and end-to-end speech recognition are performed at the same time.*

The λ parameter significantly impacts the performance of the transcription task. We thus studied the effect of varying the λ on both the accent classification and the text transcription tasks to find an optimal parameter to use. We selected the value of $\lambda = 0.9$, where the accent classification was less sensible to the variation of the parameter on the dev set. This can be observed in figure 3.

2.3. Accent embedding

We postulate that using a good representation of the accents at the input of the end-to-end network will increase the generalization of accented speech ASR training. First, we used the i-vectors approach, similarly to [11], though it is just an approximation to accent modelling, hypothesizing that different dialects are associated with different speakers, and thus the i-vectors might not model the fundamental accent characteristics. Employing i-vectors jointly with the short-term spectral features has been used in literature to enhance the performance of ASR systems with respect to speaker variation [22, 23]. Second, we trained accent embeddings directly with a standalone network trained to identify accents.

To compute the embeddings, we use the model shown in figure 4, with 5 GRU recurrent layers of 800 units and 3 fully

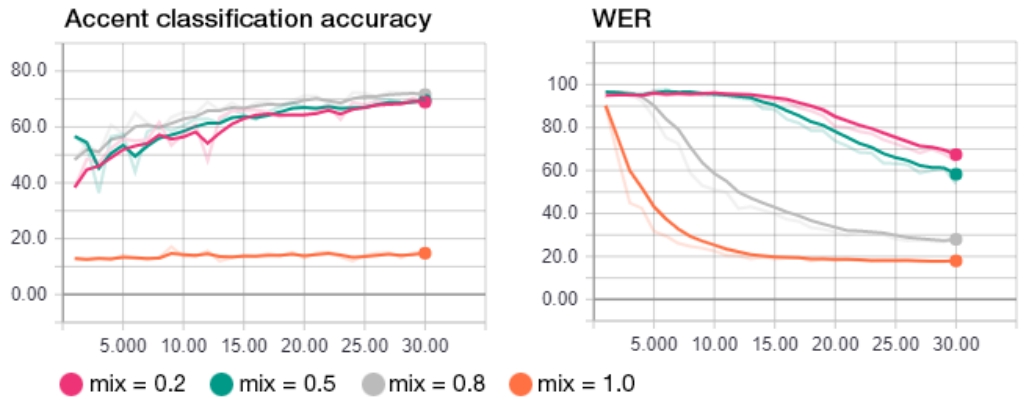

Figure 3: *Effects of the variation of the multi-task mixing factor on the two tasks of the network. The x axis represent the training epochs. We can see that as λ varies, the accent classification accuracy remains stable. Of course for $\lambda = 1$, the Cross Entropy loss is ignored and hence no progress is shown. The Word Error Rate (WER) on its side seems to get better with higher λ. The curves have been smoothed for clarity, lighter curves are the originals.*

connected layers. The model was trained using the cross-entropy loss. After training, the output of the penultimate layer, called the bottleneck layer, was used as an accent embedding for the input sample. To have a visual intuition of what the accent embedding network learns, we plotted the first two principal components of the embeddings using the Principal Component Analysis (PCA) algorithm, shown in figure 5.

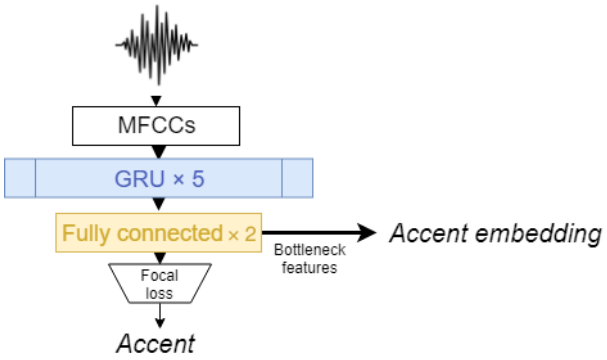

Figure 4: *The standalone network used to create accent embeddings in order to feed them to the main model.*

We then reused the model described above, but concatenated the accent embeddings to the inputs. We experimented with two size of embeddings, 100 and 256. As we will see, an embedding size of 100 performed slightly better.

2.4. Decoding

ASR task is performed with a Language Model (LM) trained on the transcripts of the training set. The trigram LM was trained in the ARPA format using the SRILM language modeling toolkit [24]. The LM contained a total of 6,854 unigrams, 29,350 bigrams and 34,741 trigrams.

The decoding is performed by a beam search decoder [25]. In contrast with a greedy decoder, which would find the most likely path in the output matrix by maximizing the product of the character probabilities at each time step, the beam search decoder considers a "beam" of n most likely decoding hypotheses,

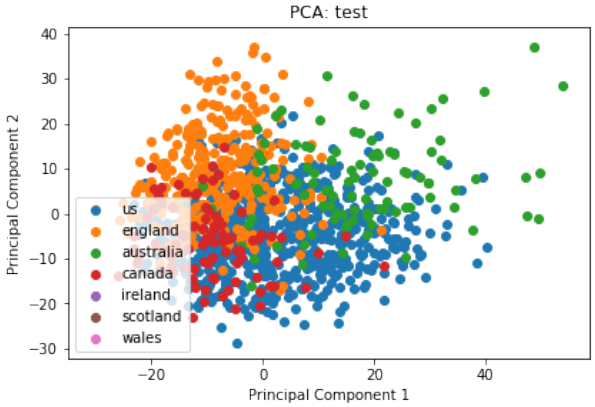

Figure 5: *PCA projection of the embeddings. The accents tend to cluster reasonably well upon visual inspection, but there is still a lot of overlap between the different classes.*

with n being the beam width parameters set to 100 in our case. When the whole probability matrix is traversed, the path with the highest score is selected as the final decoding and suppression of repeated characters and blank symbols are performed.

3. Experiments

3.1. Baseline

The baseline in our work is represented by the Kaldi-based ASR framework exploiting time-delay neural networks trained on accented English [11]. A joint training of an accent classifier and multi-accent acoustic model with optional input accent ebmeddings was used. The implementation of the baseline is also open-sourced[2].

3.2. Data

We followed the baseline work and used the same training and evaluation speech database, including the same subsets as pro-

[2] https://github.com/abhinavjain03/kaldi-accentsmultitask

Table 1: *Composition of the different data splits. The numbers in parentheses denote the percentage of an accent in the set. US: United States, EN: England, AU: Australia, CA: Canada, SC: Scotland, IR: Ireland, WE: Wales, NZ: New Zealand, IN: India. This table was taken directly from [11].*

Dataset	Accents	Hrs of speech	No. of sentences	No. of words
Train	US(32), EN(32), AU(14), CA(13), SC(5), IR(3), WE(1)	34.3	30,896	283,862
Dev	US(55), EN(30), AU(8), CA(7)	1.26	1,142	10,386
Test	US(56), EN(27), AU(9), CA(8)	1.25	1,127	10,467
Test-NZ	NZ(100)	0.59	536	5,089
Test-IN	IN(100)	1.33	1,200	10,780

vided by the authors[3]. In this section, we briefly introduce the dataset and outline its relevance to the task.

Common Voice [26] is Mozilla Research project to create an open-access dataset of human voices. The goal is to provide useful open data for ASR researchers. At the time of writing, the site reports 1.9 thousand hours recorded (with 1.5 validated) spreading across twenty-two distinct languages. The project is crowdsourced and anyone can contribute by recording themselves. Samples are then randomly shown to other users that can vote for the correctness of the sample. A majority voting scheme then labels them as valid, invalid or pending. People can self-report additional demographics such as age range, gender and accent. This last option is of utmost interest to us, since an accent label will allow us to try out our different strategies. Random sample examination gave us the impression that the labels were accurate, but we need to remain aware of the crowdsourced nature of the dataset when drawing conclusions.

3.2.1. Data splits

We used a data subset that contains only the English language recordings with reported accents, in total about 35 hours of speech. Table 1 defines specific splits of the data in detail.

In order to see how the methods performed with unseen accents, two sets contain accents that were left-out from the training, dev and test sets. Those are labeled `Test-NZ` and `Test-IN`, containing New Zealandic and Indian accents respectively. We also note that the train set contains seven different varieties of English, whereas the dev and test sets contain only a subset of four of them. This is due to the fact that the data for Scottish, Irish and Welsh accents were very scarce.

3.3. Results

A summary of the best performances of all the models is presented in Table 2. First, we have observed a significant improvement of the conventional end-to-end CTC training comparing to the baseline regular time-delay neural networks training. This difference still holds for the baseline multi-task with/without accent embeddings training. It was surprising to us as we expect CTC to need bigger amount of data to perform well. We postulate that the good results are due to the nature of the Common Voice dataset, which is very redundant, that is that several speakers say the same sentences. Second, we evaluated different variants of the CTC training, with different configurations of the multi-task network, and we have not observed an improvement. Third, by using multi-task and accent embeddings training we observed further improvements on the `Test`, `Dev` and `Test-NZ` sets. Similarly as in the baseline experiments (i.e., regular vs. multi-task and accent embeddings), we have obtained similar relative improvements. This implies that New Zealandic must be close to the accents used for training. We assume that this is due to the presence of Australian in the training set. Over the three dataset, it seems that the embedding and i-vector approaches are the most promising for an end-to-end approach. We speculate that this is due to the fact that those methods actually help the model to "look for the right features" in an accent. We note the very similar performance of the i-vectors and the accent embeddings of size 100. As this is also the size of the i-vectors, it seems that similar information was encoded in them. Combining them seems to improve the overall performance, although just the embedding seems to perform better sometimes, especially in the case of the unseen accent.

However, the performance of the proposed end-to-end systems on second unseen `Test-IN` data has not been so significant as in the `Test-NZ` case. We improved it over the baseline, but overall performance is still unsatisfactory. Indian data are probably too distinct and further work is needed.

Table 2: *Summary of the results of every methods. Multi 3-2 is our multitask model with 3 shared GRU layers and 2 not shared, while 4-1 has 4 shared layers and 1 not shared.*

Model	**WER in %**			
	Common Voice			
	Test	*Dev*	*Test-NZ*	*Test-IN*
Baseline Regular	*23.3*	*23.1*	*24.9*	*55.2*
Baseline Multi	*20.6*	*21.2*	*23.2*	*52.1*
Baseline Embed	*19.7*	*20.0*	*22.7*	*51.2*
Conventional CTC	15.3	17.1	15.6	50.9
Multi 3-2	16.0	17.8	16.2	51.5
Multi 4-1	15.8	17.3	15.6	51.1
I-Vect	15.1	16.6	15.7	**50.2***
Embed 100	15.1	16.8	**15.5***	50.5
Embed 256	15.3	16.6	16.7	52.8
Embed 100 + I-Vect	**14.7***	**16.3***	15.8	52.0

4. Conclusion

This paper presented the accent robust ASR training with multitask models and accent embeddings integrated in the end-to-end framework. The accuracy of the ASR system has been improved significantly comparing to the baseline trained with conventional time-delay neural networks. In addition, we have shown how much the end-to-end model can benefit from the multitask training. As future work, using another architecture, such as the attention networks with *Transformers* [27], can still leverage the potential of the side network. There are also avenues to explore the focal loss [28] that has proven to be efficient in context of under-resourced languages.

[3] `https://sites.google.com/view/accentsunearthed-dhvani/`

5. References

[1] G. Synnaeve. Wer are we? [Online]. Available: https://github.com/syhw/wer_are_we

[2] S. International. Number of english speakers. [Online]. Available: https://www.ethnologue.com/language/eng

[3] A. Pourhosein Gilakjani, "A study on the situation of pronunciation instruction in esl/efl classrooms," *Journal of Studies in Education*, vol. 1, 08 2011.

[4] S. Tong, P. N. Garner, and H. Bourlard, "Cross-lingual adaptation of a ctc-based multilingual acoustic model," *Speech Communication*, vol. 104, pp. 39 – 46, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016763931830030X

[5] J. Cui, B. Kingsbury, B. Ramabhadran, A. Sethy, K. Audhkhasi, X. Cui, E. Kislal, L. Mangu, M. Nussbaum-Thom, M. Picheny, Z. Tske, P. Golik, R. Schlter, 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 *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, Dec 2015, pp. 259–266.

[6] N. K Chibuye, T. Rosenstock, and B. DeRenzi, "Cross-language phoneme mapping for low-resource languages: An exploration of benefits and trade-offs," B. Yegnanarayana, Ed. ISCA, 09 2018, pp. 2623–2627. [Online]. Available: https://doi.org/10.21437/Interspeech.2018

[7] A. Bearman, K. Josund, and G. Fiore. (2017) Accent conversion using artificial neural networks. [Online]. Available: http://web.stanford.edu/class/cs224s/reports/Amy_Bearman.pdf

[8] T. Fukuda, R. Fernandez, A. Rosenberg, S. Thomas, B. Ramabhadran, A. Sorin, and G. Kurata, "Data augmentation improves recognition of foreign accented speech," in *Proc. Interspeech 2018*, 2018, pp. 2409–2413. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1211

[9] A. Siddhant, P. Jyothi, and S. Ganapathy, "Leveraging native language speech for accent identification using deep siamese networks," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2017.

[10] S. Ghorbani and J. H. Hansen, "Leveraging native language information for improved accented speech recognition," in *Proc. Interspeech 2018*, 2018, pp. 2449–2453. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1378

[11] A. Jain, M. Upreti, and P. Jyothi, "Improved accented speech recognition using accent embeddings and multi-task learning," 09 2018, pp. 2454–2458.

[12] D. Amodei, R. Anubhai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, J. Chen, M. Chrzanowski, A. Coates, G. Diamos, E. Elsen, J. Engel, L. Fan, C. Fougner, T. Han, A. Hannun, B. Jun, P. LeGresley, L. Lin, S. Narang, A. Ng, S. Ozair, R. Prenger, J. Raiman, S. Satheesh, D. Seetapun, S. Sengupta, Y. Wang, Z. Wang, C. Wang, B. Xiao, D. Yogatama, J. Zhan, and Z. Zhu, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *ICML'16: Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, M. F. Balcan and K. Q. Weinberger, Eds. JMLR.org, 2015.

[13] 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*, ser. ICML '06. New York, NY, USA: ACM, 2006, pp. 369–376. [Online]. Available: http://doi.acm.org/10.1145/1143844.1143891

[14] F. Doukkali. Batch normalization in neural networks. [Online]. Available: https://towardsdatascience.com/batch-normalization-in-neural-networks-1ac91516821c

[15] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011, iEEE Catalog No.: CFP11SRW-USB.

[16] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734. [Online]. Available: https://www.aclweb.org/anthology/D14-1179

[17] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014.

[18] L. Tth, "Phone recognition with deep sparse rectifier neural networks," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 6985–6989.

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, San Diego, CA, USA, May 7-9, 2015*, 2014.

[20] S. J. Reddi, S. Kale, and S. Kumar, "On the convergence of adam and beyond," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=ryQu7f-RZ

[21] L. Kaiser, A. N. Gomez, N. Shazeer, A. Vaswani, N. Parmar, L. Jones, and J. Uszkoreit, "One model to learn them all," 2017.

[22] S. Garimella, A. Mandal, N. Strom, B. Hoffmeister, S. Matsoukas, and S. H. K. Parthasarathi, "Robust i-vector based adaptation of dnn acoustic model for speech recognition," in *Proc. of Interspeech*, 2015.

[23] M. Karafit, L. Burget, P. Matjka, O. Glembek, and J. ernock, "ivector-based discriminative adaptation for automatic speech recognition," in *2011 IEEE Workshop on Automatic Speech Recognition Understanding*, Dec 2011, pp. 152–157.

[24] A. Stolcke, "SRILM - an extensible language modeling toolkit," in *7th International Conference on Spoken Language Processing, ICSLP2002 - INTERSPEECH 2002, Denver, Colorado, USA, September 16-20, 2002*, 2002.

[25] M. Freitag and Y. Al-Onaizan, "Beam search strategies for neural machine translation," in *Proceedings of the First Workshop on Neural Machine Translation, 2017*, 2017.

[26] Mozilla. Common voice. [Online]. Available: https://voice.mozilla.org/

[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30: 31st Annual Conference on Neural Information Processing Systems (NIPS 2017)*, 2017.

[28] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018.