IberSPEECH 2021
24-25 March 2021, Valladolid, Spain

TRIBUS: An end-to-end automatic speech recognition system for European Portuguese

Carlos Carvalho[1,2], *Alberto Abad*[1,2]

[1]INESC-ID, Lisbon, Portugal
[2]Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal
carlos.f.carvalho@tecnico.ulisboa.pt, alberto.abad@inesc-id.pt

Abstract

End-to-end automatic speech recognition (ASR) approaches have emerged as a competitive alternative to traditional HMM-based ASR systems. Unfortunately, most end-to-end ASR systems are not easily reproduced since they require vast amounts of data and computational resources that are only available for a reduced set of companies and labs worldwide. Consequently, the performance of these systems is not very well known for low resource languages to the best of our knowledge. European Portuguese is one of those languages. In this work, we present a set of experiments to train and assess some of the current most successful end-to-end ASR approaches for European Portuguese. The proposed system, named TRIBUS, is a hybrid CTC-attention end-to-end ASR combining data from three different domains: read speech, broadcast news and telephone speech. For comparison purposes, we also train a state-of-the-art HMM-based baseline on the same data. Experimental results show that TRIBUS achieves 8.40% character error rate (CER) on the broadcast news test set without the need of a language model, in contrast to the 4.33% CER attained by the HMM baseline on the same set using an in-domain language model. We consider this result quite promising, especially for highly unpredictable vocabulary ASR applications.

Index Terms: automatic speech recognition, end-to-end, hybrid CTC-attention, low resources

1. Introduction

Speech recognition technology is submerged in our society more than ever. Products like Siri, Cortana, Google Now and Amazon Echo Alexa which belong to big companies, like Apple, Microsoft, Google and Amazon, respectively, are part of our every day lives. This high tech translates into a significant number of applications (e.g., healthcare [1] and autonomous vehicles [2]) which have contributed to increase the quality of live in our society.

Traditionally, large vocabulary continuous speech recognition (LVCSR) systems, i.e., HMM-based systems, rely on sophisticated modules including acoustic, phonetic and language models, which are manually created by specialized computational linguists and engineers . Since all these modules do not optimize the same goal, the ASR system final objective typically has more difficulties in achieving a global optimum. Furthermore, HMM systems and n-gram language models make conditional independence assumptions, whereas real speech does not follow those strict assumptions. To overcome these limitations, it is possible to replace the HMM-based system with a single deep neural network, which is trained following a global optimization procedure. Also, by removing the engineering required for the usual alignment, bootstrapping, clustering and decoding with finite-state transducers (FSTs), characteristic of most HMM-based systems, the training and decoding process becomes more straightforward. This new paradigm, named end-to-end, directly maps an input sequence of acoustic features to an output sequence of tokens, i.e., characters or sub-words, [3, 4, 5, 6].

Some widely used contemporary end-to-end approaches are: connectionist temporal classification (CTC) [3, 7], attention encoder-decoder (AED) [5, 8] and RNN Transducer (RNN-T) [9]. CTC's main problem is that it is not capable of modelling language [10] because it considers each label in the output sequence to be independent of each other. To solve CTC-based models independence assumption, the RNN-T approach was proposed [9]. In contrast to CTC, RNN-T does not make assumptions about label independence when enumerating the hard alignments. However, the main disadvantage of CTC-based and RNN-T systems is that, since they first enumerate all hard alignments and then aggregate them, there could be many illogical paths. Attention-based models solve this problem by creating a direct soft alignment between input and output, with the support of an attention mechanism. One of the main issues of attention-based models is the monotonic alignment problem. As a result, the attention mechanism can allow extremely nonsequential alignments between input frames and output tokens [11]. To solve this, hybrid CTC-attention models were proposed in [12]. These models use the advantages of both CTC-based and attention-based architectures in training and decoding.

The main drawback of these end-to-end systems, mentioned above, is that they require a considerable number of training hours to achieve state-of-the-art performance results when compared to traditional HMM-based systems [4]. For English ASR, corpora such as TED-LIUM [13], and Librispeech [14] offer great possibilities for researchers to experiment and compare large end-to-end ASR systems. However, this is not the case for European Portuguese (EP), mainly due to the lack of large scale speech data resources publicly available, either paid or for free.

The main contribution of this work is the development and assessment of the first known end-to-end ASR system for EP in a low resource scenario, by using one of the most successful end-to-end ASR approaches. All corpora used for the experiments of this work correspond to small to medium sized data sets collected by INESC-ID over the past years. For comparison purposes, we also report results obtained with a conventional HMM system trained on the same data.

The remainder of the paper is organized as follows. We start by describing the corpus used to train the end-to-end sys-

This work has been partially supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UIDB/50021/2020. Authors would like to thank Pedro Esteves and Thomas Rolland.

10.21437/IberSPEECH.2021-40

tem and the HMM-based baseline in Section 2. Section 3 gives a brief description of the acoustic feature extraction and a description of the central architecture used to train the end-to-end ASR TRIBUS system. Section 4 details the experimental setup including the baseline system, the results and a comparison between the proposed model and baseline. Finally, a concluding summary is presented in Section 5.

2. EP corpus

This section provides a detailed overview of the speech data resources collected by INESC-ID that helped to create the TRIBUS corpus, followed by a description of the language and pronunciation models used.

2.1. Speech data resources and partitions

The TRIBUS corpus training set is a collection of three training sets from three datasets representing different domains: read speech, broadcast news and telephone read speech. INESC-ID participated in the design, collection, processing, transcription and/or distribution of these corpora over the past years. The validation and test sets of the TRIBUS corpus used in the present work are the original ones from each corpus, except for telephone read speech data, where the design process will be detailed below:

Read speech: The read speech corpus used is BD-PÚBLICO [15]. Similar to Wall Street Journal corpus [16], BD-PÚBLICO was created from the Portuguese newspaper Público in 1997. BD-PÚBLICO contains 120 different speakers, where ages range from 19 to 28 years old. It is particularly designed for research and development of speaker-independent continuous speech recognition approaches. The training set contains around 23 hours with 100 speakers and 8389 utterances, and the validation and test set contains 2 hours and 10 speakers each. The validation set contains 584 utterances and the test set contains 592 utterances. Finally, the sampling rate of this corpus is 16kHz.

Broadcast news speech: The EP broadcast news (BN) corpus used in this work is ALERT [17], which contains spontaneous speech from BN shows. ALERT was created in cooperation with RTP, a public service broadcasting organization from Portugal. The training data contains around 60 hours of speech with 1366 speakers and 47552 utterances. The validation set contains 8 hours of data with 260 speakers and 6222 utterances, and the test set has 6 hours has 175 speakers and 4701 utterances. Finally, this corpus is sampled at 16kHz.

Telephone speech: The telephone speech data considered belongs to the SPEECHDAT corpus [18], a collection of read speech utterances from telephone calls, collected by Portugal Telecom, a Portuguese telecommunications operator currently known as Altice Portugal. SPEECHDAT contains two main recording phases: SPEECHDAT 0 and SPEECHDAT 1. Each telephone call included in the database contains 33 read items and 7 spontaneous answers, where some contain demographic information. Only 9 phonetically rich sentences, from the set of 33 items, were used in this work. As opposite to ALERT and BD-PÚBLICO, an experimental setup for SPEECHDAT was created. When working with SPEECHDAT, we noticed that from all the 36243 utterances from SPEECHDAT 1 only 3622 are unique, and from the total 9000 utterances from SPEECHDAT 0 only 904 are unique. Furthermore, SPEECHDAT 0 and SPEECHDAT 1 are two disjoint sets. For this reason, SPEECHDAT 1 was chosen for the training set, and we divided SPEECHDAT 0 into two parts: the validation set and the test set. This data splitting process was made such that the number of female and male speakers was approximately the same for each set. Overall, the training set contains approximately 63 hours with 4027 speakers and 36243 utterances, and the validation and test set contains 9 hours each. Finally, the sampling rate of this corpus is 8kHz.

Table 1: *Summary of the number of utterances, speakers and hours for the training, validation and test set of the TRIBUS corpus.*

	#utterances	**#speakers**	**#hours**
Training set	92184	5493	146
Validation sets			
ALERT	6222	260	8
BD-PÚBLICO	584	10	2
SPEECHDAT	4497	500	9
Test sets			
ALERT	4701	175	6
BD-PÚBLICO	592	10	2
SPEECHDAT	4503	501	9

The total amount of hours, number of utterances and number of speakers in the training, validation and test partitions of the TRIBUS corpus are presented in Table 1. All data was downsampled to 8kHz to match the telephone data sampling rate.

2.2. Language and pronunciation models

The language model (LM) for each set, used in the HMM-based baseline, is the one that comes with each corpus, except for SPEECHDAT. ALERT language model was designed by interpolating three distinct LMs. The first is a backoff 4-gram LM, trained on a word corpus of newspapers texts containing 700M words. This out-of-domain corpus was collected from the web. The second LM is a backoff 3-gram LM trained on an in-domain corpus of broadcast news transcripts, with around 531k words. Finally, the third model is a backoff 4-gram LM, estimated from the EP web newspapers, collected the week before creating the interpolated LM. The final interpolated LM is a 4-gram LM with Kneser-Ney modified smoothing, 100k 1-gram, 7.5M 2-gram, 14M 3-gram and 7.9M 4-gram. Following, BD-PÚBLICO language model is a backoff 3-gram closed model.

To create the language model for SPEECHDAT, we first estimated a backoff 3-gram model with Kneser-Ney smoothing combined with Good-Turing smoothing. To avoid over-fitting due to the small linguistic variability in the training set, mentioned above, we interpolated this 3-gram LM model with BD-PÚBLICO LM [15]. An additional step was performed to normalize the notation of all the noise (e.g., _nsnoise_) and disfluencies (e.g., _ehm_hmm_) across the three datasets. Finally, for the TRIBUS corpus, we collected a lexicon of 108358 pronunciations, obtained from publicly available resources.

3. End-to-end model for EP ASR

First, we will describe how the acoustic features are created. Next, the main idea behind the attention architecture used will be mentioned, and finally, the end-to-end hybrid CTC-attention system named TRIBUS, depicted in Figure 1, will be described.

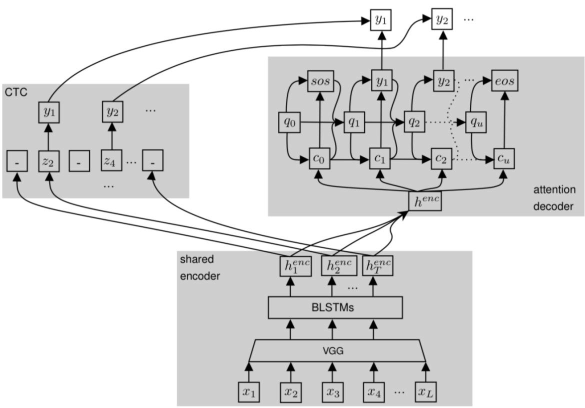

Figure 1: *TRIBUS hybrid CTC-attention architecture. Adapted from [22].*

3.1. Acoustic features

The acoustic features consist of 80-dimensional Mel filterbank energies plus 3 additional pitch features, extracted with Kaldi [19], making the final size of the acoustic vector equal to 83.

3.2. Attention-based architecture

The attention architecture contains three models: the encoder, the hybrid attention mechanism and decoder. The encoder network is described as follows:

$$\boldsymbol{h}_t^{enc} = \mathit{Encoder}(\boldsymbol{x}), \tag{1}$$

which is in charge of converting the input features $\boldsymbol{x}$ into a framewise hidden vector $\boldsymbol{h}_t^{enc}$. Then, the hybrid attention weight is computed as:

$$\alpha_{ut} = \mathit{Hybrid\ attention}(\boldsymbol{q}_{u-1}, \{\alpha_{u-1}\}_{t=1}^{T}, \boldsymbol{h}_t^{enc}), \tag{2}$$

where α_{ut} is the weight that says how much attention is going to vector $\boldsymbol{h}_t^{enc}$, in order to compute output y_u, and $\boldsymbol{q}_{u-1}$ is the last hidden state of the long short-term memory (LSTM) [20] present in the decoder network, mentioned with more detail below. After computing all weights corresponding to all framewise hidden vectors $\boldsymbol{h}_t^{enc}$, we compute a weighted summation of hidden vectors $\boldsymbol{h}_t^{enc}$ to form the hidden vector $\boldsymbol{c}_u$:

$$\boldsymbol{c}_u = \sum_{t=1}^{T} \alpha_{ut} \boldsymbol{h}_t^{enc}. \tag{3}$$

At last, the decoder uses the weighted summation $\boldsymbol{c}_u$ and the last output y_{u-1} to compute the new output y_u:

$$p(y_u | y_1 ... y_{u-1}, \boldsymbol{x}) = \mathit{Decoder}(\boldsymbol{c}_u, y_{u-1}). \tag{4}$$

3.2.1. Encoder network

The encoder network used, Eq. 1, consists of two initial blocks of the VGG layer [21]. With this, the number of frames is reduced approximately by a factor of 4. Following, there are 4 BLSTM layers with 1024 hidden and output units. Each BLSTM layer is followed by a linear projection layer and the final output is 1024 features for every reduced frame.

3.2.2. Hybrid attention mechanism

The hybrid attention mechanism in Eq. 2 is decomposed as:

$$\{\boldsymbol{f}_t\}_{t=1}^{T} = \boldsymbol{K} * \boldsymbol{\alpha}_{u-1}, \tag{5}$$

where each $\boldsymbol{f}_t$ is a vector of size 10 and $*$ denotes a 1D convolution operation along axis t, with the convolution parameter $\boldsymbol{K}$, to produce the set of features $\{\boldsymbol{f}_t\}_{t=1}^{T}$.

Then, we can compute the energy value as:

$$\begin{aligned} e_{ut} = \boldsymbol{g}^T tanh(&\text{LinearNN}(\boldsymbol{q}_{u-1}) \\ &+ \text{LinearNNB}(\boldsymbol{h}_t^{enc}) \\ &+ \text{LinearNN}(\boldsymbol{f}_t)), \end{aligned} \tag{6}$$

where LinearNN corresponds to a linear transformation with learnable matrix parameters and LinearNNB to an affine transformation, with learnable matrix and bias vector parameters. The number of output features used for the three linear networks is 320. Then, we use all e_{ut} values and apply a softmax function to get the attention weight α_{ut}, so that we can compute the target output y_u:

$$\alpha_{ut} = Softmax(\{e_{ut}\}_{t=1}^{T}). \tag{7}$$

3.2.3. Decoder network

The decoder network (Eq. 4) computes:

$$Decoder(.) = Softmax(\text{LinearNNB}(LSTM_u)). \tag{8}$$

The $LSTM_u$ is conditioned on three variables. The first is the previous hidden state $\boldsymbol{q}_{u-1}$. The second is the ground truth character, y_{u-1}, which is extracted from an embedding layer, trained while training the full end-to-end network. At last, the third is the attention vector $\boldsymbol{c}_u$, which is concatenated with the previous character vector, giving a vector of size 2048 as input to the $LSTM_u$ cell.

For this architecture, two LSTM cells with 1024 units were used, where the new hidden state $\boldsymbol{q}_u$ is computed as:

$$\boldsymbol{q}_u = LSTM_u(\boldsymbol{q}_{u-1}, \boldsymbol{c}_u, y_{u-1}). \tag{9}$$

3.3. Hybrid CTC-attention network

In the hybrid CTC-attention architecture, the CTC and attention decoder networks share the same encoder. Also, when training, the CTC and attention loss are combined, to achieve more robustness and converge faster [12]:

$$Loss_{Total} = \lambda Loss_{CTC} + (1 - \lambda) Loss_{Attention}, \tag{10}$$

where $\lambda \in [0, 1]$. The λ value used for all end-to-end experiments is 0.2, the same as in [23].

3.4. Additional details

All noise and disfluencies from the TRIBUS corpus mentioned in Section 2 are mapped to a special token named <noise>. Also, it is important to note that there are special tokens for CTC and attention-based systems among all other output characters that exist, respectively. CTC requires a <blank> token [7], and the attention architectures requires the *start-of-sentence* and *end-of-sentence* (<sos/eos>) token. Therefore, the full hybrid CTC-attention system will have two special tokens plus an unknown token, <unk>, to map out-of-vocabulary (OOV) symbols. Finally, the total number of output symbols for TRIBUS is 49.

4. Experiments

For all end-to-end experiments, we used the second version of ESPnet toolkit [22] to implement and investigate our proposed methods, which is still under development by the time we write this paper. To evaluate the performance of our end-to-end ASR TRIBUS system, we compared it to a robust HMM-DNN baseline using the same corpus trained with Kaldi [19].

4.1. End-to-end experiments

The model was trained for at most 30 epochs, with early stopping (patience of 4 epochs) based on the validation loss. The training process takes approximately 9 hours in a single GeForce GTX 1080 Ti. Adadelta [24], an adaptive learning rate back-propagation algorithm, was the optimizer chosen, with an initial learning rate of 1.0, a mini-batch size of 30 and gradient clipping of 5. All weights were initialized using Xavier initialization [25]. It was also used a scheduler for the learning rate, where the scale factor was 0.5 and the patience 1 epoch. For data augmentation, we used speed perturbed factors of 0.9, 1.0 and 1.1, and SpecAugment [26]. For SpecAugment, F and T are set to 20 and 100 respectively, m_F and m_T are both set to 2, and W is set to 5. The decoding process of the hybrid CTC-attention model follows the setup in [22]. It is relevant to note that no language model was used when decoding with the end-to-end ASR system.

The word error rate (WER) results for the end-to-end TRIBUS ASR system, are presented in Table 2. From the results, we can see that BD-PÚBLICO has the lowest WER, mainly because it is read speech.

Table 2: *WERs [%] and CERs [%] on the end-to-end and HMM-based ASR systems, using TRIBUS corpus.*

	valid (WER)	test (WER)	test (CER)
HMM-GMM			
ALERT	33.26	34.89	-
BD-PÚBLICO	10.21	11.78	-
SPEECHDAT	8.42	13.49	-
HMM-DNN			
ALERT	9.69	9.65	4.33
BD-PÚBLICO	2.56	3.04	0.95
SPEECHDAT	2.49	4.86	3.26
End-to-end			
ALERT	18.80	19.40	8.40
BD-PÚBLICO	8.60	9.10	2.70
SPEECHDAT	21.20	20.00	8.40

4.2. HMM-based experiments

To create a robust HMM-based baseline for the TRIBUS corpus, we designed a similar procedure to the 's5' recipe of WSJ corpus, from Kaldi. First, to create the alignments for the HMM-DNN system, we trained an HMM-GMM system using the TRIBUS corpus, mentioned in Section 2. The training stages that created the HMM-GMM system are the following: (1) monophone stage, (2) triphones + delta + delta-delta stage, (3) triphones + LDA + MLLT stage and finally, (4) the triphones + SAT stage. For the first training stage (1), only the 2000 shortest utterances from the training set were used. For the second (2), a subset of 30000 utterances from the total of 92184, mentioned in Section 2, were used. Finally, for the last two training stages, (3) and (4), all utterances were used. The results for the last HMM-GMM system trained are depicted in Table 2. After creating the HMM-GMM system, the HMM-DNN system was trained following the Chain recipe from WSJ ("run_tdnn_1i.sh"), in Kaldi. The main difference is that only 12 layers were used to train the model, instead of 13 layers.

The WER results for the HMM-DNN ASR system, with respect to the TRIBUS corpus, are presented in Table 2. From results, we can notice that BD-PÚBLICO achieves the lowest WER. When comparing with the WERs from the end-to-end ASR system, we can observe that there is still a significant difference between the traditional HMM-DNN ASR systems and the end-to-end ASR systems for this low resourced scenario. In contrast to the end-to-end system, these results are obtained using the matched in-domain LMs described in section 2.2. In fact, the performance difference is more noticeable in the telephone data, for which the limited linguistic variability allows the LM to have a strongest impact. Finally, it is worth noticing that the baseline HMM system outperforms by a large margin the last reported result for ALERT [17], where we were able to decrease the absolute WER from 23.50% to 9.65%.

4.3. HMM-based vs end-to-end EP systems

For a better comparison between the end-to-end ASR and the HMM-DNN ASR systems, we also report results using CER in Table 2. We notice that the relative difference in terms of CERs of the proposed end-to-end system with respect to the HMM baseline are similar for the broadcast and read speech domain, but significantly better for the telephone speech domain. For instance, for ALERT, the performance degradation both in terms of WER and CER is approximately a factor of two, while for SPEECHDAT there is a factor of around 2.5 and 4 in terms of CER and WER, respectively. As pointed out previously, the LM seems to have a stronger impact in the telephone domain. Nevertheless, we can conclude that the HMM-based systems are still better than the end-to-end systems in this low resource setting.

5. Conclusions

In this paper, we presented the first known work using state-of-the-art end-to-end ASR systems for low resource EP. From the experimental results, and as it could be expected from the literature, we observe that the TRIBUS end-to-end performance is still far from the HMM-DNN system performance in such a low-resource setting. HMM-DNN systems have the advantage of using an in-domain language model that comprises almost all linguistic variation present in validation and test sets, and a pronunciation dictionary as well. Nonetheless, it is quite remarkable how the end-to-end hybrid CTC-attention systems can learn so much using less than 150 hours of training data and without any language model or pronunciation dictionary.

Overall, end-to-end ASR systems are known to have difficulties to generalize when in the presence of novel data [4], limiting its potential applicability to low resource scenarios. Thus, this problem needs to be specifically addressed in the future, for instance, based on a new architecture with better priors or, perhaps, considering new unsupervised learning algorithms able to create better representations.

6. References

[1] C. Pérez, Y. Campos-Roca, L. Naranjo, and J. Martín, "Diagnosis and tracking of parkinson's disease by using automatically extracted acoustic features," *J Alzheimers Dis Parkinsonism*, vol. 6, no. 260, pp. 2161–0460, 2016.

[2] J. Ivanecký and S. Mehlhase, "An in-car speech recognition system for disabled drivers," in *International Conference on Text, Speech and Dialogue*. Springer, 2012, pp. 505–512.

[3] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *PMLR*, 2014, pp. 1764–1772.

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

[5] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *ICASSP*, 2016, pp. 4960–4964.

[6] Z. Xiao, Z. Ou, W. Chu, and H. Lin, "Hybrid ctc-attention based end-to-end speech recognition using subword units," in *ISCSLP*, 2018, pp. 146–150.

[7] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *ICML*, 2006, pp. 369–376.

[8] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," *NIPS*, vol. 1, no. 9, pp. 577–585, 2015.

[9] A. Graves, "Sequence transduction with recurrent neural networks," *CoRR*, vol. abs/1211.3711, 2012.

[10] A. Y. Hannun, A. L. Maas, D. Jurafsky, and A. Y. Ng, "First-pass large vocabulary continuous speech recognition using bi-directional recurrent dnns," *arXiv preprint arXiv:1408.2873*, 2014.

[11] J. Chorowski, D. Bahdanau, K. Cho, and Y. Bengio, "End-to-end continuous speech recognition using attention-based recurrent nn: First results," *arXiv preprint arXiv:1412.1602*, 2014.

[12] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. 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.

[13] A. Rousseau, P. Deléglise, and Y. Estève, "Ted-lium: an automatic speech recognition dedicated corpus." in *LREC*, 2012, pp. 125–129.

[14] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," *ICASSP*, pp. 5206–5210, 2015.

[15] J. P. Neto, C. A. Martins, H. Meinedo, and L. B. Almeida, "The design of a large vocabulary speech corpus for portuguese," in *Fifth European Conference on Speech Communication and Technology*, 1997.

[16] D. B. Paul and J. Baker, "The design for the wall street journal-based csr corpus," in *Speech and Natural Language: Proceedings of a Workshop Held at Harriman*, 1992, pp. 899–902.

[17] H. Meinedo, D. Caseiro, J. Neto, and I. Trancoso, "Audimus. media: a broadcast news speech recognition system for the european portuguese language," in *International Workshop on Computational Processing of the Portuguese Language*, 2003, pp. 9–17.

[18] H. Hoge, H. S. Tropf, R. Winski, H. van den Heuvel, R. Haeb-Umbach, and K. Choukri, "European speech databases for telephone applications," in *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, 1997, pp. 1771–1774.

[19] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE Signal Processing Society*, 2011.

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

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2015.

[22] 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," *Proc. Interspeech*, pp. 2207–2211, 2018.

[23] S. Kim, T. Hori, and S. Watanabe, "Joint ctc-attention based end-to-end speech recognition using multi-task learning," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 4835–4839.

[24] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[25] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[26] 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.