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

Autoencoder bottleneck features with multi-task optimisation for improved continuous dysarthric speech recognition

Zhengjun Yue[1], *Heidi Christensen*[1,2] *and Jon Barker*[1]

[1]Speech and Hearing Group (SPandH), Dept. of Computer Science, University of Sheffield, UK
[2]Centre for Assistive Technology and Connected Healthcare (CATCH), University of Sheffield, UK
{z.yue,heidi.christensen,j.p.barker}@sheffield.ac.uk

Abstract

Automatic recognition of dysarthric speech is a very challenging research problem where performances still lag far behind those achieved for typical speech. The main reason is the lack of suitable training data to accommodate for the large mismatch seen between dysarthric and typical speech. Only recently has focus moved from single-word tasks to exploring continuous speech ASR needed for dictation and most voice-enabled interfaces. This paper investigates improvements to dysarthric continuous ASR. In particular, we demonstrate the effectiveness of using unsupervised autoencoder-based bottleneck (AE-BN) feature extractor trained on out-of-domain (OOD) LibriSpeech data. We further explore multi-task optimisation techniques shown to benefit typical speech ASR. We propose a 5-fold cross-training setup on the widely used TORGO dysarthric database. A setup we believe is more suitable for this low-resource data domain. Results show that adding the proposed AE-BN features achieves an average absolute (word error rate) WER improvement of 2.63% compared to the baseline system. A further reduction of 2.33% and 0.65% absolute WER is seen when applying monophone regularisation and joint optimisation techniques, respectively. In general, the ASR system employing monophone regularisation trained on AE-BN features exhibits the best performance.

Index Terms: continuous dysarthric speech recognition, autoencoder bottleneck features, multi-task optimisation

1. Introduction

Dysarthria is a speech disorder caused by a neuro-motor interface disruption [1]. People with dysarthria have poorer control of their articulators [2], and have difficulties with planning when trying to produce long sequences of words. This often causes heavily slurred speech, abnormal pauses, false starts and repetitions. As a result, there is a significant mismatch between dysarthric and typical speech, and a need to research approaches for automatic speech recognition (ASR) systems dedicated to dysarthric speech. Until now, most research has focused on the isolated word task because dysarthric speech datasets are not large enough to train *continuous* speech systems using conventional approaches. This paper investigates ways of addressing this problem by building an ASR system for dysarthria capable or learning from a large corpus of *typical* speech.

Previous studies have demonstrated the benefit of employing effective speech representations such as articulatory [3, 4] and bottleneck (BN) features [5, 4] to improve acoustic modeling of dysarthric speech. In particular, BN features have been shown to capture complementary information for dysarthric speech that can be beneficially fused with standard short-time spectral input features [5, 4]. Recently, there has been growing interest in *autoencoder-based* bottleneck features (AE-BNs) [6, 7]. In contrast to conventional BN features, extracted from a neural network bottleneck layer using a supervised criterion such as phoneme prediction accuracy [8], AE-BN features are learnt by reconstructing the input features in an unsupervised manner [6]. This makes them attractive for low-resource ASR tasks [9]. Although AEs have been applied for feature enhancement to improve dysarthric speech recognition by learning non-linear mappings from the dysarthric speech to the typical speech [10, 11], this has only been done using isolated-word dysarthric corpora such as UASpeech [12]. In addition, this approach is limited to corpora with parallel recordings for both typical and dysarthric speech. We propose to apply AE-BN features extracted using the reconstruction objective driven by the same input and output. This makes the approach applicable to a wider range of datasets and tasks.

The small amount of dysarthric training data limits the performance achievable using mainstream data-hungry ASR approaches designed for typical speech, for which training data is plentiful. However, exploiting out-of-domain (OOD) data has been shown to be beneficial in sparse data domains [13, 4, 14]. In particular, pretraining with OOD data can be especially crucial for speech feature extraction when little in-domain training data is available. The OOD typical-data pretraining framework was first introduced in [13] to boost the dysarthric speech representation learning process. Different BN extractor and acoustic model (AM) training strategies using both typical and dysarthric data were further investigated in [4]. They concluded that the best performance is achieved by training the BN feature extractor on a large amount of OOD typical speech while the AM is trained on the extracted dysarthric BN features.

In this work, we develop a benchmark for continuous dysarthric ASR system on TORGO [15], which has been proven to be the best database available for exploring continuous dysarthric ASR [16]. We firstly explore the effectiveness of employing an AE-BN feature extractor pretrained on OOD LibriSpeech [17] data to continuous dysarthric ASR. We then expand on this work by using two multi-task optimisation techniques (described in Section 2.3): i) *joint optimisation* [18] of the AE-BN feature extractor and the speech recogniser to learn better AE-BN features for dysarthric ASR, and ii) *monophone regularisation* [19] as an approach to strengthen the acoustic modeling (and hence the feature extractor, via joint optimisation). We evaluate our proposed models on the sentence subset of TORGO using an independent trigram language model (LM) trained on LibriSpeech. To the best of our knowledge, this is the first paper to demonstrate the effectiveness of multi-task optimisation techniques in the dysarthric speech domain.

http://dx.doi.org/10.21437/Interspeech.2020-2746

2. Background

2.1. Autoencoder-based bottleneck feature extractor

An AE is an unsupervised way to learn a compact data representation [20], consisting of an encoder and a decoder. The encoder encodes the high dimensional input feature vector into a lower-dimensional latent variable (in the following called an AN-BN feature). The decoder reconstructs the input using the generated latent variable. The AE-BN feature is driven by two opposing constraints: i) the reconstruction objective which forces the AE-BN feature to capture as much of the input data characteristics as possible, and ii) the bottleneck (i.e., the dimension reduction) which forces the network to discard the redundant information that is not needed for the inversion.

Note, whereas an autoencoder trained on a small amount of dysarthric speech data would be prone to overfitting, an autoencoder trained on typical speech may not be optimal for representing dysarthric signals. Further, without suitable regularisation the encoder may form an inefficient representation by capturing information relevant for signal reconstruction but not important for phoneme classification (e.g., speaker variability, pitch). We attempt to address these potential deficiencies using a multi-task learning optimisation described in Section 2.3.

2.2. Acoustic model architecture

Dysarthric ASR performance improvements have been made by exploring various deep neural network (DNN) architectures such as CNNs, TDNNs and LSTMs [21, 22, 23, 24] in the past few years. Recently, Light Gated Recurrent Units (LiGRU) [25] have been shown to outperform existing architecture on large typical speech datasets such as LibriSpeech and TIMIT [26]. It is widely used in Pytorch-Kaldi's ASR framework [27]. As an advanced Recurrent Neural Network (RNN), the LiGRU model has the capability to exploit large time contexts and to capture long-term speech modulations. Compared with the commonly-used LSTMs [28], LiGRUs have a simpler cell design that allows for faster training. The design also avoids the numerical issue of learning long-term dependencies and mitigates the vanishing gradient problem by employing Rectified Linear Unit (ReLU) activation with batch normalisation.

As the LiGRU model has not been used for continuous dysarthric ASR, we tested it on the sentence subset of TORGO by keeping the same experimental settings as in [22, 16] except for replacing the AM with the LiGRU model. We found that the performance achieved by the LiGRU AM in Pytorch-Kaldi is comparable to other AMs trained in Kaldi presented in previous papers. For instance, the TDNN model achieves 70.72% WER averaged across all speakers, while LiGRU achieves 71.08%. For speakers with severe dysarthria, the LiGRU model performs even better (83.90% VS. 86.40%). These comparable results are achieved in Pytorch-Kaldi without the benefits of the (computationally expensive) lattice-free maximum mutual information training used in the Kaldi systems. We therefore employ the LiGRU acoustic model in the remainder of this work.

2.3. Multi-task optimisation

Two optimisation techniques are introduced: i) a joint optimisation strategy for training the integrated network (feature extractor and AM) with a multi-task training criterion, and ii) a monophone regularisation applied to the AM.

2.3.1. Joint optimisation

The feature extractor and speech recogniser are often designed independently. This means that the feature extractor is tuned according to a criteria which is not directly related to ASR performance. Recently, DNNs have made the integration of various components of a typical ASR system possible. In [18] a DNN-based integrated network for distant speech recognition was proposed that combined speech enhancement and speech recognition modules are allowed for the joint updating of parameters. It was shown that this yields better results than training each part separately. They also demonstrated that a pretraining strategy with a fine-tuning phase improves performance. In this paper, we explore a similar approach and evaluate the effect of jointly optimising the AE-BN feature extractor and the speech recogniser where the feature extractor is pretrained on LibriSpeech data and fine-tuned using TORGO dysarthric data. The core idea of joint training is that the feature extractor should provide more discriminative representations for the ASR task as it is in part guided by the speech recognition cost function [18]. In this case, the speech recognition gradient is also backpropagated through the feature extraction module.

2.3.2. Monophone regularisation

To train a better AM, the multi-task learning (MLT) technique has been applied to hybrid DNN systems in [19]. They added a secondary task of predicting alternative context-dependent (CD) (i.e., triphone) or context-independent (CI) (i.e., monophone) targets. Consistent improvements have been shown over the standard single target training approach on large-vocabulary typical speech recognition tasks. In our case, the two tasks are jointly estimated by using a weighted sum cost function between the two predictions from the two softmax classifiers. Importantly, this strategy does not require additional data making it suitable for our low-resource data domain. This MTL scheme can be regarded as a technique to regularise the AM, preventing it from over-fitting to a single senone target classification by learning additional CI or CD labels. This encourages a better presentation of the data to be learnt by the AM (and by extension, by the auto-encoder when joint optimisation is engaged).

3. Experiments

3.1. Data description and training setup

TORGO is one of the few available dysarthric speech datasets and has been widely used. It contains aligned acoustic and articulatory recordings collected from 15 speakers. Eight of the speakers (5 males, 3 females) have different dysarthric severity, while the other seven are typical speakers (4 males, 3 females). The acoustic data is recorded by a head-mounted as well as a single directional microphone, simultaneously. TORGO comprises both word and sentence prompts: 615 unique words and 354 unique sentences with a total vocabulary size of 1573.

Since TORGO does not come with a pre-defined training and test partition, we applied an *N-fold cross-training* setup, with the total dataset (including all speakers) being divided into five folds (i.e., one fifth of each speaker in every fold)[1]. This maximises the available training and test data while maintaining the need for disjoint training and test sets. Table 1 summarises the duration of the recordings in each fold (after excluding the recordings that are shorter than 25 ms and any wrongly anno-

[1]The pre-defined training and test partition set is available at `https://github.com/zhengjunyue/bntg`.

tated audio). The ratio of the duration of the two utterance type subsets (isolated word vs sentence) is about 1.5:1.1.

We have noticed that most of the previous TORGO-based work used the *leave-one-speaker-out* (LOSO) approach to train speaker-independent (SI) models [21, 22, 24]. With only 8 speakers, there are insufficient speakers in TORGO to capture the wide inter-speaker variability observed in dysarthria. In a LOSO SI setting, speaker performances will be more determined by the chance degree of matched-ness of the target speaker to the few others in the training set, i.e., rather than to any intrinsic difficulty of the speech itself. Our 5-fold approach ensures a good trade-off between having a reasonably large training set, while providing some matched speaker training data to allow for more meaningful comparison of recognition performance across speakers.

Table 1: *Duration (hours) of the training and test data in each fold using the 5-fold cross-training setup*

subset	fold 1	fold 2	fold 3	fold 4	fold 5
train_all	10.71	10.69	10.71	10.83	10.57
train_sentence	4.63	4.54	4.60	4.71	4.59
train_word	6.10	6.15	6.11	6.12	6.16
test_all	2.71	2.73	2.72	2.59	2.67
test_sentence	1.14	1.22	1.17	1.06	1.18
test_word	1.57	1.51	1.55	1.53	1.49

3.2. Architecture

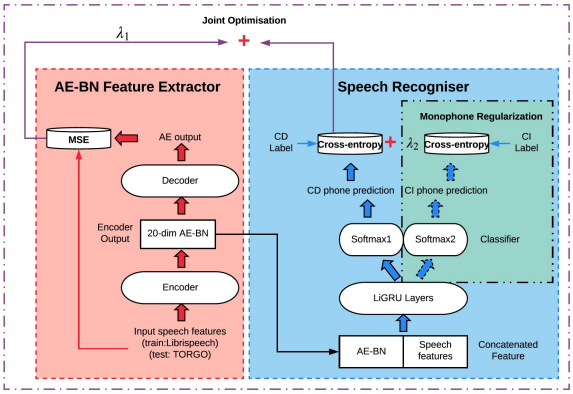

Figure 1: *Architecture of the proposed models.*

Figure 1 depicts the architecture of the proposed AE-BN with multi-task optimisation dysarthric ASR system. The red box on the left shows the feature extractor and the blue box on the right represents the acoustic model. First, the AE-BN feature extractor is trained on the 100-hour subset of LibriSpeech corpus, which is a large typical read speech dataset. The dysarthric AE-BN features extracted from the encoder output are then concatenated with the input acoustic features and fed into the acoustic model. The parameters of AE are updated by minimising the mean square error (the reconstruction error) calculated between the reconstructed input data y_i and the true input data x_i:

$$Loss_{AE} = (\frac{1}{n}) \sum_{i=1}^{n} (y_i - x_i)^2 \quad (1)$$

The acoustic model includes LiGRU-based layers followed by a softmax layer as a classifier. The classifier estimates the standard CD states and calculates the cost function (cross-entropy loss ($Loss_{CD}$) between the CD labels and the predictions.

In addition to training the AE-BN feature extractor and the speech recogniser separately, we explore an integrated network where these two parts are jointly optimised. The recently proposed PyTorch-Kaldi framework provides a platform to implement the joint optimisation which would be difficult to perform in the Kaldi [29] toolkit. The parameters are updated by back-propogating a weighted sum of the AE reconstruction loss and the cross-entropy loss,

$$Loss_{Joint} = \lambda_1 * Loss_{AE} + Loss_{ASR} \quad (2)$$

where λ_1 controls the trade-off between the reconstruction quality of the feature extractor and the effectiveness of the speech recogniser.

We also applied multi-task regularisation to the AM, using monophone classification as a secondary task by adding another softmax classifier to estimate the CI states. The joint optimisation cost function becomes the sum of the $Loss_{CD}$ and the cross-entropy loss $Loss_{CI}$ between the true CI labels and the predictions:

$$Loss_{ASR} = Loss_{CD} + \lambda_2 * Loss_{CI} \quad (3)$$

where λ_2 indicates the weighting between each task's loss.

3.3. Experimental Setup

The training data was augmented using speed perturbation (using factors 0.9, 1.0 and 1.1). We used 40-dimensional feature-space maximum likelihood linear regression (fMLLR) transformed features [30] with splicing of 11 contextual frames (i.e., a total dimensionality of 440) as the inputs of the AE-BN feature extractor. The encoder consists of four layers. The first two layers are convolution layers with filter length 3 and ReLU activation to allow rich local representations. The last two layers are feed-forward ReLU layers with 768 units and 20 units to encode the input features into a 20-dimensional representation. The decoder comprises two ReLU layers fed by the learned AE-BN features and aims to produce an output matching the 440 dimensional input.

The LiGRU-based acoustic model follows the design from [25], containing five stacked bidirectional LSTM layers [31] and a final softmax classifier. Recurrent dropout (0.15) is used as a regularisation technique. The minibatch sizes are 128 and 16 for the AE-BN feature extractor and the acoustic model, respectively. Stochastic gradient descent (SGD) optimisation is used in the feature extractor and RMSProp [32] in the LiGRU model. Learning-rate annealing is applied with a factor of 0.5. When setting up the evaluation framework, we employed a 200k vocabulary size LibriSpeech trigram LM as in [16]. To reflect the diversity of the data as best as possible, we compute and report results on individual speakers.

4. Results and Discussion

Results are shown in Table 2. The first row displays the baseline system using just the LiGRU AM trained on 39-dimensional MFCC feature and without using the AE-BN feature extractor. The MFCC features were then substituted with the 40-dimensional fMLLR features (second row). It is seen that the speaker adapted fMLLR features outperform the baseline MFCCs reducing WER by 3% for moderately and severely

Table 2: *ASR performance [WER] using different speech representations and AMs for per (F)emale or (M)ale speaker with different dysarthria severity, and the averaged result of all speakers 'M/S': moderate to severe level of dysarthria.*

	Severe				**M/S**	**Moderate**	**Mild**		
Features used in the models	F01	M01	M02	M04	M05	F03	F04	M03	**Average**
MFCC	77.93	77.91	76.17	91.66	85.46	51.47	22.27	22.04	59.22
fMLLR	73.86	76.36	73.12	88.66	83.74	49.18	21.71	21.69	57.33
fMLLR+BN20	69.84	71.55	72.26	85.97	78.9	47.06	19.75	19.86	54.70
fMLLR+BN20 + mono	71.47	69.3	70.88	**79.91**	**77.18**	**44.21**	**18.26**	**18.23**	**52.37**
fMLLR+BN20 + joint	**69.29**	70.54	71.65	83.37	80.4	47.74	19.5	19.65	54.05
fMLLR+BN20 + mono + joint	70.65	**69.07**	**70.81**	81.82	78.4	45.18	18.42	19.15	52.99

dysarthric speakers. Therefore, we continued to use fMLLR features as the input in the following experiments.

When introducing the AE-BN feature extractor, we first explored the optimal dimensionality of the AE-BN features since the recognition loss depends on the width of the bottleneck. It was found that the best recognition performance arose using a dimensionality of 20, with results reported in the third row of Table 2. Introducing the AE-BN features reduced WER by a further 1.77% to 4.84% absolute.

Further improvements are made by applying multi-task optimisation techniques. Comparing rows 3 and 4 in Table 2, the AM regulariser successfully reduces WER by an absolute 2.33% across speakers. For speakers with severe dysarthria, the WERs are decreased by 1.83% to 6.06% with the exception of speaker F01. For speakers with moderate dysarthria, there is also a 2.85% recognition performance improvement. This indicates that a single set of triphone targets is not optimal for the discriminative clustering process. The additional CI label learning step strengthens the dysarthric acoustic model. When tuning the jointly optimised model, different values of λ_1 (Eq. 2) ranging from 0.1 to 1 were tested with 0.2 producing the best ASR performance. Comparing the third and the fifth rows in Table 2, the joint optimisation technique achieves a WER reduction of 0.65% absolute compared to the model that trains the feature extractor and acoustic model separately.

The "BN20+fMLLR + mono + joint" in the last row in Table 2 is a model that applies the joint optimisation technique to the AM with monophone regularisation. Comparing the last three rows shows that the monophone regularisation technique provides a further improvement on the joint optimisation model and vice versa except for some speakers with severe dysarthria. Almost all the benefits seen in the last row are coming from monophone regularisation, therefore it appears that the joint optimisation provides no significant benefit when coupled with a sufficiently strong AM. The possible reason is that the joint training was actually performed as a fine-tuning procedure, and the hyperparameters such as learning rate need to be selected properly to take advantage of the pretraining. Although the joint optimisation did not provide the benefits expected, it remains an under-explored research direction deserving of further investigation. The overall best result (52.37% WER) is obtained when employing monophone regularisation alone.

The results show that achieving an acceptable performance for a continuous dysarthric speech recogniser remains challenging. This is exacerbated by the fact that some speakers with dysarthria produce many repetitions and false starts when having to speak in full sentences. Figure 2 illustrates this. It shows WERs for not just the TORGO sentence task, but also for the isolated word task and the full, combined test set across all speakers. In general, and as expected, the sentence task is harder for everyone, however, for some speakers (e.g., M04 and M05) the sentence performances are much worse. Inspection of the audio confirmed that the ASR transcription had many insertions caused by disfluencies typical for speakers with dysarthria.

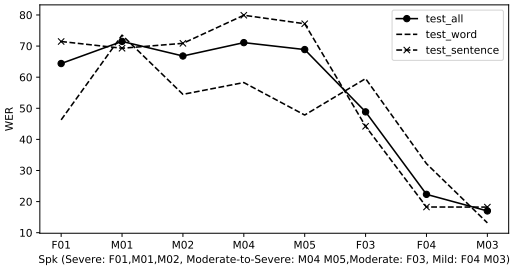

Figure 2: *The ASR performance [WER] for different utterance subsets using the proposed fMLLR+BN20+mono model*

5. Conclusions

We investigated how autoencoder-based bottleneck features (AE-BN) trained on typical speech can be used to improve the performance of a continuous dysarthric ASR system. Using the TORGO dysarthric speech database, we demonstrated that augmenting conventional acoustic features with features extracted by an AE-BN extractor pretrained on typical speech reduces WERs by 2.63% absolute on average. A further 2.33% and 0.65% absolute recognition improvements were achieved by exploiting two multi-task optimisation techniques: monophone regularisation and joint optimisation. However, the joint optimisation technique provided no consistent additional benefit when applied in conjunction with monophone regularisation. The best performance is achieved by the AE-BN feature model applying monophone regularisation with an average absolute WER improvement of 4.96% over the baseline system. Future work will focus on exploring more advanced AEs to produce better AE-BN features, and fine-tuning the joint optimisation technique. In addition, we will investigate how to incorporate the real articulatory dysarthric data available in the TORGO dataset in the pretrained AE-BN extractor.

6. Acknowledgements

This work is supported under the European Union's H2020 Marie Skłodowska-Curie programme TAPAS (Training Network for PAthological Speech processing; Grant Agreement No. 766287).

7. References

[1] W. Gowers, "Clinical speech syndromes of the motor systems," *Webb WG, Adler RK, Love RL. Neurology for the Speech-Language Pathologist. Fifth edition. Philadelphia: Butter worth_ Heinemenn*, pp. 196–203, 2001.

[2] R. D. Kent, K. Rosen, and B. Maassen, "Motor control perspectives on motor speech disorders," *Speech motor control in normal and disordered speech*, pp. 285–311, 2004.

[3] F. Xiong, J. Barker, and H. Christensen, "Deep learning of articulatory-based representations and applications for improving dysarthric speech recognition," in *Speech Communication; 13th ITG-Symposium*. VDE, 2018, pp. 1–5.

[4] E. Yılmaz, V. Mitra, G. Sivaraman, and H. Franco, "Articulatory and bottleneck features for speaker-independent asr of dysarthric speech," *Computer Speech & Language*, vol. 58, pp. 319–334, 2019.

[5] Y. Takashima, T. Nakashika, T. Takiguchi, and Y. Ariki, "Feature extraction using pre-trained convolutive bottleneck nets for dysarthric speech recognition," in *2015 23rd European Signal Processing Conference (EUSIPCO)*. IEEE, 2015, pp. 1411–1415.

[6] T. N. Sainath, B. Kingsbury, and B. Ramabhadran, "Auto-encoder bottleneck features using deep belief networks," in *2012 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2012, pp. 4153–4156.

[7] J. Chorowski, R. J. Weiss, S. Bengio, and A. van den Oord, "Unsupervised speech representation learning using wavenet autoencoders," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 27, no. 12, pp. 2041–2053, 2019.

[8] F. Grezl and P. Fousek, "Optimizing bottle-neck features for lvcsr," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2008, pp. 4729–4732.

[9] S. Thomas, M. L. Seltzer, K. Church, and H. Hermansky, "Deep neural network features and semi-supervised training for low resource speech recognition," in *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013, pp. 6704–6708.

[10] B. Vachhani, C. Bhat, B. Das, and S. K. Kopparapu, "Deep autoencoder based speech features for improved dysarthric speech recognition." in *Interspeech*, 2017, pp. 1854–1858.

[11] C. Bhat, B. Das, B. Vachhani, and S. K. Kopparapu, "Dysarthric speech recognition using time-delay neural network based denoising autoencoder." in *Interspeech*, 2018, pp. 451–455.

[12] H. Kim, M. Hasegawa-Johnson, A. Perlman, J. Gunderson, T. S. Huang, K. Watkin, and S. Frame, "Dysarthric speech database for universal access research," in *Ninth Annual Conference of the International Speech Communication Association*, 2008.

[13] H. Christensen, M. Aniol, P. Bell, P. D. Green, T. Hain, S. King, and P. Swietojanski, "Combining in-domain and out-of-domain speech data for automatic recognition of disordered speech." in *INTERSPEECH*, 2013, pp. 3642–3645.

[14] F. Xiong, J. Barker, Z. Yue, and H. Christensen, "Source domain data selection for improved transfer learning targeting dysarthric speech recognition," in *Proceedings of the 45th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2020)*. IEEE, 2020.

[15] F. Rudzicz, A. K. Namasivayam, and T. Wolff, "The torgo database of acoustic and articulatory speech from speakers with dysarthria," *Language Resources and Evaluation*, vol. 46, no. 4, pp. 523–541, 2012.

[16] Z. Yue, F. Xiong, H. Christensen, and J. Barker, "Exploring appropriate acoustic and language modelling choices for continuous dysarthric speech recognition," in *Proceedings of the 45th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2020)*. IEEE, 2020.

[17] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[18] M. Ravanelli, P. Brakel, M. Omologo, and Y. Bengio, "Batch-normalized joint training for dnn-based distant speech recognition," in *2016 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2016, pp. 28–34.

[19] P. Bell, P. Swietojanski, and S. Renals, "Multitask learning of context-dependent targets in deep neural network acoustic models," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 2, pp. 238–247, 2016.

[20] C.-Y. Liou, W.-C. Cheng, J.-W. Liou, and D.-R. Liou, "Autoencoder for words," *Neurocomputing*, vol. 139, pp. 84–96, 2014.

[21] K. T. Mengistu and F. Rudzicz, "Adapting acoustic and lexical models to dysarthric speech," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 4924–4927.

[22] C. Espana-Bonet and J. A. Fonollosa, "Automatic speech recognition with deep neural networks for impaired speech," in *International Conference on Advances in Speech and Language Technologies for Iberian Languages*. Springer, 2016, pp. 97–107.

[23] M. Kim, B. Cao, K. An, and J. Wang, "Dysarthric speech recognition using convolutional lstm neural network," *Proc. Interspeech 2018*, pp. 2948–2952, 2018.

[24] E. Hermann *et al.*, "Dysarthric speech recognition with lattice-free mmi," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, no. CONF, 2020.

[25] M. Ravanelli, P. Brakel, M. Omologo, and Y. Bengio, "Light gated recurrent units for speech recognition," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 2, pp. 92–102, 2018.

[26] V. Zue, S. Seneff, and J. Glass, "Speech database development at mit: Timit and beyond," *Speech communication*, vol. 9, no. 4, pp. 351–356, 1990.

[27] M. Ravanelli, T. Parcollet, and Y. Bengio, "The pytorch-kaldi speech recognition toolkit," in *In Proc. of ICASSP*, 2019.

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

[29] 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," IEEE Signal Processing Society, Tech. Rep., 2011.

[30] D. Povey and G. Saon, "Feature and model space speaker adaptation with full covariance gaussians," in *Ninth International Conference on Spoken Language Processing*, 2006.

[31] A. Graves, N. Jaitly, and A.-r. Mohamed, "Hybrid speech recognition with deep bidirectional lstm," in *2013 IEEE workshop on automatic speech recognition and understanding*. IEEE, 2013, pp. 273–278.

[32] T. Tieleman and G. Hinton, "Rmsprop: Divide the gradient by a running average of its recent magnitude. coursera: Neural networks for machine learning," *COURSERA Neural Networks Mach. Learn*, 2012.