Word Error Rate Estimation Without ASR Output: e-WER2

Ahmed Ali[1], *Steve Renals*[2]

[1]Qatar Computing Research Institute, HBKU, Doha, Qatar
[2]Centre for Speech Technology Research, University of Edinburgh, UK

amali@hbku.edu.qa, s.renals@ed.ac.uk

Abstract

Measuring the performance of automatic speech recognition (ASR) systems requires manually transcribed data in order to compute the word error rate (WER), which is often time-consuming and expensive. In this paper, we continue our effort in estimating WER using acoustic, lexical and phonotactic features. Our novel approach to estimate the WER uses a multistream end-to-end architecture. We report results for systems using internal speech decoder features (glass-box), systems without speech decoder features (black-box), and for systems without having access to the ASR system (no-box). The no-box system learns joint acoustic-lexical representation from phoneme recognition results along with MFCC acoustic features to estimate WER. Considering WER per sentence, our no-box system achieves 0.56 Pearson correlation with the reference evaluation and 0.24 root mean square error (RMSE) across 1,400 sentences. The estimated overall WER by e-WER2 is 30.9% for a three hours test set, while the WER computed using the reference transcriptions was 28.5%.

Index Terms: word error rate estimation. multistream, end-to-end

1. Introduction

Automatic Speech Recognition (ASR) has accomplished great success, primarily due to advances in the end-to-end neural networks and the modular hybrid HMM-DNN architectures. As a result, the quality of ASR has improved dramatically, leading to growing adoption in personal assistant devices, smart phones and broadcast media monitoring. Despite this progress, ASR performance is still closely tied to how well the training data matches the test conditions, such as the variability of different microphones or background noises. While researchers in [1, 2] discussed achieving human parity on conversational speech, recent competitions in ASR [3, 4] reported considerably poorer results due to dialectal speech, simultaneous recordings from multiple microphone arrays, and background noise.

Word Error Rate (WER) is the standard approach to evaluate the performance of a large vocabulary continuous speech recognition (LVCSR) system. To obtain a reliable estimate of the WER, at least two hours of manually transcribed test data is typically required – a time-consuming and expensive process. It is, thus, of interest to develop techniques which can automatically estimate the quality of the ASR.

Such quality estimation techniques have been extensively investigated for machine translation [5, 6, 7], with extensions to spoken language translation [8, 9]. Although there is a long history of exploring word-level confidence measures for speech recognition [10, 11, 12, 13, 14, 15, 16, 17], there has been fewer attempts on the direct estimation of speech recognition errors [18, 19].

Previously, we proposed e-WER [20], a method to estimate the total number of errors per utterance ($E\hat{R}R$) and the total number of words in the reference ($\hat{N}$) as shown in section 2.1. However, that work assumed having access to a graphemic speech recognition for the predicted language and being able to see the ASR transcription . In this paper, we extend this work by deploying an end-to-end multistream architecture to predict the WER per sentence using language-independent phonotactic features. Our novel system is able to learn acoustic-lexical embeddings to estimate the error rate directly without having access to the ASR results nor the ASR system – this is our "no-box" WER estimation method, e-WER2[1].

2. Related Work

Several studies have explored estimating the WER in LVCSR. TranscRater [21, 22, 23, 24, 25] estimated the WER per utterance using a large set of extracted features (not including ASR decoder features) to train a regression model (e.g., extremely randomised trees). This work did not report WER estimates for complete recordings or test sets, although it is possible that this could be done using utterance length estimates.

Fan et al [26] proposed a novel neural zero-inflated model to predict the WER of the ASR result without transcripts. They deployed a bidirectional transformer language model conditional on speech features (speechBERT). They adopted the pre-training strategy of token level mask language modeling for speech-BERT as well, and further fine-tune with zero-inflated layer for the mixture of discrete and continuous outputs. They reported results in WER prediction using the metrics of Pearson correlation and mean absolute error (MAE).

Vyas et al [27] used dropout in a novel framework to model uncertainty in prediction hypotheses. They systematically exploited this uncertainty in the output of the acoustic models through the Monte Carlo sampling of the neural networks using dropout at the test time. They were able to estimate the WER without the need for explicit transcription. However, the models must have access to the ASR models to model the uncertainty in the prediction.

2.1. e-WER

In this section, we give a brief overview of our previous e-WER framework [20]. We used two speech recognition systems; a word-based LVCSR system and a grapheme-sequence based system. Following [28], we assumed that when two corresponding ASR systems disagree on a sentence or part of a sentence, there is a pattern of the error to be learned. The e-WER architecture also benefits from utterance-based LVCSR internal decoder features. The e-WER approach is looking for the overall error pattern and not particularly concerned with the

[1]https://github.com/qcri/e-wer

error. We directly estimated the numerator in the WER, which is the summation of insertion, deletion and substitution errors, which we refer to as $E\hat{R}R$, the estimated total number of errors per utterance. We also directly estimated $\hat{N}$, an estimate of the total number of words in the reference as shown in 1. The e-WER predicts two values for each utterance: $E\hat{R}R$ and $\hat{N}$. More details about the e-WER can be found here [20]. We use the e-WER system as a baseline reference for this paper.

$$\text{e-WER} = \frac{E\hat{R}R}{\hat{N}} \times 100\% \quad (1)$$

3. e-WER2 Framework

In this paper, we develop models to predict the WER per sentence rather than $E\hat{R}R$ or $\hat{N}$. WER per sentence can be scaled by the corresponding sentence duration to calculate the overall e-WER2.

3.1. Features

We combine features from the word-based LVCSR system with features from the phoneme-based system. We split the studied features into the following four groups:

- *L*: lexical features – the word sequence extracted from the LVCSR;
- *P*: phoneme features – the phonotactic sequence extracted from the phoneme recognition, see 3.1.1;
- *D*: decoder features – total frame count, average log-likelihood, total acoustic model likelihood, and total language model likelihood; and
- *A*: acoustics features – the MFCC features are extracted by segmenting each utterance into 25 ms long frames with a 10 ms shift. A Hamming window is applied and the FFT with 512 points is computed. Then, we compute the logarithmic power of 26 Mel-frequency filter-banks over a range from 0 to 8 kHz. Finally, a discrete cosine transform (DCT) is applied to extract the first 13 MFCCs.

3.1.1. Phonotatic features

In our phonotactic systems, we use Arabic and non-Arabic phone recognizers. For the Arabic recogniser, the HMM-LSTM based acoustic model is trained using 1,200 hours of training data from the MGB-2 datasets [29] . In addition to the Arabic recognizer, we used a phone recognizer from a toolkit developed by Brno University of Technology [30], trained on Hungarian, but empirically observed to be applicable to multiple languages. This phone recogniser is based on a long temporal context, and has been widely used to discriminate between various languages and dialects. The intuition for using this system is that a robust phone recogniser is capable of extracting an accurate phonotactic pattern for the recognised language. We benchmarked Hungarian results against Arabic and the results were similar, thus we decided to deploy this phone recognition system for the multilingual extraction of phonotactic features.

3.2. Modelling

In this study, we consider four different streams; numerical (*D*), lexical (*L*), phonotactic (*P*) and finally acoustic features (*A*).

*3.2.1. Numerical modelling: (*D*)*

We deploy a feed-forward neural network for the numerical features with fully-connected hidden layers (ReLU activation function), with 64 neurons in the first layer and 32 neurons in the second layer followed by a softmax layer with mean squared error loss function. We use dropout rate between layers 0.2, minibatch size of 32 and the number of epochs was up to 50 with an early stopping criterion.

*3.2.2. Acoustic modelling: (*A*)*

We employed deep CNN models, each with five layers where four of them are CNN layers. The same dropout rate, batch size and number of epochs was used as above. Table 1 shows details of the deployed models. More details about this model can be found here [31].

Table 1: *The acoustic features deep CNN architecture.*

Layer	Type	Details
1	Conv	500 filters + Relu + Stride=1 + kernal wdith=5
2	Conv	500 filters + Relu + Stride=2 + kernal wdith=7
3	Conv	500 filters + Relu + Stride=2 + kernal wdith=1
4	Conv	500 filters + Relu + Stride=1 + kernal wdith=1
5	MaxPool1D	Global

*3.2.3. Textual modelling: (*L *and* P*)*

We use CNN models for phoneme and text processing. The input word sequences are trimmed to a maximum of 100 words for the long sentences, and we padded shorter sentences with zeros. This was followed by an embedding layer of a dimension of 256. Followed by three convolutional layers in parallel to each other with the same number of filters: 512 each, and ReLU activation function. The filters' sizes were different for each convolution layer: 3, 4 and 5, respectively. The three-convolutional layers were then merged into a single tensor. The same CNN is used for the phoneme sequence. However, a maximum of 200 dimensions were used for phoneme-based CNN. More details about this model can be found here [32].

3.2.4. Multistream system:

We combine the four streams: lexical, phonotactic, acoustics and numerical features into a single end-to-end network to estimate word error rate directly. We jointly train the multistream network and their final hidden layers are concatenated to obtain a joint feature space in which another fully connected layer with 32 neurons and Relu activation function to estimate the WER directly. Figure 1 shows the architecture of the multistream approach developed in this paper.

4. Speech Recognition System

The LVCSR system is trained using the second Multi-Genre Broadcast challenge data, MGB-2 [33]. The data comprised recorded programs over 10 years of the Aljazeera Arabic TV channel with a total of 1,200 hours of audio that could be used

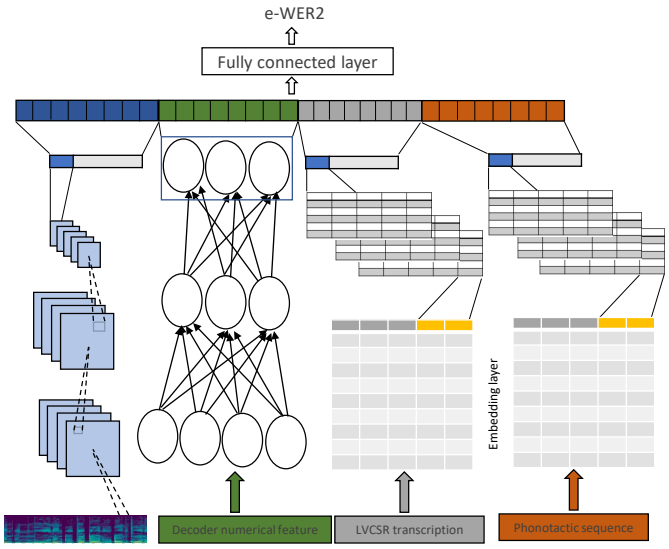

Figure 1: *Multistream model architecture of e-WER2. Based on a combination of four features: decoder, acoustics, textual and phonotactics features.*

Table 2: *Data used for acoustic model training, development and evaluation*

Type	Hours	Programs	#segments
Training	1,200h	2,214	370K
Development	10h	17	5,800
Evaluation	10h	17	5,600

Table 3: *In-domain data refers to the training transcripts and Background data refers to the extra Arabic language modeling text provided for the challenge*

Type	Tokens	Vocab
In-domain	8M	200k
Background	130M	1M

for the acoustic model (AM). The original transcription has no timing information and is not verbatim, having been generated as closed captions for viewers; the quality of the transcription varies significantly. We, therefore, use lightly supervised alignment algorithms in order to recover the timing information for each word.

For language modelling, we use 130M words crawled from the Aljazeera Arabic website from the period 2000–2011 (background text), as provided for the MGB-2 challenge. We have used the provided Buckwalter[2] format for the transcription as well as for the background text.LM experiments used a grapheme lexicon of 1.3M words. The grapheme based lexicon has a 1:1 word-to-grapheme mapping, which means the vocabulary size is the same as the lexicon size. More details about the data can be found in tables 2 and 3.

Acoustic modelling: There are many architectures used for the hybrid HMM neural acoustic modeling, with a recent trend in ASR modeling combining different types of layers. Peddinti et al [34] explored using dropout to improve generalisation in DNN training. They reported that combining a time delay neural network (TDNN) with long short term memory (LSTM) layers outperformed bidirectional LSTM (BLSTM) acoustic modelling. We adopt this architecture. The TDNN-LSTM model consists of 5 hidden layers, each layer containing 1,024 hidden units. We use purely sequence trained neural networks using lattice-free maximum mutual information (LF-MMI) [35]. Acoustic models are built using Kaldi ASR toolkit [36].

Language modelling: We train two *n*-gram LMs: a big four-gram LM (bLM4), trained using the spoken transcripts and the background text as shown in table 3; and a smaller four-gram LM (sLM4) obtained by pruning bLM4 using pocolm[3]. The small LM is used for first-pass acoustic decoding to generate lattices. These lattices are then rescored using the bLM4.

5. Data

In our study, we use the same data as [20] to benchmark our results. The e-WER2 training and development data sets are the same as the Arabic MGB-2 development and evaluation sets [33], which is comprised of audio extracted from Al-Jazeera Arabic TV programs recorded by Brightcove in the last months of 2015. They each comprise 10 hours of audio that were not used in the MGB-2 training data. (Other episodes of the same program may have been included in the training set). To test whether our approach generalises to test sets from a different source, and not tuned to the MGB-2 data set, we validate our results on another three hours test set collected by BBC Media Monitoring from different broadcasters during November 2016, as part of the SUMMA project [4]. The SUMMA data is referred to as the test set. All data were manually segmented and labeled. Table 4 shows more details about the data used for these experiments.

[2]Buckwalter is a one-to-one mapping allowing non-Arabic speakers to understand Arabic scripts, and it is also left-to-right, making it easy to render on most devices.

[3]https://github.com/danpovey/pocolm

[4]http://summa-project.eu

Table 4: *Analysis of the train, dev and test data.*

	Train	Dev	Test
# of programs in corpus	17	17	24
Utterances	5.6K	5.8K	1.4K
Duration (in hours)	10.2	9.9	3.2
2-20 words sentences	95%	96%	96%
Word count (N)	**69K**	**75K**	**20K**
ASR word count (hyp)	60K	58K	18K
WER	**33.1%**	**42.6%**	**28.5%**
Sentence Error Rate (SER)	89.1%	88.7%	86.0%
Total INS	1.8K	1.9K	130
Total DEL	10.2K	19.1K	2.6K
Total SUB	10.8K	11.1K	2.9K
ERR count (ERR)	**22.8K**	**32.1K**	**5.7K**

Table 5: *Pearson correleation and RMSE report per system. The overall WER reported in %, the reference overall WER is 28.5%*

	WER Per Sentence		**Overall WER**
	Pearson	$RMSE$	e-WER
Glass-box baseline	0.8	0.17	26.5
Black-box baseline	0.66	0.35	28.6
$\mathcal{A}$	0.79	0.2	28.0
$\mathcal{B}$ e-WER2 glass-box	**0.81**	**0.18**	**27.7**
$\mathcal{C}$	0.68	0.22	35.3
$\mathcal{D}$ e-WER2 black-box	**0.72**	**0.2**	**22.4**
$\mathcal{E}$	0.11	0.28	46.1
$\mathcal{F}$ e-WER2 no-box	**0.56**	**0.24**	**30.9**

6. Experiments and discussions

We train our end-to-end system to estimate WER per sentence as regression problem. The hyper-parameters for the system were tuned using two evaluation metrices: Pearson correlation and root mean square error (RMSE) for the development set and results are reported for the test set. In our feature ablation study, we evaluated the six following systems:

- $\mathcal{A}$: Decoder features + MFCC + lexical features
- $\mathcal{B}$: $\mathcal{A}$ + phonotactic features
- $\mathcal{C}$: MFCC + lexical features
- $\mathcal{D}$: $\mathcal{C}$ + phonotactic features
- $\mathcal{E}$: MFCC
- $\mathcal{F}$: $\mathcal{E}$ + phonotactic features

The first two rows in table 5 show the glass-box and black-box results from our baseline system; the e-WER where we combined word-based and grapheme-based ASR results for the same sentence. system $\mathcal{A}$ shows our first multistream architecture which combines acoustics, lexical and decoder features. System ($\mathcal{B}$ **e-WER2 glass-box**) achieves Pearson correlation of 0.81, which outperforms the glass-box in e-WER with no need to run grapheme based speech recognition for the same language. System $\mathcal{C}$ is trained using the lexical and acoustics features only. System ($\mathcal{D}$ **e-WER2 black-box**) achieves Pearson correlation of 0.68, which outperforms the e-WER black-box reference systems. At this stage, we are confident that our multistream system is capable of learning a joint representation for acoustics and linguistic (textual and phonotactic) features to estimate the WER.

Our experiments show that the proposed multistream architecture can estimate the WER efficiently without requiring graphemic recognition. Fan et al [26] estimated WER without

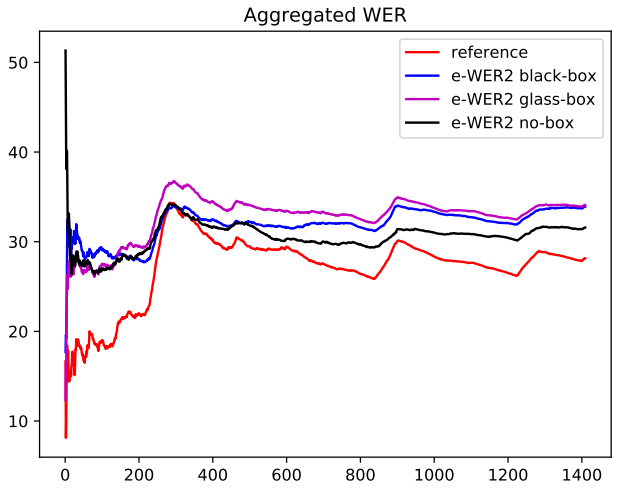

Figure 2: *Test set cumulative WER over all sentences X-axis is duration in hours and Y-axis is WER in %.*

the requiring explicit transcriptions, but did require access to the acoustic models. Here, we ask; Can we estimate the WER without having access to neither the transcript nor the speech recognition system? In our attempt to answer this, we build system $\mathcal{E}$ in table 5 which uses only acoustic features. Clearly, the system is not capable of learning any pattern given that the MFCC features. When combined with phoneme recognition output ($\mathcal{F}$) we see a large improvement in Pearson correlation by combining acoustic and phonotactic features, still without access to the ASR system.

To further visualise these results, figure 2 plots the cumulative WER, glass-box, black-box and no-box in the e-WER2 framework, across the three hours test set. The large difference during the first 100 utterances arises owing to the glass-box and black-box systems in the e-WER2 framework are capable of better estimation with fewer data points. It is worth to mention that when we swap train and dev, the results are similar.

7. Conclusions

This paper continues our effort in predicting speech recognition WER without requiring a gold-standard reference transcription. We presented an end-to-end multistream based regression model to predict the WER per sentence. Our approach benefits from combining word-based and phoneme-based ASR results, in addition to the MFCCs for the same sentence. Our experiments indicate that this approach can effectively estimate WER per sentence and we have aggregated the estimated results to predict WER for complete test sets without the need for a reference transcription. We also introduced a "no-box" WER estimation approach (e-WER2) which does not need to have access to the ASR system. A potential limitation of this work is the restriction to only one language, so for future work, we shall continue our investigation to estimate WER across different languages and multilingual ASR systems. We also plan to use e-WER for lattice *n*-best ranking for second pass rescoring.

Acknowledgements: This work was partially supported by EU H2020 project "European Language Grid" (grant agreement ID: 825627).

8. References

[1] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Achieving human parity in conversational speech recognition," *arXiv preprint arXiv:1610.05256*, 2016.

[2] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L.-L. Lim, B. Roomi, and P. Hall, "English conversational telephone speech recognition by humans and machines," *arXiv preprint arXiv:1703.02136*, 2017.

[3] S. Watanabe, M. Mandel, J. Barker, and E. Vincent, "Chime-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," *arXiv preprint arXiv:2004.09249*, 2020.

[4] A. Ali, S. Vogel, and S. Renals, "Speech Recognition Challenge in the Wild: Arabic MGB-3," in *ASRU*, 2017.

[5] K. Fan, J. Wang, B. Li, F. Zhou, B. Chen, and L. Si, "Bilingual expert can find translation errors," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 6367–6374.

[6] M. Yang, X. Hu, H. Xiong, J. Wang, Y. Jiaermuhamaiti, Z. He, W. Luo, and S. Huang, "Ccmt 2019 machine translation evaluation report," in *China Conference on Machine Translation*. Springer, 2019, pp. 105–128.

[7] E. Fonseca, L. Yankovskaya, A. F. Martins, M. Fishel, and C. Federmann, "Findings of the WMT 2019 shared tasks on quality estimation," in *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, 2019, pp. 1–10.

[8] R. W. Ng, K. Shah, L. Specia, and T. Hain, "A study on the stability and effectiveness of features in quality estimation for spoken language translation," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[9] ——, "Groupwise learning for ASR k-best list reranking in spoken language translation," in *ICASSP*, 2016.

[10] G. Evermann and P. Woodland, "Posterior probability decoding, confidence estimation and system combination," in *Proc. Speech Transcription Workshop*, 2000.

[11] S. Cox and S. Dasmahapatra, "High-level approaches to confidence estimation in speech recognition," *IEEE Transactions on Speech and Audio processing*, 2002.

[12] H. Jiang, "Confidence measures for speech recognition: A survey," *Speech Communication*, 2005.

[13] M. S. Seigel, P. C. Woodland *et al.*, "Combining information sources for confidence estimation with CRF models." in *Interspeech*, 2011.

[14] P.-S. Huang, K. Kumar, C. Liu, Y. Gong, and L. Deng, "Predicting speech recognition confidence using deep learning with word identity and score features," in *ICASSP*, 2013.

[15] K. Kalgaonkar, C. Liu, Y. Gong, and K. Yao, "Estimating confidence scores on ASR results using recurrent neural networks," in *ICASSP*, 2015.

[16] M. A. Del-Agua, A. Gimenez, A. Sanchis, J. Civera, and A. Juan, "Speaker-adapted confidence measures for ASR using deep bidirectional recurrent neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 7, pp. 1198–1206, 2018.

[17] P. Swarup, R. Maas, S. Garimella, S. H. Mallidi, and B. Hoffmeister, "Improving ASR confidence scores for Alexa using acoustic and hypothesis embeddings," in *INTERSPEECH*, 2019.

[18] M. S. Seigel and P. C. Woodland, "Detecting deletions in ASR output," in *ICASSP*. IEEE, 2014.

[19] E. Simonnet, S. Ghannay, N. Camelin, Y. Estève, and R. De Mori, "ASR error management for improving spoken language understanding," *arXiv preprint arXiv:1705.09515*, 2017.

[20] A. Ali and S. Renals, "Word error rate estimation for speech recognition: e-WER," in *ACL*, 2018.

[21] M. Negri, M. Turchi, J. G. de Souza, and D. Falavigna, "Quality estimation for automatic speech recognition." in *COLING*, 2014.

[22] J. G. de Souza, H. Zamani, M. Negri, M. Turchi, and D. Falavigna, "Multitask learning for adaptive quality estimation of automatically transcribed utterances." in *NAACL*, 2015.

[23] S. Jalalvand and D. Falavigna, "Stacked auto-encoder for ASR error detection and word error rate prediction," in *INTERSPEECH*, 2015.

[24] S. Jalalvand, D. Falavigna, M. Matassoni, P. Svaizer, and M. Omologo, "Boosted acoustic model learning and hypotheses rescoring on the Chime-3 task," in *ASRU*, 2015.

[25] S. Jalalvand, M. Negri, F. Daniele, and M. Turchi, "Driving rover with segment-based ASR quality estimation," in *ACL*, 2015.

[26] K. Fan, J. Wang, B. Li, B. Chen, and N. Ge, "Neural zero-inflated quality estimation model for automatic speech recognition system," *arXiv preprint arXiv:1910.01289*, 2019.

[27] A. Vyas, P. Dighe, S. Tong, and H. Bourlard, "Analyzing uncertainties in speech recognition using dropout," in *ICASSP*, 2019.

[28] Y.-C. Tam, Y. Lei, J. Zheng, and W. Wang, "ASR error detection using recurrent neural network language model and complementary ASR," in *ICASSP*, 2014.

[29] M. Najafian, S. Khurana, S. Shan, A. Ali, and J. Glass, "Exploiting convolutional neural networks for phonotactic based dialect identification," in *ICASSP*. IEEE, 2018.

[30] P. Matejka, P. Schwarz, J. Cernockỳ, and P. Chytil, "Phonotactic language identification using high quality phoneme recognition." in *INTERSPEECH*, 2005.

[31] Y. Hifny and A. Ali, "Efficient Arabic emotion recognition using deep neural networks," in *ICASSP*, 2019.

[32] A. Ali, "Multi-dialect Arabic broadcast speech recognition," Ph.D. dissertation, The University of Edinburgh, 2018.

[33] A. Ali, P. Bell, J. Glass, Y. Messaoui, H. Mubarak, S. Renals, and Y. Zhang, "The MGB-2 challenge: Arabic multi-dialect broadcast media recognition," in *SLT*, 2016.

[34] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *INTERSPEECH*, 2015.

[35] D. Povey, V. Peddinti, D. Galvez, P. Ghahrmani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free mmi," in *Interspeech*, 2016.

[36] 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 *ASRU*, 2011.