Prosodically-enhanced Recurrent Neural Network Language Models

Siva Reddy Gangireddy[1], *Steve Renals*[1], *Yoshihiko Nankaku*[2] *and Akinobu Lee*[2]

[1]Centre for Speech Technology Research, University of Edinburgh, UK
[2]Department of Computer Science, Nagoya Institute of Technology, Nagoya, Japan
s.gangireddy@sms.ed.ac.uk, s.renals@ed.ac.uk, {nankaku, ri}@nitech.ac.jp

Abstract

Recurrent neural network language models have been shown to consistently reduce the word error rates (WERs) of large vocabulary speech recognition tasks. In this work we propose to enhance the RNNLMs with prosodic features computed using the context of the current word. Since it is plausible to compute the prosody features at the word and syllable level we have trained the models on prosody features computed at both these levels. To investigate the effectiveness of proposed models we report perplexity and WER for two speech recognition tasks, Switchboard and TED. We observed substantial improvements in perplexity and small improvements in WER.

Index Terms: RNNLMs, 3-gram, prosody features, pause duration, duration of the word, syllable duration, syllable F0, GMM-HMM, DNN-HMM, Switchboard conversations and TED lectures

1. Introduction

Current large vocabulary speech recognition systems typically comprise an acoustic model which relates acoustic features to sub-word units, a lexical model of some kind (typically a dictionary), and a language model which provides probability estimates for word sequences. Suprasegmental prosodic features, such as intonation and timing information, fit uneasily into such a framework. However, prosodic features are of potential interest in speech recognition: they are relatively robust to noise and provide rich additional information. Prosodic information is available at various levels in a speech signal: within words (for instance, word and phone duration), between the words (for instance, pause duration), and across multiple words (for instance, F0 contour).

Prosodic information has been successfully used for topic segmentation [1], disfluency detection [2], processing of multi party meetings [2], dialogue act classification [3], sentiment classification [4] and emotion recognition [5]. There have also been a number of attempts to include prosodic information in language modelling.

Vergyri et al [6] and Gadde [7] investigated the use of modelling word durations, using an explicit Gaussian mixture model, trained on feature vectors constructed by considering the duration of the phones in that word; Vergyri et al also used n-grams to model the pause duration between the words. Prosodic structure may be interpreted as correlating with non-word phenomena such as sentence boundaries and speech disfluencies including repetitions, deletions and filled pauses; Stolcke et.al [8] attempted to incorporate this structure using a hidden event n-gram language model, with prosody of hidden events modelled using a decision tree classifier trained using speech data annotated for sentence boundaries and disfluencies.

Huang and Renals [9] argued that prosodic information could be more naturally captured at a syllabic level, and used an acoustically-based system for automatic syllable identification. Four-dimensional prosodic feature vectors were extracted, containing F0, energy, slope of F0 and durational information for the syllable, which were vector quantised using a 16-word codebook. Thus a syllable was represented by a single codeword, and a word by a sequence of syllable codewords. This representation is amenable to n-gram modelling, using factorisation and a hierarchical prior in this case. Maximum entropy language models have also been used to capture prosodic information, such as learning the dependencies between the syntactic features, such as POS tags, and prosodic features, such as accent and duration [10].

An n-gram model defines a distribution over discrete symbol sequences which is not the most natural representation for continuously valued prosodic features. Neural network language models [11, 12, 13, 14] transform symbolic word representations to a continuous space, and a number of recent papers have augmented the word feature input to a neural network language model to incorporate additional context [15] or to learn correlations between words and richer annotations such as part-of-speech tags [16]. We have built on these approaches by developing a neural network language model with an extra feature layer to jointly model words and the related prosodic features computed from the context of the current word. We also model prosody at the syllabic level using an automatic syllable detection algorithm discussed in Section 2 and vary the amount of syllabic context (from 1–10 syllables).

We have performed language modelling experiments on the Switchboard and TED corpora reporting results in terms of both perplexity and word error rate (WER) on standard test sets.

2. Recurrent Neural Network Language Model

Neural network language models have been proposed to address some of the drawbacks of n-grams [11, 12]. Recurrent neural network language models (RNNLMs) estimate the probability of a word given its (potentially infinite) context in a low dimensional continuous space. The recurrent hidden connections in the RNNLM are responsible for learning the temporal dependencies. RNNLMs have been shown to consistently improve the perplexity and WER of standard speech recognition tasks, compared to n-grams and feed-forward neural network language models (NNLMs) [12, 13, 14, 17, 18].

This research was supported by the Core Research for Evolutional Science and Technology (CREST) from the Japan Science and Technology Agency (JST) (*uDialogue* project). The authors would like to thank Liang Lu for providing the Switchboard DNN acoustic models, Pawel Swietojanski for providing the TED DNN acoustic models and Matthew Aylett for discussions on syllable detection algorithm.

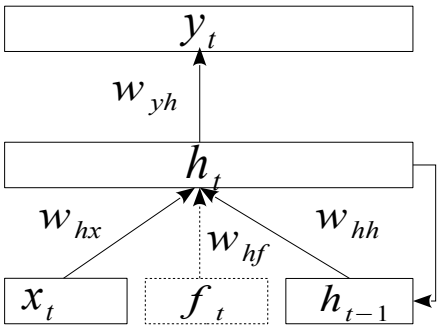

Figure 1: Recurrent neural network language model with a feature layer [13, 15]

The architecture of an RNNLM with an extra feature layer is shown in Figure 1. The input to the network at time t comprises the index of the previous word, the feature vector (f_t) at time t and the state of the hidden layer at time t–1. The index of the previous word is encoded using 1 of N coding. The hidden layer at time t and the output probability distribution are computed as follows:

$$h_t = f(W_{hx}x_t + W_{hh}h_{t-1} + W_{hf}f_t) \quad (1)$$

$$y_t = g(W_{yh}h_t), \quad (2)$$

where x_t is the input vector, h_{t-1} is the state of the hidden layer at time $t-1$, h_t is the state of hidden layer at time t, f_t is the prosodic feature vector at time t, and y_t is the output probability distribution. f, g are sigmoid activation and softmax functions, respectively. RNNLMs are trained using the back propagation through time (BPTT) algorithm [19]. In BPTT, training is accomplished by unfolding the recurrent network through time, and training as a deep network with hidden layer weights for each time step constrained to be equal. In the experiments reported in this work, the error is propagated three steps back in time. The objective of training is to maximize the likelihood of training data by minimizing the cross entropy difference between the target and output probability distributions.

We propose an RNNLM which learns a hidden recurrent state that combines lexical and prosodic information, enabling the RNNLM to learn dependences between word context and prosodic context. We explored a number of of prosodic features in this work:

Word duration (*RNNLM-worddur*): Duration of the previous word, obtained by forced alignment of the training data.

Pause duration (*RNNLM-pause*): Duration of the pause preceding the current word (zero if no pause).

Final phone duration (*RNNLM-fphonedur*): The pause duration between the words effects the duration of the previous word. This effect is known as pre-pausal lengthening [7]. To model this effect the RNNLMs are given with duration of the final phone in preceding word.

Syllable duration (*RNNLM-syldur*): Duration of the syllables extracted from the context of the current word, encoded as the index of the previous word and durations of the syllables in the context. The syllables are extracted from the acoustics using an automatic syllable extraction algorithm [20]. A simple alignment procedure was used to obtain the syllabic context, in which a fixed number of syllable durations preceding the current word are given as an input to the RNNLM, irrespective of sentence boundaries. To investigate the effect of context length, RNNLMs were trained on the durations of the three, five and ten preceding syllables.

Syllable F0 (*RNNLM-F0*): Similar to the syllable duration experiments, RNNLMs are given with word feature and F0 features computed from the syllables in the context of the current word. Four different features are computed at each syllable: mean, maximum, minimum and range of F0. Before computing the features the F0 values of each syllable are normalised using z-score normalisation. To investigate the effect of length of the context the RNNLMs were trained on F0 features computed from three, five and ten preceding syllables.

All duration features were unnormalised for speaker, channel, or recording session.

3. Text Experiments

Our first experiments used the Switchboard training transcripts, a total of 3.4M tokens, of which the first 130K tokens were used as validation data, to tune the parameters. We report perplexity (PPL) results using the validation data and the Switchboard evaluation set *eval2000* (LDC2002S09). The *eval2000* data contains 20 Switchboard and 20 CallHome English (CHE) conversations. In this work we report PPLs and WERs on Switchboard conversations only. Hereafter, the Switchboard part of *eval2000* is referred as *eval2000-swbd*.

We estimated a back-off trigram LM by interpolating the LMs trained on 3.2M tokens of Switchboard training transcripts and 11M tokens of Fisher English part1 transcripts (LDC2004T19). The LMs are trained using Kneser-Ney smoothing and the interpolation coefficients are optimized for better PPL on validation data. This pruned trigram LM is used in first pass decoding to generate the lattices.

The RNNLMs were trained only on the 3.2M tokens of Switchboard training transcripts. Further details about the acoustic models used to align the data are given in Section 4.1. A vocabulary of 30,000 words was used. The RNNLM used 300 hidden units in the recurrent hidden layer. A factored output layer with 100 classes was used to reduce the computational complexity [14].

Perplexity results for the Switchboard validation and evaluation sets are shown in Table 1. These results indicate that the RNNLM improves the perplexity over the 3-gram LM by about 5% relative. The results using the pause duration features result in an improvement of about 13% relative over the baseline RNNLM. The word duration features also reduce the perplexity over the baseline RNNLM, but by a much smaller amount, and the final phone duration features result in an 8–10% relative reduction over the RNNLM baseline. The syllable duration and F0 features result in a reduction of perplexity of over 15% and 13% relative respectively. Similar improvements can be observed after linear interpolation with the 3-gram baseline. The interpolation coefficient was 0.5 in all cases.

4. ASR Experiments

Following the experiments to evaluate language model perplexity, we carried out a set of speech recognition experiments on two tasks: the recognition of Switchboard telephone conversations, and the recognition of TED talks. The language models were evaluated in terms of WER and for each corpus we used standard evaluation protocols: NIST CTS evaluation for Switchboard and IWSLT for TED. We note that differ-

Model	Validation	+KN3	*eval2000-swbd*	+KN3
KN3	82.4	82.4	81.9	81.9
RNNLM	78.4	70.6	77.5	70.8
RNNLM-pause	68.6	64.4	66.5	63.9
RNNLM-worddur	70.8	65.8	76.7	68.2
RNNLM-fphonedur	70.2	65.3	70.7	66.1
RNNLM-syldur	63.7	61.4	65.0	62.0
RNNLM-sylF0	70.1	64.6	67.3	63.6

Table 1: Perplexities of 3-gram, RNNLM and prosody RNNLMs trained on word duration, pause duration, duration of final phone, syllable durations and syllable F0 features. Here the RNNLM-syldur and RNNLM-sylF0 models are trained on syllable context length of five. 3-gram LM is trained on combination of **Switchboard and Fisher** transcripts and the RNNLMs are trained only on part of Switchboard training transcripts (3.2M).

ent prosodic effects are observed, since Switchboard consists of conversational telephone speech, and TED consists of well-prepared public talks.

4.1. Switchboard

We report the WERs on the *eval2000-swbd* test set, which comprises a total of 20 conversations, containing 21,000 word tokens with an out of vocabulary (OOV) rate of 5% with respect to our vocabulary. Two speech recognition acoustic models were used to compute the prosody features: GMM-based and DNN-based.

The GMM-based acoustic models were trained on 300 hours of switchboard data (LDC97S62), with the Mississippi state transcripts (`http://www.isip.piconepress.com/`). The acoustic features comprised 7 frames (± 3) of 13-dimension MFCC features, with the dimension reduced to 40 using a linear discriminant analysis (LDA) transform, followed by a decorrelating semi-tied covariance (STC) transform. The features were adapted per speaker using feature space (constrained) maximum likelihood linear regression (fMLLR). The maximum likelihood system trained on LDA+STC+fMLLR features is then discriminatively trained using the boosted maximum mutual information (bMMI) criteria, with a b value of 0.1. All the baseline experimental results reported here are reproduced using Kaldi speech recognition recipe given in [21][1].

The DNN-based acoustic models were also trained on 300 hours of Switchboard telephone conversation data [22]. The acoustic features comprised 11 frames (± 5) of MFCC features, including delta and acceleration coefficients. The features were transformed using LDA and decorrelated using STC, as for the GMM-based system. However, they were not speaker adapted. The output layer consists of 8827 nodes, corresponding to the set of context-dependent HMM states, used in the GMM-based system. Six hidden layers of 2048 units with a sigmoid non-linearity were used.

The RNNLMs trained on word and prosody features were incorporated into the ASR process by rescoring the 100-best lists, generated from the lattices of the GMM-based system. The prosodic features were computed by aligning the 100-best lists with the acoustics using both GMM-based and DNN-based acoustic models. To compute the final score, the scores of RNNLM are interpolated with the scores of n-grams from the 100-best lists. The interpolation coefficients are optimized for better WERs. The WERs computed on Switchboard conversations of *eval2000* are given in Table 2.

[1]Due to changes in the Kaldi recipe the reported results here are different to those originally published in [21]

Model	GMM(%WER)	DNN(%WER)
3-gram	19.5	19.5
RNNLM	18.1	18.1
RNNLM-pause	18.0	17.9
RNNLM-worddur	17.6	17.9
RNNLM-fphonedur	17.8	18.0

Table 2: %WERs computed on 100-best lists of *eval2000* data set (Switchboard conversations only). **GMM** and **DNN**-based acoustic models are used to force align the transcripts

In second column of Table 2, we can observe that the RNNLM improves the baseline system by 1.4% absolute. In case of proposed prosody RNNLMs, we can observe 0.5% absolute (3% relative) reduction in WER with RNNLM-worddur and 0.3% absolute with RNNLM-fphonedur models. The improvement with the RNNLM-pause model is much smaller compared to the other models.

In third column of Table 2 we report the WERs on same *eval2000* data set but aligned using the DNN-based acoustic models. After aligning the 100-best lists with the DNN-based acoustic models the RNNLM-pause model improves the baseline RNNLM by 0.2% absolute. The accuracy of RNNLM-worddur and RNNLM-fphonedur models were reduced compared to the WERs reported in second column of Table 2.

4.2. TED talks

DNN-based acoustic models were used for training the acoustic models for TED [23, 24]. The DNNs were trained using 143 hours of TED lectures, recorded before 2010 and 78 hours of AMI meeting data (`http://corpus.amiproject.org`). The DNN acoustic features comprised 11 frames (± 5) of MFCC features, including delta and acceleration features. The features were transformed using LDA and adapted per speaker using fMLLR. The DNNs were trained by optimising the cross entropy objective function. There were six 2048-unit hidden layers with sigmoid non-linearity. A pruned 3-gram language model trained on 351M word tokens was used in the first pass decoding to generate the lattices [25].

We report PPLs and WERs on *tst2011*, an evaluation set for IWSLT[2]. The RNNLM and prosodically-enhanced RNNLMs were trained using 2.8M tokens, a combination of TED lectures and AMI data. The development data for tuning the parameters and for early stopping was *tst2012*, also an evaluation set for IWSLT, consisting of 20,000 tokens. 400 hidden units were again used with a sigmoid non-linearity and a factored output layer with 200 classes was used to reduce the computational requirements.

The WERs of *tst2011* dataset are given in Table 3. The RNNLM improves the baseline system by 0.7% absolute (5% relative). As expected, the proposed RNNLM-pause and RNNLM-worddur models are not effective enough to improve the WERs. However, the RNNLM-fphonefur model improves the WERs by 0.2% absolute (2% relative).

[2]International Workshop on Spoken Language Translation

Model	PPL	%WER
3-gram	120.2	12.6
RNNLM	198.0	11.9
RNNLM-pause	184.1	12.0
RNNLM-worddur	194.1	11.8
RNNLM-fphonedur	184.2	11.7

Table 3: PPLs and %WERs computed on *tst2011* and 100-best lists of *tst2011*, respectively. DNN-HMM hybrid acoustic models trained on TED and AMI data are used to force align the transcripts and compute the prosody features

5. Syllable duration and F0

Finally we explored the use of syllable duration and F0 features for the Switchboard task. To investigate the effect of length of syllable context the RNNLM-syldur models were trained on context lengths of 3, 5 and 10. 300 recurrent hidden units were used, and a factored output layer with 100 classes was used to reduce the time complexity. From Table 4, we can observe 0.3% absolute (2% relative) reduction in WER using the RNNLM trained on durations of 5 syllables (RNNLM-syldur5) from the context of the current word. Similarly we can observe 0.2% absolute improvements with a context length of 3 and 0.1% absolute improvement with a context length of 10.

Model	PPL	%WER
3-gram	81.9	19.5
RNNLM	77.5	18.1
RNNLM-syldur3	63.5	17.9
RNNLM-syldur5	65.0	17.8
RNNLM-syldur10	63.2	18.0

Table 4: %WERs are computed on 100-best lists of *eval2000* data set (Switchboard conversations only). Automatic syllable detection algorithm is used to get the boundaries of syllables. GMM-based acoustic models are used to get the word boundary information

Similar to the RNNLM-syldur models the effect of syllable context is investigated by training the RNNLM-sylF0 models on context lengths of 3, 5 and 10. The feature vector is obtained by concatenating the F0 features computed at each syllable in the context[3]. The features computed are mean, maximum, minimum and range of F0. Hidden layer has 300 hidden neurons and a factored layer with 100 classes was used to reduce the computational complexity. From the Table 5, we can observe improvements in PPLs but not in WERs.

Model	PPL	%WER
3-gram	81.9	19.5
RNNLM	77.5	18.1
RNNLM-sylF0_3	66.1	18.4
RNNLM-sylF0_5	67.3	18.3
RNNLM-sylF0_10	73.3	18.7

Table 5: %WERs are computed on 100-best lists of *eval2000* data set (Switchboard conversations only). Before computing the F0 features the F0 sequence of each syllable is normalised using z-score normalisation. GMM-based acoustic models are used to get the word boundary information

[3]Kaldi pitch tool is used to compute the F0 features [26]

6. Discussion and Summary

From the experimental results given in Table 2 we can observe moderate improvements with the RNNLMs trained on prosodic features over the baseline RNNLM. We can observe significant improvements with the RNNLM-worddur models than the other models. The RNNLM-worddur model improves the baseline by reducing the number of deletions or in other terms during re-ranking the model selects the longer sentences. In case of RNNLM-pause model, surprisingly there is no correlation between the PPL and WER improvements. A possible reason for this behaviour is that the n-best lists can be noisy and it is sometimes difficult to get precise alignments to the pause regions. We can observe another 0.1% absolute with the pause features computed by DNN-based hybrid acoustic models. From the Table 4, we can observe that the length of the context has an effect on percentage of errors. As the length of the context increases the accuracy of the models got reduced (RNNLM-syldur10). Given RNNLM-sylF0 models (in Table 5) are doing worse than baseline models further experiments can be done by training the models on combination of syllable duration and F0 features. Here the features are normalised by z-score normalisation, more experiments can be done by training the models on unnormalised features or normalised by other techniques.

Given the improvements with the prosody RNNLMs, the correlations between the words and prosody features can alternatively be modelled by using multitask learning [27, 28, 29]. In multitask learning an extra output layer can be added to predict the prosodic features. During testing the network only predicts the probability distribution over the words. In the experiments reported here the durations are not normalised for speaker variability. Further experiments can be performed by normalising the durations for speaker and channel variations. In this work, the effect of pause duration on preceding word is modelled by training the RNNLMs on duration of the final phone in the preceding word. This effect can be further investigated by training the RNNLMs on duration of the final vowel or syllable in preceding word. Given the prosody RNNLMs trained only on 3.2M tokens of acoustic data, more experiments can be done to investigate how well the prosody RNNLMs trained on much more data are complimentary to the baseline models, also trained on more data.

In this work we have trained the RNNLMs on duration of the previous word, the pause duration between the words, the duration of the final phone in the preceding word, the duration of the syllables and syllable F0 features. The proposed models improve perplexity before and after interpolation with the 3-gram baseline. We report WERs for the Switchboard and TED tasks, observing small reductions in WER, compared with an RNN baseline.

7. References

[1] G. Tür, A. Stolcke, D. Hakkani-Tür, and E. Shriberg, "Integrating prosodic and lexical cues for automatic topic segmentation," *Comput. Linguist.*, vol. 27, no. 1, pp. 31–57, Mar. 2001.

[2] E. Shriberg, A. Stolcke, and D. Baron, "Can prosody aid the automatic processing of multi-party meetings? evidence from predicting punctuation, disfluencies, and overlapping speech," Sept. 2001.

[3] E. Shriberg, R. Bates, A. Stolcke, P. Taylor, D. Jurafsky, K. Ries, N. Coccaro, R. Martin, M. Meteer, and C. Van Ess-Dykema, "Can prosody aid the automatic classification of dialog acts in conversational speech?" June 2000.

[4] F. Mairesse, J. Polifroni, and G. D. Fabbrizio, "Can prosody in-

form sentiment analysis? experiments on short spoken reviews," in *Proc. of ICASSP*, 2012, pp. 5093–5096.

[5] I. Luengo, E. Navas, I. Hernáez, and J. Sánchez, "Automatic emotion recognition using prosodic parameters," in *Proc. of INTERSPEECH*, 2005, pp. 493–496.

[6] D. Vergyri, A. Stolcke, V. Gadde, L. Ferrer, and E. Shriberg, "Prosodic knowledge sources for automatic speech recognition," in *Proceedings of ICASSP*, vol. 1, April 2003, pp. 208–211.

[7] V. R. R. Gadde, "Modeling word durations," in *Proc. Interspeech*, 2000, pp. 601–604.

[8] A. Stolcke, E. Shriberg, D. Hakkani-tür, and G. Tür, "Modeling the prosody of hidden events for improved word recognition," in *Proc. EUROSPEECH*, 1999, pp. 307–310.

[9] S. Huang and S. Renals, "Modeling prosodic features in language models for meetings," in *Machine Learning for Multimodal Interaction IV*, ser. Lecture Notes in Computer Science, 2007, vol. 4892, pp. 191–202.

[10] O. Chan and R. Toggery, "Prosodic features for a maximum entropy language model," in *Proc. of Interspeech*, 2006.

[11] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of Machine Learning Research*, pp. 1137–1155, 2003.

[12] H. Schwenk, "Continuous space language models," *Computer Speech & Language*, pp. 492–518, 2007.

[13] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. Interspeech*, 2010, pp. 1045–1048.

[14] T. Mikolov, S. Kombrink, L. Burget, J. Cernocky, and S. Khudanpur, "Extensions of recurrent neural network language model," in *Proc. of ICASSP*, May 2011, pp. 5528–5531.

[15] T. Mikolov and G. Zweig, "Context dependent recurrent neural network language model," in *Spoken Language Technology Workshop (SLT)*, 2012, pp. 234–239.

[16] Y. Wu, X. Lu, H. Yamamoto, S. Matsuda, C. Hori, and H. Kashioka, "Factored language model based on recurrent neural network," in *Proceedings of COLING*, 2012, pp. 2835–2850.

[17] J. Park, X. Liu, M. J. F. Gales, and P. C. Woodland, "Improved neural network based language modelling and adaptation," in *Proc. Interspeech*, 2010, pp. 1041–1044.

[18] S. Gangireddy, F. McInnes, and S. Renals, "Feed forward pretraining for recurrent neural network language models," in *Proc. Interspeech*, September 2014, pp. 2620–2624.

[19] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[20] M. Aylett, "Detecting high level dialog structure without lexical information," in *Proc. of ICASSP*, vol. 1, May 2006, pp. I–I.

[21] K. Vesely, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Proc. of Interspeech*, Lyon, France, August 2013.

[22] L. Lu and S. Renals, "Feature-space speaker adaptation for probabilistic linear discriminant analysis acoustic models," in *Proc. of Interspeech*, 2015.

[23] P. Swietojanski, A. Ghoshal, and S. Renals, "Hybrid acoustic models for distant and multichannel large vocabulary speech recognition," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, Dec 2013, pp. 285–290.

[24] P. Bell, P. Swietojanski, J. Driesen, M. Sinclair, F. McInnes, and S. Renals, "The UEDIN english asr systems for the IWSLT 2014 evaluation," in *Proc. International Workshop on Spoken Language Translation*, 2014.

[25] P. Bell, F. McInnes, S. Gangireddy, M. Sinclair, A. Birch, and S. Renals, "The UEDIN english asr system for the IWSLT 2013 evaluation," in *Proc. International Workshop on Spoken Language Translation*, 2013.

[26] P. Ghahremani, B. BabaAli, D. Povey, K. Riedhammer, J. Trmal, and S. Khudanpur, "A pitch extraction algorithm tuned for automatic speech recognition," in *Proc. of ICASSP*, May 2014, pp. 2494–2498.

[27] R. Caruana, "Multitask learning," *Machine Learning*, vol. 28, no. 1, pp. 41–75, 1997.

[28] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 160–167.

[29] M. Seltzer and J. Droppo, "Multi-task learning in deep neural networks for improved phoneme recognition," in *Proc. of ICASSP*, May 2013, pp. 6965–6969.