STIMULATED TRAINING FOR AUTOMATIC SPEECH RECOGNITION AND KEYWORD SEARCH IN LIMITED RESOURCE CONDITIONS

A. Ragni, C. Wu, M. J. F. Gales, J. Vasilakes, K. M. Knill

Department of Engineering, University of Cambridge
Trumpington Street, Cambridge CB2 1PZ, UK
{ar527,cw564,mjfg,jav39,kate.knill}@eng.cam.ac.uk

ABSTRACT

Training neural network acoustic models on limited quantities of data is a challenging task. A number of techniques have been proposed to improve generalisation. This paper investigates one such technique called stimulated training. It enables standard criteria such as cross-entropy to enforce spatial constraints on activations originating from different units. Having different regions being active depending on the input unit may help network to discriminate better and as a consequence yield lower error rates. This paper investigates stimulated training for automatic speech recognition of a number of languages representing different families, alphabets, phone sets and vocabulary sizes. In particular, it looks at ensembles of stimulated networks to ensure that improved generalisation will withstand system combination effects. In order to assess stimulated training beyond 1-best transcription accuracy, this paper looks at keyword search as a proxy for assessing quality of lattices. Experiments are conducted on IARPA Babel program languages including the surprise language of OpenKWS 2016 competition.

Index Terms— limited resources, stimulated training, joint decoding, keyword search

1. INTRODUCTION

There are several important issues one needs to address when training neural network acoustic models. For small sample problems that arise in limited resource conditions generalisation may be one of the most important issues. As the amount of data gradually decreases, standard procedures for building automatic speech recognition (ASR) systems yield less and less accurate transcriptions [1]. Another related issue is that of a model complexity control [2] that becomes particularly acute with these forms of models. Finally, non-convex optimisation makes parameter initialisation important. A lot of work has been done to address these interconnecting issues. For instance, approaches examined for network initialisation range from using generative model pre-training [3], monophone initialisation [4] to the use of multi-language data [5]. Rather than using monophone networks for initialisation only, it is also possible to train a network with both monophone and context-dependent output layers [6]. A similar approach is often used to train multi-language networks [7]. These multi-task networks are expected to yield representations that generalise better due to the need to solve multiple tasks simultaneously [8]. Another group of approaches attempts to increase the amount of training data. The extra data may come from various sources such as other languages [5], untranscribed data [9], waveform [10] or parameter sequence [11, 12] perturbation. Finally, the most related to this work is a group that enhances generalisation through a modification of the standard training process. Examples include dropout [13] and stimulated training [14, 15]. Procedurally, the dropout consists of randomly eliminating activation function values during training. This is supposed to improve generalisation since it encourages a network to learn robust representations. Stimulated training [14, 15], in addition to robustness, addresses another issue that all neural networks have in speech processing. This is a poor interpretability of quantities such as network weights and activations. By organising activations into a grid with superimposed phone targets, the stimulated training enables representations to be learnt that yield high activations for any given phone only in the vicinity of that phone superimposed on the grid.

The previous work with stimulated training has looked at both interpretability [14] as well as generalisation for ASR of English and Javanese [15]. There are a number of important questions that remain to be answered. The nature of the phones superimposed on the grid is fundamental to stimulated training. Are language independent attributes, such as position in a word or syllable, as well as language dependent attributes, such as diacritics in languages like Pashto, of any use in producing representations that generalises well? It would be interesting to see how well stimulated training can handle model complexity issues such as network size. Another question is whether gains seen from stimulated training of systems would translate over to ensembles. Finally, in applications beyond 1-best transcription, the quality of generated lattices is of a more paramount importance. This paper looks at keyword search as a proxy. Experiments are conducted on 8 option period 3 languages of the IARPA Babel program including the surprise language of the OpenKWS 2016 competition.

This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense U. S. Army Research Laboratory (DoD/ARL) contract number W911NF-12-C-0012. The U. S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U. S. Government.

The rest of the paper is organised as follows. Section 2 describes stimulated training. Section 3 discusses the choice of units for embedding into the grid. Experimental results are presented in Section 4. Conclusions drawn from this work are given in Section 5.

2. STIMULATED TRAINING

A number of different neural networks have been examined for acoustic modelling in speech recognition [16, 17]. Among them, a feed-forward form is one of the simplest. This network applies layers of non-linear transformations to the input observation $\mathbf{o}$ to yield a distribution over targets at the output

$$\boldsymbol{\theta} = \boldsymbol{\sigma}^{(L)}(\mathbf{W}^{(L)}\boldsymbol{\sigma}^{(L-1)}(\ldots\boldsymbol{\sigma}^{(1)}(\mathbf{W}^{(1)}\mathbf{o}+\mathbf{b}^{(1)})+\mathbf{b}^{(L-1)})+\mathbf{b}^{(L)}) \quad (1)$$

where $\mathbf{b}^{(l)}$, $\mathbf{W}^{(l)}$ and $\boldsymbol{\sigma}^{(l)}$ are bias, weight matrix and non-linear transformation associated with the l-th layer, $\theta_i = P(S=i|\mathbf{o})$ is the posterior probability of the i-th target given observation $\mathbf{o}$. Targets typically correspond to hidden Markov model (HMM) states with probability density functions given by

$$p(\mathbf{o}|s) = \frac{1}{P(s)}P(s|\mathbf{o})p(\mathbf{o}) \quad (2)$$

where state s prior $P(s)$ is usually estimated from training counts and the distribution of observations $p(\mathbf{o})$ is usually set to a constant. The feed-forward networks are usually trained in stages. The first stage optimises a frame-level objective function such as cross-entropy

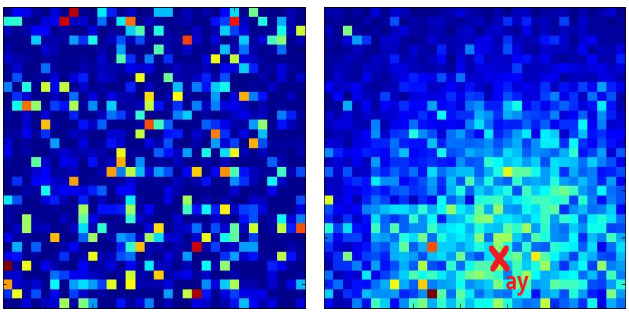
$$\mathcal{L}(\boldsymbol{\lambda}) = -\frac{1}{T}\sum_{t=1}^{T}\log(P(s_t|\mathbf{o}_t)) \quad (3)$$

where $\boldsymbol{\lambda}$ are network parameters. The second stage optimises a sequence-level objective function such as minimum Bayes risk

$$\mathcal{L}(\boldsymbol{\lambda}) = \frac{1}{R}\sum_{r=1}^{R}\sum_{\mathbf{w}}P(\mathbf{w}|\mathbf{O}^{(r)})\ell(\mathbf{w},\mathbf{w}_{\mathbf{ref}}^{(r)}) \quad (4)$$

where R is the number of sequences, $\mathbf{w}_{\mathbf{ref}}$ and $\mathbf{O}$ are reference word and observation sequences, ℓ is a loss function that may be defined at various levels such as state and phone [18].

One standard issue with these forms of models is a poor interpretability. Consider, for example, an 1024-dimensional output from one of the non-linearities arranged in a two-dimensional 32×32 grid in Figure 1 (a). There, bright regions,



(a) Unstimulated Activations (b) Stimulated Activations

Fig. 1. A typical impact of stimulated training on activations

corresponding to high activations, are scattered all over the place as one would expect from a distributed representation. Unfortunately, this may cause issues for regularisation and speaker adaptation as it is hard to relate one weight to another [15]. Although various approaches have been proposed to visualise feature space transformations [19, 20], they rarely focus on how to modify network behaviour simply by altering the space. Stimulated training [14, 15], in contrast, attempts to encourage activations to group in an interpretable way. Consider superimposing a phone set on the grid, which is roughly divided in half with vowels clustered at the bottom and consonants at the top, as shown in Figure 2. If activations corresponding to the vowels could have been

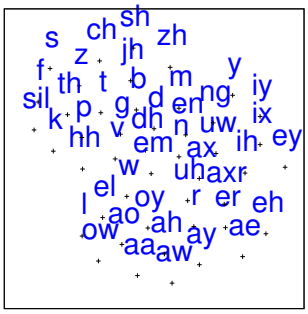


Fig. 2. Superimposed phone set

enhanced at the top and weakened at the bottom such an approach would have not only improved interpretability of the network but also encouraged better discrimination. This is exactly what stimulated training does but at even a finer phone level. Figure 1 (b) shows activation pattern corresponding to observation of phone /ay/. As can be seen activations are the highest in the vicinity of the phone.

Stimulated training can be implemented as a simple modification to the standard training procedure. Consider augmenting an objective function, such as equation (3) or (4), with a regularisation term

$$\mathcal{F}(\boldsymbol{\lambda}) = \mathcal{L}(\boldsymbol{\lambda}) + \alpha\mathcal{R}(\boldsymbol{\lambda}) \quad (5)$$

where $\mathcal{R}(\boldsymbol{\lambda})$ is the average per frame Kullback-Leibler divergence from normalised activation to a phone-specific prior given by

$$\mathcal{R}(\boldsymbol{\lambda}) = \frac{1}{T}\sum_{t=1}^{T}\sum_{l=1}^{L}\sum_{i=1}^{N^{(l)}} g(\mathbf{s}_i^{(l)},\mathbf{s}_{pt}^{(l)})\log\left(\frac{g(\mathbf{s}_i^{(l)},\mathbf{s}_{pt}^{(l)})}{\overline{\sigma}_{i,t}^{(l)}}\right) \quad (6)$$

The normalised activation is given by

$$\overline{\sigma}_{i,t}^{(l)} = \frac{\sigma_{i,t}^{(l)}\beta_i^{(l)}}{\sum_{j=1}^{N^{(l)}}\sigma_{j,t}^{(l)}\beta_j^{(l)}} \quad (7)$$

where $\beta_i^{(l)}$ reflects the importance of activation $\sigma_{i,t}^{(l)}$ at time t to the weights of the following layer

$$\beta_i^{(l)} = \sqrt{\sum_{j=1}^{N^{(l+1)}} W_{i,j}^{(l+1)^2}} \quad (8)$$

The phone-specific prior

$$g(\mathbf{s}_i^{(l)}, \mathbf{s}_{p_t}^{(l)}) = \frac{\exp\left(-\frac{1}{2\gamma^2}\|\mathbf{s}_i^{(l)} - \mathbf{s}_{p_t}^{(l)}\|_2^2\right)}{\sum_{j=1}^{N^{(l)}} \exp\left(-\frac{1}{2\gamma^2}\|\mathbf{s}_j^{(l)} - \mathbf{s}_{p_t}^{(l)}\|_2^2\right)} \quad (9)$$

is the normalised distance of the i-th activation to the target phone at time t with γ controlling smoothness. There are few options how activations can be arranged in a grid to map any individual activation $\sigma_i^{(l)}$ to its position on the two-dimensional grid $\mathbf{s}_i^{(l)}$. However, there is a great flexibility in arranging phones to map any individual phone p to its position $\mathbf{s}_p^{(l)}$ on the grid. One option is to used data-driven approaches such as t-SNE [21]. This consists of collecting phone-specific first and second order statistics in the observation space and then projecting it down to the two-dimensional grid space. Figure 2 shown earlier is an example of a typical projection.

3. UNIT SELECTION

The choice of a phone set is of fundamental importance as it defines the space where regularisation is performed. Standard phonetic lexica provide many interesting choices. Consider for a example an entry from a Cantonese dictionary

喫啡 g^II aM^MF;3 f^MI EM^FF;1

Here, each character represents a syllable and maps into two phonemes. There are two sorts of extra phone information: position and tone. The former offers information about position within a word and syllable after a caret symbol (^). Letters I, M, F are used to denote initial, middle and final position respectively. Hence, MI stands for the first phone of a syllable that is located in the middle of a word. The tones are specified using their numeric value after a semi-colon (;). Both position and tone may have a large impact on phonetic realisation with the latter being also linked with semantics.

For limited resource languages orthographic dictionaries are a popular alternative as they typically do not require expert phonetic knowledge to make [1]. These make use of written symbols, graphemes, to construct "pronunciation". Typically, rules are enforced to handle special cases such as signs and diacritics in languages like Kazakh and Pashto. Consider an example from Kazakh showing a phonetic and orthographic entry for English word seven

семь sAP^IIP e^MMP mAP^FFP
семь G41^IIP;D2 G10^MMP;D2 G30^MMP;D2D8

where P is a primary stress that illustrates another type of positional information. The place of phones in the orthographic entry are taken by graphemes G1, G2, etc. In contrast to Cantonese, Kazakh graphemes carry attributes such as script (D2 for Cyrillic) and sign. The soft sign ь despite being marked in the orthography is treated similar to a diacritic by altering the preceding grapheme with an attribute D8. Another example is from Pashto, which illustrates the use of attributes to communicate diacritics

اسحاقزي G1 G10 G24 G1 G21 G14 G6
اسحاقزى G1 G10 G24 G1 G21 G14 G6;DF
اسحاقزىٔ G1 G10 G24 G1 G21 G14 G6;DT

where DF stands for Farsi (letter) and DT stands for tail diacritic. Other attributes include non-full letters (hamza), diacritics (madda), nunations (fathatan).

The decision tree construction may also have an impact on what is the best unit for grid generation. In limited resource conditions state-specific [22, 23] decision trees may be preferred over state and grapheme specific [24] trees as they enable model synthesis for unseen graphemes. Although questions regarding grapheme identity may be asked, there is no guarantee that different graphemes may not end up in the same leaf node. This issue makes separation of these graphemes impossible, which may also complicate stimulated training when identical targets map to different regions on the grid.

Thus, the use of 'pure' phone or grapheme sets may not be the best choice for grid generation. Although, the state-specific decision tree issue may not be easy to address it is possible to examine the usefulness of extra phone/grapheme information.

4. EXPERIMENTS

Experiments in this section were conducted on 7 development languages and 1 surprise, Georgian, language of the IARPA Babel program in the option period 3.[1] Table 1 provides basic information about each language. For all languages an

Language	Family	System	Script	Graphemes
Pashto	Indo-European	Abjad	Arabic	47
Guarani	Tupian	Alphabet	Latin	71†
Igbo	Niger-Congo	Alphabet	Latin	52†
Amharic	Afro-Asiatic	Abugida	Ethiopic	247
Mongolian	Mongolic	Alphabet	Cyrillic	66†
Javanese	Austronesian	Alphabet	Latin	52†
Dholuo	Nilo-Saharan	Alphabet	Latin	52†
Georgian	Kartvelian	Alphabet	Mkhedruli	33

Table 1. Summary of languages used in this study

automatic, unicode based, graphemic dictionary generation [1] was applied. 'Pure' graphemes are appended with position information and language dependent attributes. Scripts marked with † utilise capital letters. Amharic graphs represent consonant-vowel sequences where vowels are clearly marked. Splitting each such graph into two yields 77 graphemes including singleton graphs.

A full language pack (FLP) was used for each language. This consists of 40 hours of conversational telephone speech (CTS). An additional 10 hours are available for development. Language models (LM) are standard n-grams and recurrent neural networks (RNN) trained using the CUED RNN LM toolkit [25]. These were trained on acoustic data transcripts containing about 500,000 words. Additional n-gram LMs were trained on data scraped by Columbia University from the internet [26]. These web LMs were then interpolated with the FLP LMs by optimising weights on the development data. Acoustic models are speaker adaptively trained Tandems

[1] Pashto IARPA-babel104b-v0.4bY, Guarani IARPA-babel305b-v1.0a, Igbo IARPA-babel306b-v2.0c, Amharic IARPA-babel307b-v1.0b, Mongolian IARPA-babel401b-v2.0b, Javanese IARPA-babel402b-v1.0b, Dholuo IARPA-babel403b-v1.0b, Georgian IARPA-babel404b-v1.0a

and (stacked) Hybrids which share the same set of features. Features are a concatenation of perceptual linear prediction coefficients [27], pitch [28], probability of voicing [28] and multi-language bottleneck (BN) features extracted by IBM and RWTH Aachen. These were trained on FLP data of 24 Babel languages and CTS data of 4 additional languages, English, Spanish, Arabic and Mandarin, released by LDC. IBM features are language independent whereas RWTH Aachen additionally fine-tuned their BN extractors to each target language. Thus a total of 4 acoustic models were built for each language as illustrated by Figure 3. Stacked

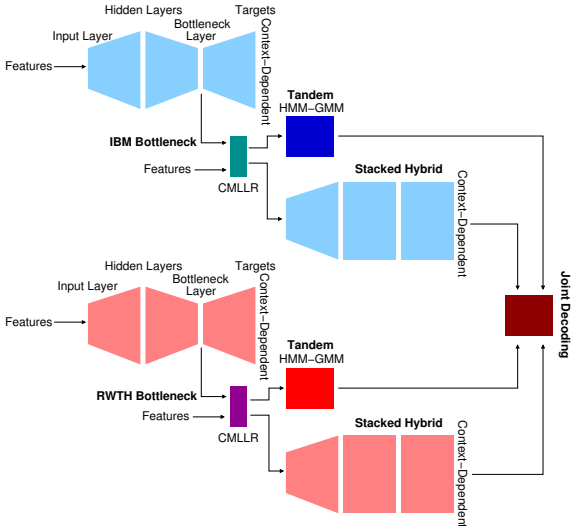


Fig. 3. 4-way Joint Decoding

Hybrids were trained with and without stimulated training using monophone initialisation followed by cross-entropy training and Minimum Phone Error training [24]. Unless otherwise stated, the grids for stimulated training were built using the sets of graphemes extended with position information and attributes. The regularisation term weight α in equation (5) was set to 0.1. In order to achieve high accuracy of transcription the final system combined all 4 acoustic models. In order to avoid decoding the data 4 times, a single joint decoding was used [4]. As shown in Figure 3, joint decoding combines acoustic models at test time. The combination is performed in the log-likelihood domain

$$\log(p(\mathbf{o}|s;\mathbf{\Lambda})) \leftarrow \alpha_1 \log(p(\mathbf{o}|s;\boldsymbol{\lambda}_1)) + \ldots + \alpha_N \log(p(\mathbf{o}|s;\boldsymbol{\lambda}_N)) \quad (10)$$

where α_1, ..., α_N are acoustic model weights set in this work to 0.125 for Tandem and 0.5 for Hybrid. The same approach, excluding Tandems, produced hypotheses refined with RNN LMs for speaker adaptation. Keyword search is performed using joint decoding lattices pruned to yield on average 20,000 $\frac{\text{arcs}}{s}$ densities. About 2,000 keywords available for each language [29]. The performance is measured using maximum term weighted value (MTWV).

The first experiment looked at the importance of position and attribute information for Pashto, which provides the most interesting set of attributes. A simpler cross-entropy Hybrid trained on RWTH Aachen BN features and FLP language model were used. The grid size is 32×32 which corresponds to 1024 activation functions. Table 2 summarises token error rate (TER) results for all possible combinations. Stimulated

Position	Attribute	Graphemes	TER (%)
–	–	–	48.4
✗	✗	37	48.0
✓	✗	107	48.0
✗	✓	49	48.0
✓	✓	137	48.1

Table 2. Impact of position and attribute information on stimulating training ASR performance in Pashto.

training shown on lines 2-5 shows gains over standard training shown on line 1. Among different combinations of word position and attribute information it seems that simpler grapheme sets with position or attribute only or none information show marginally better results. Such results may be explained by a rather small size of training data which does not permit robust representations to be derived that discriminate well.

The second experiment compared standard and stimulated training on all languages in a more challenging configuration combining 4 acoustic models and interpolated FLP and web data LMs in a single joint decoding run. The overall MTWV results are presented alongside in-vocabulary (IV) and out-of-vocabulary (OOV) query only results. Such a split is useful to assess whether an additional sub-word decoding is needed to improve performance on OOV queries which are otherwise searched in a generally less accurate phone index. The results

Language	Stimulated	TER (%)	MTWV		
			IV	OOV	Total
Pashto	✗	44.6	0.4720	0.3986	0.4644
	✓	44.4	0.4752	0.4032	0.4672
Guarani	✗	45.2	0.5823	0.5614	0.5800
	✓	44.9	0.5885	0.5712	0.5869
Igbo	✗	55.3	0.4007	0.3673	0.3974
	✓	55.1	0.4020	0.3680	0.3986
Amharic	✗	41.1	0.6500	0.5828	0.6402
	✓	40.8	0.6619	0.5935	0.6521
Mongolian	✗	47.8	0.5382	0.4805	0.5316
	✓	47.6	0.5497	0.4910	0.5431
Javanese	✗	50.9	0.4991	0.4448	0.4924
	✓	50.7	0.5024	0.4679	0.4993
Dholuo	✗	38.5	0.6547	0.5551	0.6434
	✓	38.3	0.6563	0.5585	0.6451
Georgian	✗	39.4	0.7184	0.7066	0.7179
	✓	38.9	0.7275	0.7197	0.7265

Table 3. Stimulated training performance on all languages

in Table 3 show that ASR gains are seen even after system combination for all languages. Similarly, gains can be seen in KWS performance for all languages which can be as small as 0.0012 for Igbo and as large as 0.0119 for Mongolian.

Experiments have so far examined a 32×32 grid. In order to assess whether stimulated training scales with increasing the grid size another experiment was performed on the 4 most challenging languages. The use of a larger 45×45 grid in Table 4 shows ASR and KWS gains for all languages. Further increase in the grid size for the most challenging language, Igbo, shows little benefit. The results in Tables 3 and 4

Language	Grid	TER (%)	MTWV		
			IV	OOV	Total
Pashto	32×32	44.4	0.4752	0.4032	0.4672
	45×45	43.8	0.4828	0.4083	0.4750
Igbo	32×32	55.1	0.4020	0.3680	0.3986
	45×45	54.7	0.4071	0.3680	0.4026
	55×55	54.6	0.4079	0.3555	0.4024
Mongolian	32×32	47.6	0.5497	0.4910	0.5431
	45×45	46.8	0.5606	0.5171	0.5559
Javanese	32×32	50.7	0.5024	0.4679	0.4993
	45×45	50.5	0.5043	0.4679	0.5001

Table 4. Impact of grid size on four most challenging languages.

illustrate advantages of stimulated training which results in good ASR and KWS gains across all examined languages.

5. CONCLUSIONS

Limited resource conditions cause generalisation issues for training neural network acoustic models. It is also hard to regularise these models as relationships between quantities such as targets and activations are distributed and hard to interpret. One exception is a stimulated training which enforces spatial ordering such that different phones cause different activations. A total of 8 limited resource languages have been considered confirming the benefits of such training against strong baselines. This paper has also discussed options for selecting the set of phones or graphemes which may be extended with additional information such as position, tone, stress, diacritic, etc. Finally, it confirmed that such training produces not only better 1-best hypotheses but also lattices by showing improved performance in keyword search tasks for all examined languages.

6. REFERENCES

[1] M. J. F. Gales, K. M. Knill, and A. Ragni, "Unicode-based graphemic systems for limited resource languages," in ICASSP, 2015.

[2] X. Liu, Model complexity control and linear projections for large vocabulary speech recognition, Ph.D. thesis, Cambridge University, 2005.

[3] G. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," Neural Computation, vol. 18, pp. 1527–1554, 2006.

[4] H. Wang, A. Ragni, M. J. F. Gales, K. M. Knill, P. C. Woodland, and C. Zhang, "Joint decoding of tandem and hybrid systems for improved keyword spotting on low resource languages," in Interspeech, 2015.

[5] A. Ghoshal, P. Swietojanski, and S. Renals, "Multilingual training of deep neural networks," in ICASSP, 2013, pp. 7319–7323.

[6] P. Bell and S. Renals, "Regularization of context-dependent deep neural networks with context-independent multi-task training," in ICASSP, 2015.

[7] J. Cui, B. Kingsbury, B. Ramabhadran, A. Sethy, K. Audhkhasi, X. Cui, E. Kislal, L. Mangu, M. Nussbaum-Thom, M. Picheny, Z. Tüske, P. Golik, R. Schlüter, H. Ney, M. Gales, K. Knill, A. Ragni, H. Wang, and P. Woodland, "Multilingual representations for low resource speech recognition and keyword search," in ASRU, 2015.

[8] R. Caruana, "Multitask learning: A knowledge-based source of inductive bias," Machine Learning, vol. 28, pp. 41–75, 1997.

[9] K. Veselý, M. Hannemann, and L. Burget, "Semi-supervised training of deep neural networks," in ASRU, 2013.

[10] K. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in Interspeech, 2015.

[11] N. Jaitly and G. E. Hinton, "Vocal tract length perturbation (VTLP) improves speech recognition," in ICML, 2013.

[12] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep neural network acoustic models," IEEE/ACM TASLP, vol. 23, no. 9, pp. 1469–1477, 2015.

[13] G. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," in NIPS, 2012.

[14] S. Tan, K. C. Sim, and M. J. F. Gales, "Improving the interpretability of deep neural networks with stimulated learning," in ASRU, 2015.

[15] C. Wu, P. Karanasou, K. C. Sim, and M. J. F. Gales, "Stimulated deep neural network for speech recognition," in Interspeech, 2016.

[16] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," Sig Proc Mag, vol. 29, pp. 82–97, 2012.

[17] A. Graves, N. Jaitly, and A.-R. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in ASRU, 2013.

[18] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in ASRU, 2013.

[19] A.-R. Mohamed and G. Hinton, "Understanding how deep belief networks perform acoustic modelling," in ICASSP, 2012.

[20] D. Palaz, M. Magimai.-Doss, and R. Collobert, "Analysis of cnn-based speech recognition system using raw speech as input," in Interspeech, 2015.

[21] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," J. MLR, vol. 1, pp. 1–49, 2008.

[22] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," in Eurospeech, 1999.

[23] B. Mimer, S. Stüker, and T. Schultz, "Flexible deciscion trees for grapheme based speech recognition," in ESSV, 2004.

[24] S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, A. Ragni, V. Valtchev, P. C. Woodland, and C. Zhang, The HTK Book (for HTK Version 3.5), University of Cambridge, http://htk.eng.cam.ac.uk, 2015.

[25] X. Chen, X. Liu, Y. Qian, M. J. F Gales, and P. C. Woodland, "CUED-RNNLM – an open-source toolkit for efficient training and evaluation of recurrent neural network language models," in ICASSP, 2016.

[26] G. Mendels, E. Cooper, V. Soto, J. Hirschberg, M. Gales, K. Knill, A. Ragni, and H. Wang, "Improving speech recognition and keyword search for low resource languages using web data," in Interspeech, 2015.

[27] H. Hermansky, "Perceptual Linear Predictive (PLP) analysis of speech," Journal of the Acoustic Society of America, vol. 87, no. 4, pp. 1738–1752, 1990.

[28] P. Ghahremani, B. BabaAli, D. Povey, K. Riedhammer, J. Trmal, and S. Khudanpur, "A pitch extraction algorithm tuned for automatic speech recognition," in ICASSP, 2014.

[29] J. Cui, J. Mamou, B. Kingsbury, and B. Ramabhadran, "Automatic keyword selection for keyword search development and tuning," in ICASSP, 2014.