BUT 2014 Babel System: Analysis of adaptation in NN based systems

Martin Karafiát, František Grézl, Karel Veselý, Mirko Hannemann, Igor Szoke, Jan "Honza" Černocký

Brno University of Technology, Speech@FIT and IT4I Center of Excellence, Brno, Czech Republic

karafiat,grezl,vesely,ihannema,szoke,cernocky@fit.vutbr.cz

Abstract

Features based on a hierarchy of neural networks with compressive layers – Stacked Bottle-Neck (SBN) features – were recently shown to provide excellent performance in LVCSR systems. This paper summarizes several techniques investigated in our work towards Babel 2014 evaluations: (1) using several versions of fundamental frequency (F0) estimates, (2) semi-supervised training on un-transcribed data and mainly (3) adapting the NN structure at different levels. They are tested on three 2014 Babel languages with full GMM- and DNN-based systems. Separately and in combination, they are shown to outperform the baselines and confirm the usefulness of bottle-neck features in current ASR systems.

Index Terms: speech recognition, discriminative training, bottle-neck neural networks, deep neural networks, adaptation of neural networks, fundamental frequency

1. Introduction

This paper presents our recent effort to build an automatic Keyword Spotting (KWS) system for Spring 2014 Babel evaluations based on Automatic Speech Recognition (ASR) front-end. We focus on Stacked Bottle-Neck (SBN) features [1], which was recently shown to be a superior architecture for Neural Network (NN) based feature extraction. SBN feature extraction involves two NNs (see figure 1): the bottle-neck (BN) outputs from the first one are stacked, down-sampled, optionally adapted, and taken as an input vector for the second NN. This second NN has again a BN layer, of which the outputs are taken as input features for a conventional Gaussian Mixture Model - Hidden Markov Model (GMM-HMM) speech recognition system. More detailed description of SBN can be found in section 2.1. In this paper, we describe and analyze our recent development centered around the SBN features:

Fundamental frequency (F0) related features were found to be important features in speech recognition systems for both tonal languages non-tonal languages [2]. We experiment with different F0 features as additional inputs to SBN. In section 2.2, we compare F0 features obtained from several different pitch estimators and we experiment with their combinations.

Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense US Army Research Laboratory contract number W911NF-12-C-0013. 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 work was also partly supported by, Technology Agency of the Czech Republic grant No. TA01011328 and by IT4Innovations Centre of Excellence CZ.1.05/1.1.00/02.0070. M. Karafiat was supported by Grant Agency of the Czech Republic post-doctoral project No. P202/12/P604.

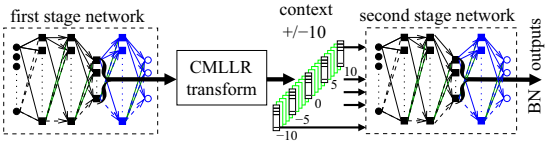

Figure 1: *Stacked Bottle-Neck Neural Network feature extraction.*

In our recent work [3], we have experimented with a Semi-supervised training (SST) allowing us to train on data with no available manual transcriptions. Transcriptions automatically generated with a "seed" ASR system were used for this purpose. We have applied such SST to train both the SBN NN parameters and the GMM-HMM system parameters. Much larger gains were obtained in the former case. In section 3.1, we show that further gains can be obtained with "fine-tuning", where the SST trained SBN NN is further re-trained on the supervised (manually transcribed) data only in a few iterations with a small learning rate.

We have also recently experimented with speaker adaptation applied in the context of SBN features. In [4], we have found that a very promising approach is to apply CMLLR adaptation to the bottle-neck output in the first stage of the SBN architecture (see figure 1). In section 3.2, we further analyze this approach and we compare it to the more conventional NN adaption strategy, where CMLLR transformation is applied to the original features (PLP in our case) at the NN input.

Finally, in section 4, we show that the improvement obtained with the SBN architecture can be very successfully transferred to Deep Neural Network (DNN) based speech recognition systems [5]. Specifically, the output from the first stage of the speaker adapted and SST trained SBN is used as the input to DNN. Significant gains were obtained with this architectures compared to our previous best SST DNN-HMM system [6]. Note that bottle-neck was already successfully used as the input to DNN in [7]. In this work, however, we are more interested in the improvements obtained from the SBN adaptation and SST.

2. Data and baseline experiments

The IARPA Babel Program data[1] simulate a case of what data one could collect in a limited time for a completely new language: it consists of two parts: scripted (speakers read text through telephone channel) and conversational (spontaneous telephone conversations). The *dev* data contains conversational speech only. Two training scenarios are defined for each language – Full Language Pack (FLP), where all collected data are available for training; and Limited Language Pack (LLP) which

[1] Collected by Appen http://www.appenbutlerhill.com

Table 1: *Evaluation data statistics. The LM and dictionary statistics are taken from LLP which is used to train HMM system. The OOV rate is reported with respect to LLP.*

Language	Bengali	Haiti	Lao
FLP hours	74.1	72.3	71.6
LLP hours	8.9	7.9	8.1
LM words	84334	93131	93328
dictionary	9497	5333	3856
# tied states	1310	1257	1453
dev hours	6.9	7.4	6.6
OOV rate [%]	8.5	4.1	1.8

consists only of one tenth of FLP, but the remaining part can be used for unsupervised training. Vocabulary and language model (LM) training data are also defined with respect to the Language Pack. They consists of transcripts of the given data pack.

In this work, all experiments are done for 3 different languages: Bengali IARPA-babel103b-v0.4b, Haitian Creole IARPA-babel201b-v0.2b and Lao IARPA-babel203b-v3.1a. The systems are trained on LLP only or LLP + untranscribed data. The statistics for the evaluation languages are given in Table 1. Further information can be found in [8]. The reported amounts of data for FLP and LLP refer to the speech segments after dropping the silence.

Our speech recognition system is HMM-based on cross-word tied-states triphones, it is trained from scratch using standard maximum likelihood training. Final word transcriptions are decoded using 3-gram Language Model (LM) trained only on the transcriptions of training data.

Mel-PLP features are generated in classical way, the resulting number of coefficients is 13. Deltas, double- and in the HLDA system [9] also triple-deltas are added, so that the feature vector has 39 and 52 dimensions, respectively. Cepstral mean and variance normalization is applied with the means and variances estimated per conversation side. HLDA is estimated with Gaussian components as classes to reduce the dimensionality to 39.

2.1. SBN feature extraction

The NN input features are 24 critical band energies (squared FFT magnitudes binned by Mel-scaled filter-bank and logarithmized) concatenated with estimates of F0. BUT F0 has 2 coefficients (F0 and probability of voicing), snack F0 is just single F0 and Kaldi F0 are 3 coefficients (Normalized F0 across sliding window, probability of voicing and delta). The FFV is a 7 dimensional vector. Therefore, the whole feature vector has 37 coefficients.

Conversation-side based mean subtraction is applied on the speaker basis and 11 frames are stacked together. Hamming window followed by DCT consisting of 0th to 5th base are applied on the time trajectory of each parameter (37x6) resulting in 222 coefficients at the first stage NN input (see fig. 1).

The first-stage NN has four hidden layers with 1500 units each except the BN layer. The BN layer is the third hidden layer and its size is 80 neurons. Its outputs are stacked over 21 frames (+/-10) and down-sampled (every 5 is taken) before entering the second-stage NN. This NN has the same structure and sizes of hidden layers as the first one. The size of BN layer is 30 neurons and its outputs are the final outputs forming the BN features for the recognition system. Neurons in both BN layers have linear activation functions as they were reported to provide better performance [10]. The NN targets are triphone states obtained by forced alignment of training data. To train the system on Bottle-Neck features, the BN outputs are transformed by Maximum Likelihood Linear Transform (MLLT), which considers HMM states as classes. Finaly, the GMM models are trained by Single Pass Retraining from initial PLP system.

Table 2: *Analysis of F0 feature extractors as additional input to NN (%WER).*

System	Bengali	Haiti	Lao
No F0	70.6	67.0	65.5
BUT F0	70.9	66.8	64.3
(B) BUT F0+pVocing	70.6	66.7	64.2
GetF0	70.5	66.9	64.4
KaldiF0	70.1	66.5	63.3
KaldiF0+pVoicing	69.7	65.7	62.8
(K) KaldiF0+pVoicing+Delta	69.5	65.6	62.6
FFV	70.0	66.6	64.2
(B)+(K)	69.7	65.5	62.3
(B)+(K)+FFV	**69.2**	65.4	62.3
(B)+(K)+FFV+GetF0	69.3	**65.4**	**62.2**

2.2. Analysis of F0 features

Fundamental frequency (F0) related features are important in speech recognition systems for tonal languages. Recently, F0 was found useful also for non-tonal languages [2]. In this work, we experiment with different F0 features as an additional input to SBN. Since the pitch estimators can suffer from robustness issues [11], we decided to examine four different pitch estimators. The first three estimators are based on normalized cross-correlation function: (1) BUT F0 was implemented according to [12]. (2) GetF0 is a tool using snack library[2] and (3) Kaldi F0 was recently implemented in Kaldi toolkit[3] [13]. Beside the F0 value, BUT F0 and Kaldi F0 also provide probability of voicing (pVoicing) as an additional features. The last estimators provides (4) Fundamental Frequency Variations (FFV), which continuous vector-valued representation of F0 variation. It is obtained by comparing the harmonic structure of the frequency magnitude spectra of the left and right half of an analysis frame [14].

Table 2 presents the results for SBN with no F0 features (i.e. only the filter bank energies form the SBN input) and with various combinations of F0 features used as the additional SBN input. The systems were based on simple Maximum Likelihood (ML) trained GMM-HMMs. As can be seen, Kaldi F0 provides the largest improvement as a stand-alone F0 estimator. Even thought SBN sees a context of +/- 15 frames, the delta F0 features providing an extra contextual information are still helpful in the case of Kaldi F0 estimator. This can be caused by the different postprocessing of the straight and delta F0 features as explained in [13].

The best performance was obtained when fusing F0 features form all the estimators, which corresponds to the SBN features used in the following sections. Interestingly, we clearly see that the F0 features are not only effective for the tonal languages (Lao), but also for non-tonal languages (Bengali and Haiti).

[2] `http://www.speech.kth.se/snack`
[3] `http://kaldi.sourceforge.net`

3. GMM-HMM SBN System

The final system is based on feature level fusion by Region Dependent Transform (RDT) [15]. Three feature streams PLP-HLDA (39 dimensions), SBN features (30 dim.) and BUT F0 with delta and acceleration coefficients (3 dim.) are concatenated, which results in 72 dimensional feature stream (called PLP-NN-F0 in the following text). Then, new models are trained by single-pass retraining from PLP basic system. 12 Gaussian components per state were found to be sufficient for these features. The models and features serve as a starting point for RDT training.

In RDT framework, an ensemble of linear transformations is trained with the discriminative Minimum Phone Error (MPE) criterion. Each transformation corresponds to one region in feature space partitioned by a GMM. According to our previous experiments [16], GMM with 125 components was chosen.

Our RDT settings performs dimensionality reduction from 72 to 69 which gives us better convergence than full feature vector. The final GMM-HMM system was trained using MPE [17] on top of RDT features. This system is denoted later in text as MPE-RDT or MPE-SAT-RDT, depending on the adaptation technique. The whole SBN system training can be described as follows:

1. Training of the initial ML PLP models, which is used to estimate PLP-HLDA transform and to generate triphone state targets for NN training
2. Training of SBN Neural Net and the whole MPE-RDT system. This system is used as a First-Pass system for speaker adaptation purposes.
3. The SBN NN is cut after First Stage NN and this 80-dimensional feature stream is adapted by speaker-based CMLLR (BN CMLLR). Consequently, the Second Stage NN is re-trained in SAT fashion [4] (SATNN).
4. The new adapted SBN features generate new PLP-SATNN-F0 feature stream which is further speaker-adapted by CMLLR. Further, RDT system is also trained in SAT fashion. [4]

Note, the NN features are adapted twice. On 1st stage NN level and on whole concatenated feature stream, too. We also experimented with system based on two separately adapted features streams, PLPCMLLR and SATNN, with no further adaptation in RDT level. Table 4 shows 0.5% improvement against the two separately adapted features stream. Therefore, we stick with original architecture.

This system is used to generate unsupervised transcripts on untranscribed data (the rest of FLP set) and also to rebuild triphone state NN targets on training data. This procedure was found effective especially on tonal languages because the initial targets are generated with plain PLP only [4].

3.1. Semi-supervised training

SST allowing us to train on data with no available manual transcriptions. Transcriptions automatically generated with a "seed" ASR system are used for this purpose. Such automatic transcriptions are naturally erroneous due to many reasons such as imperfect acoustic model, OOVs or poor language model. Thus it is important to select sentences with reasonable transcriptions. We use utterance-level confidence defined as a weighted average of non-silence word confidences in the segment: $C_{utt} = \frac{1}{T}\sum_{w=1}^{W} t^w C_{max}^w$, where W is number of words, C_{max}^w is word confidence measure [18], t^w is the length of the word in frames and T is the length of all the non-silence words.

The $(C_{utt} > 0.5)$ rule was used in our experiments as it was found in [3] as a safe value for the NN training (it covers about 70% of untranscribed data).

Table 3 presents 3.3-4.6% absolute improvement coming from SBN trained on SST data ($SST^{(1)}$). For simplicity, we use GMM-HMM ML trained on SBN features extracted only from LLP data (i.e. SST is applied only for SBN training; not for GMM-HMM training). The improvements were encouraging. Therefore, we experimented with regeneration of unsupervised data by system based on $SST^{(1)}$ NN. The new NN $SST^{(2)}$ was trained, which gave further $\sim 0.5\%$ absolute improvement.

Next, we experimented with "fine-tuning" of SST SBN by re-training it further only on the transcribed data with the learning rate set to one tenth of its original value. Only the second stage NN is tuned to keep the training process simple and fast. Table shows 3 additional $\sim$ 0.5% absolute improvement obtained with this procedure.

Table 3: *Effect of adding unsupervised data.*

System	Bengali %WER	Haiti %WER	Lao %WER
LLP	69.5	65.4	61.7
$SST^{(1)}$	66.2	61.1	56.9
$SST^{(2)}$	65.6	59.5	55.6
$SST^{(2)} \rightarrow LLP$	65.1	59.1	55.0

Table 4: *Various adaptation composition on final system.*

System	Bengali %WER
MPE-SAT-RDT$^{(2)}$ PLP-SATNN-F0	60.5
MPE-RDT$^{(2)}$ PLPCMLLR-SATNN-F0	61.0

3.2. PLP vs. FBANK in CMLLR adaptation of NN

Common approach to train NN based system is to build system on top of PLP or MFCC features due to the straightforward speaker adaptation although FBANK are known to work better [19]. Table 5 presents direct comparison of SBN architecture with (SATNN) and without (NN) the proposed adaptation technique on FBANK, PLP or PLP with speaker adaptation as the input features. Note that all features were concatenated with F0 features to have fair comparison. NNs were trained on $SST^{(2)}$ transcriptions but evaluated GMM-HMM systems were trained for simplicity on SBN features only (no MPE-RDT).

Table 5 shows that SATNN approach is complementary to the common approach based on PLP-CMLLR, about 1% absolute improvement was reached with second NN based adaptation. When we compare our proposed technique (FBANK in SATNN column) with common approach (PLP-CMLLR in NN), about 1% absolute improvement is observed. This results guided us to try this first stage BN CMLLR features also as an input for our HMM-DNN system.

[4] Note, that our previous experiments showed a marginal effect of VTLN on the PLP feature stream if CMLLR was used, therefore, VTLN was not applied for simplicity.

Table 5: *PLP vs FBANK in SAT NN system (%WER).*

System	Bengali		Haiti		Lao	
	NN	SATNN	NN	SATNN	NN	SATNN
FBANK	65.2	63.7	59.5	57.1	55.6	53.1
PLP	65.7	64.3	59.5	57.5	55.2	53.0
PLPCMLLR	64.8	63.9	58.4	57.2	54.2	52.9

4. DNN system

The DNN training followed the recipe described in [20] including SST. Baseline input DNN features were based on PLPs augmented with KaldiF0. These features were mean/variance normalized, spliced by +/- 4 frames next to the central frame and projected down to 40 dimensions using linear discriminant analysis (LDA) and MLLT. Moreover, speaker adaptive training (SAT) was done using a single feature-space CMLLR transform estimated per speaker. These CMLLR features were spliced using context of +/- 5 frames, and shifted / rescaled in order to have zero mean and unit variance on the DNN input. For all the experiments, we used the same DNN topology: 6 hidden layers, where each hidden layer has 2048 neurons with sigmoids, 440 inputs (11x40). The hidden layers of DNN were initialized with stacked Restricted Boltzmann Machines (RBMs) that were pre-trained in a greedy layer-wise fashion [21]. After pre-training, we added the output layer with random weights and we performed frame-classification training (we classify frames into triphone tied-states). We used mini-batch Stochastic Gradient Descent (SGD) to minimize per-frame cross-entropy between the labels and network output. Finally, the seed network was re-trained by sequence-discriminative training by optimizing Sequence Minimum Bayes Risk (sMBR) objective [22]. This aimed to maximize expected frame accuracy of being in a correct state. The expectation was calculated over the possible state sequences represented by lattices. The reference sequences were obtained by force-alignment to transcription.

This DNN was used as a seed system for further semi-supervised training. Unlabeled training data were automatically transcribed and per-frame confidences were produced. The confidences were based on re-scaled frame-by-frame state posteriors (extracted from lattices using forward-backward algorithm) that were selected by the best-paths' state-sequence from the lattice.

The semi-supervised training is applied to the per-frame Cross-Entropy training. For the sMBR training, we used the transcribed LLP dataset. Large part of the improvement from the per-frame Cross-Entropy semi-supervised training was preserved also after the sMBR training. Table 6 presents very close performance of GMM-HMM and baseline PLP CMLLR based DNN system.

4.1. CMLLR NN features in DNN system

Finally, we extended DNN system by the proposed first stage BN features with speaker adaptation. These features were stacked in context +/-10, downsampled by factor 5 (like SBN) and fed into DNN system. Table 6 presents excellent gains of 3.3-4.5% over the baseline PLP CMLLR features caused probably by SST of input features, effective speaker adaptation, and also general structure of these features (NN learning vs. basic feature extraction). Moreover, SST training on top of SST trained features gains 1.4-2.1% (compared to 2.2-3.0% gains obtained with PLP features). Therefore, these techniques are quite complementary.

Table 6: *Final ASR systems (%WER).*

System	Bengali	Haiti	Lao
GMM MPE-SAT-RDT - SST			
PLP-SATNN-F0	60.5	53.8	50.7
DNN sMBR			
PLP CMLLR	62.0	57.4	53.2
BN CMLLR	58.2	52.9	49.9
DNN sMBR - SST			
PLP CMLLR	59.8	54.4	50.8
BN CMLLR	**56.8**	**50.8**	**48.3**

5. KWS systems

The SST systems were also evaluated on final KWS task to show consistency in improvement. It is the standard word based lattice search approach [23], where the raw score of each putative hit is posterior probability of a sequence of words (links) in the lattice representing the particular term. These raw scores are then normalized and additionaly fused accroding to [24]. The KWS results are in table 7, where the numbers represent Maximum Term Weighted Value (MTWV) [25] of development and evaluation terms (approx. 5000 terms) on *dev* data. The consistent improvement over 3% from using the novel features in the DNN systems can be seen. Moreover, different architectures of GMM-HMM and DNN-HMM system provided complementary systems for fusion.

Table 7: *Final KWS systems*

System	Bengali %MTWV	Haiti %MTWV	Lao %MTWV
GMM MPE-SAT-RDT	32.02	45.38	45.94
DNN-SST PLP CMLLR	33.89	42.54	47.24
DNN-SST BN CMLLR	**37.08**	**45.47**	**50.20**
GMM+(DNN BN CMLLR)	**38.03**	**48.84**	**51.42**

6. Conclusions

The paper deals with multiple partial improvements in NN based systems. We presented gain with using multiple F0 estimators for tonal and also non-tonal languages. Next, we showed additional improvement by fine-tuning during the semi-supervised training.

The main part of the paper deals with CMLLR adaptation of NN. We showed impressive gain by using adapted Bottle-Neck features in GMM-HMM and also DNN-HMM based systems.

7. References

[1] Frantisek Grezl, Martin Karafiat, and Lukas Burget, "Investigation into bottle-neck features for meeting speech recognition," in *Proc. Interspeech 2009*, 2009, number 9, pp. 2947–2950.

[2] Florian Metze, Zaid A. W. Sheikh, Alex Waibel, Jonas Gehring, Kevin Kilgour, Quoc Bao Nguyen, and Van Huy Nguyen, "Models of tone for tonal and non-tonal languages.," in *ASRU*. 2013, pp. 261–266, IEEE.

[3] Grezl F., Karafiat M., and Vesely K., "Adaptation of neural network feature extractor for new language," in *in Proceedings of ASRU 2013*, Olomouc, Czech Republic, 2013.

[4] Martin Karafiát, František Grézl, Mirko Hannemann, and Jan "Honza" Černocký, "BUT neural network features for spontaneous vietnamese in BABEL," in *Accepted for: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, Florence, Italy, May 2014, IEEE.

[5] Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel-Rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara Sainath, and Brian Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *Signal Processing Magazine*, pp. 14–22, 2012.

[6] Karel Veselý, Mirko Hannemann, and Lukáš Burget, "Semi-supervised training of deep neural networks," in *Proc. of ASRU 2013*, Dec 2013.

[7] Jonas Gehring, Wonkyum Lee, Kevin Kilgour, Ian R. Lane, Yajie Miao, and Alex Waibel, "Modular combination of deep neural networks for acoustic modeling.," in *INTERSPEECH*, Frdric Bimbot, Christophe Cerisara, Ccile Fougeron, Guillaume Gravier, Lori Lamel, Franois Pellegrino, and Pascal Perrier, Eds. 2013, pp. 94–98, ISCA.

[8] M. Harper, "The BABEL program and low resource speech technology," in *Proc. of ASRU 2013*, Dec 2013.

[9] N. Kumar and A. G. Andreou, "Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition," *Speech Communication*, pp. 283–297, 1998.

[10] Karel Veselý, Martin Karafiát, and František Grézl, "Convolutive bottleneck network features for LVCSR," in *Proceedings of ASRU 2011*, 2011, pp. 42–47.

[11] A. de Cheveigné and H. Kawahara, "Yin, a fundamental frequency estimator for speech and music," *J. Acoust. Soc. Am.*, vol. 111, pp. 1917–1930, 2002.

[12] D. Talkin, "A robust algorithm for pitch tracking (RAPT)," in *Speech Coding and Synthesis*, W. B. Kleijn and K. Paliwal, Eds., New York, 1995, Elseviever.

[13] Pegah Ghahremani, Bagher BabaAli, Daniel Povey, Korbinian Riedhammer, Jan Trmal, and Sanjeev Khudanpur, "A pitch extraction algorithm tuned for automatic speech recognition," in *Accepted for: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, Florence, Italy, May 2014, IEEE.

[14] Kornel Laskowski, Mattias Heldner, and Jens Edlund, "The fundamental frequency variation spectrum," 2008.

[15] Bing Zhang, Spyros Matsoukas, and Richard Schwartz, "Recent progress on the discriminative region-dependent transform for speech feature extraction," in *Proc. of Interspeech 2006*, Pittsburgh, PA, USA, Sep 2006, pp. 2977–2980.

[16] Martin Karafiát, František Grézl, Mirko Hannemann, Karel Veselý, and Jan "Honza" Černocký, "BUT BABEL System for Spontaneous Cantonese," in *Proceedings of Interspeech 2013*, 2013, pp. 2589–2593.

[17] D. Povey, *Discriminative training for large vocabulary speech recognition*, Ph.D. thesis, University of Cambridge, 2003.

[18] Frank Wessel, Ralf Schlüter, Klaus Macherey, and Hermann Ney, "Confidence measures for large vocabulary continuous speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 9, pp. 288–298, 2001.

[19] Abdel-Rahman Mohamed, Geoffrey E. Hinton, and Gerald Penn, "Understanding how deep belief networks perform acoustic modelling.," in *ICASSP*. 2012, pp. 4273–4276, IEEE.

[20] Karel Vesel, Mirko Hannemann, and Luk Burget, "Semi-supervised training of deep neural networks," in *Proceedings of ASRU 2013*. 2013, pp. 267–272, IEEE Signal Processing Society.

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

[22] Karel Veselý, Arnab Ghoshal, Lukáš Burget, and Daniel Povey, "Sequence-discriminative training of deep neural networks," in *Proceedings of Interspeech 2013*. 2013, pp. 2345–2349, International Speech Communication Association.

[23] Igor Szöke, Petr Schwarz, Lukáš Burget, Michal Fapšo, Martin Karafiát, Jan Černocký, and Pavel Matějka, "Comparison of keyword spotting approaches for informal continuous speech," in *Interspeech'2005 - Eurospeech - 9th European Conference on Speech Communication and Technology*, 2005, pp. 633–636.

[24] Karakos D., Schwartz R., Tsakalidis S., Zhang L. Ranjan S., Ng T., Hsiao R., Saikumar G., Bulyko I., Nguyen L., Makhoul J., Grezl F., Hannemann M., Karafiat M., Szoke I. Vesely K., Lamel L., and Le V.-B., "Score normalization and system combination for improved keyword spotting," in *Proc. of ASRU 2013*, Dec 2013.

[25] J. G. Fiscus, J. Ajot, J. S. Garofolo, and G. Doddington, "Results of the 2006 Spoken Term Detection Evaluation," 2007, pp. 45–50.