DISCRIMINATIVE SEMI-SUPERVISED TRAINING FOR KEYWORD SEARCH IN LOW RESOURCE LANGUAGES

Roger Hsiao[1], Tim Ng[1], František Grézl[2], Damianos Karakos[1],
Stavros Tsakalidis[1], Long Nguyen[1] and Richard Schwartz[1]

[1]Raytheon BBN Technologies, Cambridge, MA, USA
[2]Brno University of Technology, Speech@FIT and IT4I Center of Excellence, Czech Republic
{whsiao,tng,dkarakos,stavros,ln,schwartz}@bbn.com, grezl@fit.vutbr.cz

ABSTRACT

In this paper, we investigate semi-supervised training for low resource languages where the initial systems may have high error rate ($\geq 70.0\%$ word eror rate). To handle the lack of data, we study semi-supervised techniques including data selection, data weighting, discriminative training and multi-layer perceptron learning to improve system performance. The entire suite of semi-supervised methods presented in this paper was evaluated under the IARPA Babel program for the keyword spotting tasks. Our semi-supervised system had the best performance in the OpenKWS13 surprise language evaluation for the limited condition. In this paper, we describe our work on the Turkish and Vietnamese systems.

Index Terms— semi-supervised training, low resource languages, keyword spotting

1. INTRODUCTION

Semi-supervised training is an important area for speech recognition and its applications. Given a relatively small amount of supervised (transcribed) data, the goal is to improve system performance using unsupervised data. Since collecting supervised data can be costly [1, 2, 3], a robust semi-supervised training procedure can significantly reduce the cost of developing a speech-to-text (STT) system. This is especially true for many low resource languages where tools and language experts are not immediately available.

Performing semi-supervised training can be challenging for low resource languages since without enough data on the target language, one may expect the bootstrap system using only the supervised data is suboptimal. Under the IARPA Babel program, we focus on rapidly developing speech technologies for new languages with limited resources, say 10 hours of transcribed audio. We notice that the initial systems under such condition often have over 70% word error rate (WER). Therefore, we revisit semi-supervised training and explore techniques which may help in this condition.

The basic approach for semi-supervised training is first building a bootstrap model using some supervised data, and then using this model to transcribe the unsupervised data. This automatically transcribed data is then used to supplement the supervised data for building the final model. While the process is similar to supervised training, additional steps like data selection [1] are often applied to select data with high confidence for the transcription. Since the performance of the bootstrap system is not ideal, the automatic transcription may contain mostly errors. In addition, using transcriptions with high error rates may have more impact on discriminative training, which tries to minimize the errors against the reference transcriptions. The difficulties of semi-supervised discriminative training have been discussed in [3, 4, 5].

In this paper, we aim to improve semi-supervised training for low resource languages, where initial systems may have high WER ($\geq 70\%$). We propose confidence weighted training which uses a confidence model to select data and also weigh the supervised and unsupervised data. Then, in the context of semi-supervised training, we study the advantages and disadvantages of two widely used objective functions: Minimum Phone Error (MPE) [6] and Boosted Maximum Mutual Information (BMMI) [7] for discriminative training, and propose an optimization algorithm for robust discriminative training. In addition, we investigate semi-supervised Multi-Layer Perceptron (MLP) training. The entire suite of semi-supervised methods presented in this paper was evaluated under the IARPA Babel program for the keyword spotting tasks. Our semi-supervised system had the best performance in the OpenKWS13 surprise language evaluation for the limited condition.

2. BABEL PROGRAM AND SYSTEM DESCRIPTION

The IARPA Babel program is a research program for rapid development of keyword spotting systems for low resource languages. In the first year of the program, Cantonese, Pashto, Turkish and Tagalog were used as the development languages and Vietnamese was chosen to be the surprise language for open evaluation. The evaluation has different conditions and one of them is the limited condition which consists of 10 hours of transcribed audio and roughly 90 hours of unsupervised data. The audio data is mainly conversational speech between two persons in a telephone channel, but each lan-

guage pack also comes with a small amount of read speech. The telephone channels can be landlines, different kinds of cellphones, or phones embedded in vehicles, and the sampling rate is 8000 Hz. The development set for each language consists of roughly 10 hours of conversational telephone speech. The evaluation set, given by IARPA, contains 15 hours of speech for each language, except Vietnamese, which has around 75 hours of data. In this paper, we evaluate our approaches on the IARPA Babel Program Turkish language collection release (babel105b-v0.4) and Vietnamese language collection release (babel107b-v0.7), and we report our results on the development set. Table 1 summarizes the data of these two languages. For Vietnamese, we attached the tones to the vowels and created a phone set of 123 phones.

	Turkish	Vietnamese
vocab	11.5k	3.1k
# phones	51	123
text data	100k	110k
transcribed data	10-hr	10-hr
untranscribed data	90-hr	90-hr
tonal	no	yes
OOV on dev	23.0%	8.5%

Table 1. Turkish and Vietnamese data in limited condition

For keyword spotting, each language has two set of keywords: a development keyword list and an evaluation keyword list. The development keyword list contains 300 to 1000 keywords which were selected by the performers for development. The evaluation keyword lists consists of 3000 to 4000 keywords, and they were given during the evaluation. Each keyword may contain several words and it may or may not be in the training vocabulary. The performance of a keyword spotting system is measured by the Actual Term Weighted Value (ATWV) and WER is also measured for the underlying STT system. ATWV is computed by,

$$ATWV = 1 - \frac{1}{K}\sum_{w=1}^{K}\left(\frac{\#miss(w)}{\#ref(w)} + \beta\frac{\#fa(w)}{T - \#ref(w)}\right) \quad (1)$$

where K is the number of keywords; $\#miss(w)$ is the number of true keyword tokens that are not detected; $\#fa(w)$ is the number of false alarms; $\#ref(w)$ is the number of words in reference; T is the number of trials (e.g., seconds in the audio), and β is a constant set at 999.9. The details and the design of this metric are available in [8].

The BBN keyword spotting system is divided into several components. At a high level, the speech recognition system [9] is run to produce a detailed lattice of word hypotheses. This lattice is used to extract keyword hits with nominal posterior probability scores produced by various methods. Different extraction methods are necessary because, for example, we can use whole-word extraction methods for the known keywords but we must use phonetic extraction for the keywords that were not known when the recognizer was run. Also, multiple extraction methods help the system to be more robust for different languages. The scores are normalized so that they are consistent across keywords and so that they are good estimates of posteriors. Details of score normalization are available in [10].

3. CONFIDENCE WEIGHTED TRAINING

As shown in figure 1, the semi-supervised training in this work consists of two steps: (1) unsupervised data selection followed by (2) semi-supervised acoustic model training. The main differentiator from our previous semi-supervised training method [11] is the use of utterance-based confidence weights in acoustic model training.

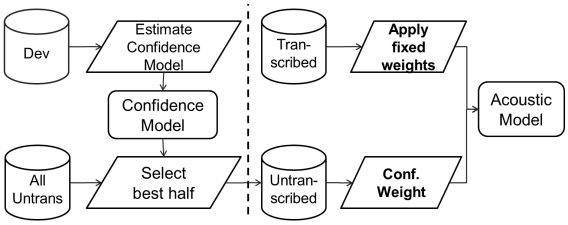

Fig. 1. Overview of the confidence weighted training

The data selection procedure for semi-supervised training is described in [11]. First, the untranscribed audio data is segmented into utterances using a speech activity detection system which is trained on the 10-hour training corpus using an architecture similar to [12]. It is then decoded using the system trained on the same 10-hour manually transcribed corpus. The confidence of each utterance is computed based on a confidence model trained on the development set. Finally, the best half of the utterances are selected according to their confidence scores for acoustic model training.

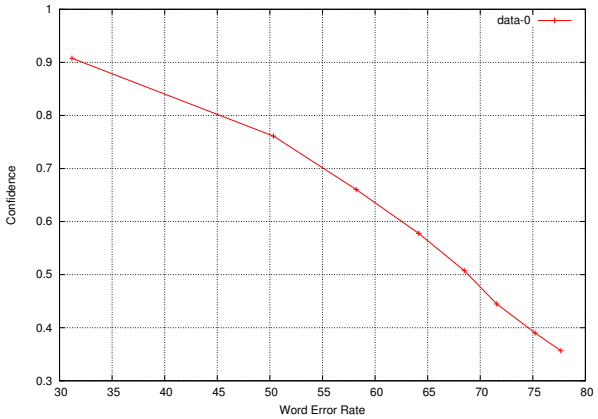

Fig. 2. Confidence score versus WER for the Turkish semi-supervised data using the 10-hr PLP-only baseline system

Although utterances with the worst confidence scores are excluded from the training, the quality of the selected automatic transcripts varies. The plot of confidence score against

WER for the selected 43 hours of the Turkish unsupervised data is shown in figure 2 and it is computedusing the 10-hour PLP-only STT system. The plot is created by grouping the utterances into eight bins according to their confidence scores and the average WER is computed for each of the bins. As shown in the figure, the average WER fluctuates from 31% to 77%, and the WERs are highly correlated to the confidence scores. To take into account the quality of the transcripts for acoustic model training, a weight is assigned to each of the utterance based on the confidence score using the following affine equation:

$$w_i = s \times c_i + b \tag{2}$$

where w_i is the weight for utterance i, s the slope, c_i confidence score for utterance i, b the bias. In this work, s is 2.0 and the average of the utterance-level weights is constrained to one. Hence, $b = 1 - \frac{\sum_{i=1}^{N} s \times c_i}{N}$ with N being the total number of utterances.

The WER on the Turkish development set from different semi-supervised training approaches shown in Table 2. By including the 43 hours of untranscribed data into acoustic model training a 2.9% absolute reduction in WER (from 70.9% to 68.0%) is obtained as compared to the 10-hour baseline system. An additional WER absolute reduction of 0.7% (from 68.0% to 67.3%) is observed by using weighted training in both ML and MPE training for the semi-supervised training for Turkish.

System	sup. : unsup.	WER(%)
10-hour baseline	-	70.9
semi-supervised	unweighted	68.0
semi-supervised	4 : 1	67.3

Table 2. WER of the Turkish MPE PLP systems using confidence weighted training

4. SEMI-SUPERVISED DISCRIMINATIVE TRAINING

Discriminative training can significantly improve acoustic models, but its application to semi-supervised training has been limited [3]. One of the reasons is because both numerator and denominator statistics are collected from recognition output. Hence, discriminative training may not be able to identify the true errors and adjust the model parameters accordingly. This also becomes more serious if the initial model does not achieve good performance. In such a case, the numerator lattices may have high error rate and give adverse effects for discriminative training. Reference [5] suggests a different view that although both numerator and denominator statistics are artificial and contain errors, the numerator lattice, which represents the reference, is generated by a strong language model, while the denominator lattice is often produced by a weak language model. Therefore, one would expect the numerator lattice may have better accuracy than the denominator lattice and discriminative training may work under semi-supervised training. To compensate the difference between supervised and unsupervised data, discriminative training is further explored in [4]. In which, unlabeled data is used for regularization and it is weighted with the discriminative criterion.

In this study, we would first compare MPE and BMMI, and study the behaviors of these two objective functions in semi-supervised training. Then, we reinvestigate the optimization problem for semi-supervised discriminative training and explore how we could improve the performance.

4.1. MPE, BMMI and the EBW algorithm

MPE [6] and BMMI [7] are both considered to be the state-of-the-art objective functions for discriminative training. MPE aims to optimize the acoustic model for the phone error rate while BMMI maximizes the margin between the Viterbi state sequences of the references and the competing sequences [13]. Although the target is different, two objective functions are similar as shown in equation 3 and 4,

$$F_{\text{MPE}}(\theta) = \sum_i \frac{\sum_{W_i'} P(X_i|W_i';\theta)P(W_i')A(W_i',W_i)}{\sum_{W_i'} P(X_i|W_i';\theta)P(W_i')} \tag{3}$$

$$F_{\text{BMMI}}(\theta) = \log \prod_i \frac{P(X_i|W_i;\theta)P(W_i)}{\sum_{W_i'} P(X_i|W_i';\theta)P(W_i')e^{-b\times A(W_i',W_i)}} \tag{4}$$

where X is the observation; W_i and W_i' are the i-th reference and competing hypothesis respectively; A is the an accuracy function to compare W_i and W_i'; θ represents the model parameters.

Although BMMI and MPE are similar and both discriminative training procedures often give similar improvements [7], the behavior can be different under semi-supervised training, and the reason is due to the extended Baum-Welch algorithm (EBW). During optimization, EBW does not operate on the MPE or the BMMI objective function directly. Instead, EBW operates on a regularized objective function. As shown in [14] and [4], EBW uses KL-divergence for regularization,

$$G_{\text{MPE}}(\theta) = F_{\text{MPE}}(\theta) + D(\theta)\text{KL}(\theta^0||\theta) \tag{5}$$

$$G_{\text{BMMI}}(\theta) = F_{\text{BMMI}}(\theta) + D(\theta)\text{KL}(\theta^0||\theta)\ . \tag{6}$$

where θ^0 is a backoff model which is often the ML model or the model from the previous EM iteration; D is a Gaussian specific constant to control the weight of the regularization. The value of D is often computed by a heuristic where D is the maximum of E times the occupancy of the denominator statistics or twice the value required to keep the covariance to be positive definite [6]. Then, the value of E is tuned empirically and it is often set between 1.0 to 2.0.

The computation of D is different between MPE and BMMI due to the way they compute the denominator statistics. For BMMI, the denominator statistics are collected from the entire denominator lattice which is generated by the

recognizer. While for semi-supervised training, we cannot attach the reference path to the lattice, but only the 1-best, the statistics remain similar. However, for MPE, whether an arc would contribute to numerator or denominator statistics depends on whether the expected phone accuracy of all paths going through this arc is higher than, or lower than the expected phone accuracy of the entire lattice [6]. As a result, this would affect the value of D and thus, the weight of the regularization. In the worst case, D can be too small which may lead weak regularization, or the regularization is too strong. In sum, we argue that the lack of references in the unsupervised data may have more impact on MPE compared to BMMI, hence, we investigate on this issue.

The results on comparing MPE and BMMI on semi-supervised training is available in table 3. As shown by the results, BMMI outperforms MPE for the unweighted semi-supervised training while the performance is roughly the same for the weighted semi-supervised training and the 10-hour supervised training. These results support our claim that BMMI may have some advantages over MPE in semi-supervised training. BMMI and MPE have roughly the same performance for 10-hour supervised training since the calculation of D for MPE is not affected by the unsupervised data, and for weighted training, since the utterances with low confidence have small weights, their contribution to the denominator statistics is reduced as well. Therefore, MPE benefits more using confidence weighted training.

System	Obj	sup. : unsup.	WER(%)
10-hr supervised	MPE	-	70.9
10-hr supervised	BMMI	-	71.0
semi-supervised	MPE	unweighted	68.0
semi-supervised	BMMI	unweighted	67.1
semi-supervised	MPE	4:1	67.3
semi-supervised	BMMI	4:1	67.1

Table 3. WER of the Turkish PLP systems with MPE and BMMI discriminative training

4.2. Robust model estimation for discriminative training

As discussed in [14], the backoff model (θ^0 in equation 5 and 6) used in discriminative training does not necessarily need to be the ML model or the model from the previous EM iteration. Instead, one could plug in any model for regularization, and it acts as a constraint that the final model should be close to the backoff model in terms of KL-divergence. Given that we have two types data: supervised and unsupervised data, we propose that we may estimate the model using the supervised data only, given that the output model should be close to the estimate using the entire unsupervised and supervised data.

The motivation of this approach is that while discriminative training would benefit more from the supervised data, the small amount of supervised data may not be sufficient for reliably estimating the model parameters. Hence, we enforce that the output model should be close to the estimate using the entire data set, which is likely to be more robust. The implementation of this procedure is simple: one only needs to accumulate the statistics from the supervised and unsupervised data separately during the E-step, then, update the model twice with different backoff models,

$$\mu_j^{(1)} = \frac{\sum_{t\in U+S}\gamma_t^n(j)x_t - \sum_{t\in U+S}\gamma_t^d(j)x_t + D_j^{(1)}\mu_j^0}{\sum_{t\in U+S}\gamma_t^n(j) - \sum_{t\in U+S}\gamma_t^d(j) + D_j^{(1)}} \quad (7)$$

$$\mu_j^{(2)} = \frac{\sum_{t\in S}\gamma_t^n(j)x_t - \sum_{t\in S}\gamma_t^d(j)x_t + D_j^{(2)}\mu_j^{(1)}}{\sum_{t\in S}\gamma_t^n(j) - \sum_{t\in S}\gamma_t^d(j) + D_j^{(2)}}, \quad (8)$$

where μ_j is the j-th Gaussian mean; x_t is the observation at time t; $\gamma_t^n(j)$ and $\gamma_t^d(j)$ are the posterior probability of j-th Gaussian at time t in the numerator and the denominator lattice respectively; U represents the set of unsupervised data while S represents the set of supervised data. As a result, $\mu_j^{(2)}$ is estimated using the supervised data and it would be close to $\mu_j^{(1)}$, which is estimated using the entire data set. Similarly, the covariance matrices can also be updated in the same way.

This approach is similar to speaker adaptation in speech recognition, where we take a speaker independent model trained with large amount of data and adapt it using a small amount of data from the target speaker. In fact, if we rewrite equation 8 as,

$$\mu_j^{(2)} = \frac{\sum_{t\in S}\gamma_t^n(j)x_t - \sum_{t\in S}\gamma_t^d(j)x_t + D_j\mu_j^0 + \tau\mu_j^{(1)}}{\sum_{t\in S}\gamma_t^n(j) - \sum_{t\in S}\gamma_t^d(j) + D_j + \tau} \quad (9)$$

then it is equivalent to the discriminative MAP adaptation (DMAP) as described in [15], when $D_j = 0$ and $\tau = D_j^{(2)}$. In such a case, $\mu_j^{(1)}$ is treated as a prior model and we adapt the model using the supervised data. The difference in our approach is that now the prior model is also iteratively updated under the EM framework. Figure 3 shows the overall procedure of this proposed discriminative training for semi-supervised data.

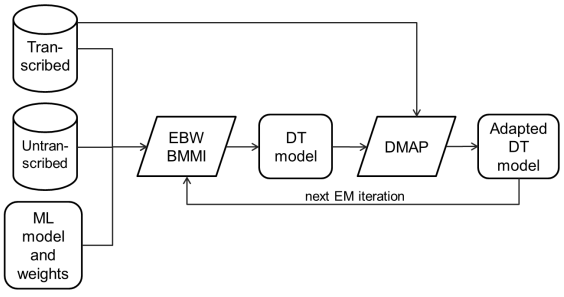

Fig. 3. Overview of our proposed semi-supervised discriminative training

Table 4 shows the performance of our proposed semi-supervised discriminative training on the PLP system. In this experiment, we also explored different adaptation strategies like the prior model was built on the unsupervised data and

adapted the model using the supervised data ($U \rightarrow S$); or the prior model was built on the supervised data and adapted the model using the entire data set ($S \rightarrow S+U$). The best strategy was the one we described which the prior model was built using entire data set and we adapted the model using the supervised data ($S + U \rightarrow S$). All BMMI results in the table used confidence weighted training, and E was set to 2.0 for the first M-step and 6.0 for the second M-step. We obtained 0.4% absolute reduction in WER compared to the BMMI baseline. While the improvement was modest, our proposed training required little extra computation since computing $\mu_j^{(2)}$ did not require reprocessing the data but simply running another M-step on top of $\mu_j^{(1)}$.

System	Obj	M-step	WER(%)
10-hr supervised	BMMI	S	71.0
semi-supervised	BMMI	$S + U$	67.1
semi-supervised	BMMI	$U \rightarrow S$	67.0
semi-supervised	BMMI	$S \rightarrow S + U$	67.0
semi-supervised	BMMI	$S + U \rightarrow S$	66.7

Table 4. WER of the Turkish PLP system with different semi-supervised discriminative training strategies

5. SEMI-SUPERVISED MULTI-LAYER PERCEPTRON TRAINING

MLP training has shown to be effective in reducing the error rate [16]. In the Babel program, we investigate whether semi-supervised training would help MLP performance. Details of the semi-supervised MLP training is available in [17].

In brief, an initial MLP is trained using the supervised data and an STT system is built using these features. This STT system is then used to transcribe the unsupervised data and measure the confidence. The confidence is for data selection where only the data with confidence above certain threshold is kept for the semi-supervised training. Finally, the semi-supervised MLPs are trained using a mix of manually and automatically transcribed data. Figure 4 is an overview of this procedure

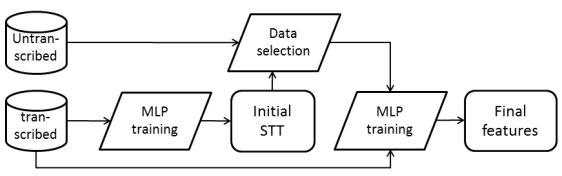

Fig. 4. Overview of the semi-supervised MLP training

Table 5 shows the performance of using semi-supervised MLP features. In this experiment, all systems used confidence weighted training and semi-supervised discriminative training. As shown in the results, the initial MLP features trained only with supervised data already improved the baseline PLP semi-supervised system by 7.0% absolute in WER. Using semi-supervised training gave an additional improvement of 1.1% absolute.

System	MLP training	WER(%)
PLP baseline	-	66.7
initial MLP	S	59.7
semi-supervised MLP	$S + U$	58.6

Table 5. WER of the Turkish semi-supervised systems using PLP and different MLP features

6. EXPERIMENTAL RESULTS

We evaluate our proposed semi-supervised training methods in the Babel Turkish and Vietnamese evaluations. Under the limited resource condition, the supervised data has about 10 hours of transcribed audio and 90 hours of untranscribed audio. The development sets of these two languages consist of 10 hours of data, and in this experiment, we report the WER and ATWV of the dev sets for these languages.

For the semi-supervised system, we first built a system using the MLP features trained on the 10-hr supervised data prepared. This system, trained solely on supervised data, is used as the baseline system and used to transcribe the unsupervised data. Then, we performed the confidence weighted training, semi-supervised discriminative training and also the semi-supervised MLP training for the final systems. Table 6 shows the improvement of each technique in terms of WER and ATWV. All keyword search results are under the known keyword condition which assumes keywords are known before decoding.

	Turkish		Vietnamese	
System	WER	ATWV	WER	ATWV
10-hr MLP sys.	62.3%	35.2%	60.3%	39.4%
+cw training	60.2%	37.7%	59.0%	40.8%
+semi. sup. DT	59.7%	38.3%	58.7%	41.0%
+semi. sup. MLP	58.6%	39.2%	55.2%	45.6%

Table 6. Performance of Turkish and Vietnamese systems

The results show that semi-supervised training can improve both speech recognition and keyword search performance. Compared to the systems trained with only 10 hours of supervised data, semi-supervised training improves the Turkish system by 3.7% and 4.0% absolute in WER and ATWV respectively. For the Vietnamese system, it improves by 5.1% absolute in WER and 6.2% absolute in ATWV.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we propose confidence weighted training, semi-supervised discriminative training and semi-supervised MLP

training, to help semi-supervised training for low resource languages and high WER environment. As shown in the Turkish experiments, while the initial system has a high WER of 70.9%, applying semi-supervised training and also MLP training can improve the system significantly to 58.6% WER. For the Vietnamese system, we start from the 10-hour supervised only MLP system with an WER of 60.3% and an ATWV of 39.4%. Our proposed semi-supervised training improves the system to 55.2% WER and 45.6% ATWV.

In the future, we will try to apply semi-supervised training to the deep neural network, and extend our semi-supervised training to language modeling. Out of vocabulary is also an important issue for semi-supervised training which we will also explore techniques which may help in this direction.

8. ACKNOWLEDGMENTS

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.

9. REFERENCES

[1] K. Yu, M.J.F. Gales, L. Wang, and P.C. Woodland, "Unsupervised Training and Directed Manual Transcription for LVCSR," *Speech Communications*, vol. 52, no. 7–8, pp. 652–663, 2010.

[2] F. Wessel and H. Ney, "Unsupervised Training of Acoustic Models for Large Vocabulary Continuous Speech Recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 1, pp. 23–31, 2005.

[3] L. Wang, M.J.F. Gales, and P.C. Woodland, "Unsupervised Training for Mandarin Broadcast News and Conversation Transcription," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2007, vol. 4, pp. 353–356.

[4] X. Cui, J. Huang, and J.T. Chien, "Multi-View and Multi-Objective Semi-Supervised Learning for HMM-Based Automatic Speech Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 7, pp. 1923–1935, 2012.

[5] B. Strope, D. Beeferman, A. Gruenstein, and X. Lei, "Unsupervised Testing Strategies for ASR," in *Proceedings of the INTERSPEECH*, 2011.

[6] D. Povey and P.C. Woodland, "Minimum Phone Error and I-smoothing for Improved Discriminative Training," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2002, vol. 1, pp. 105–108.

[7] D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. Visweswariah, "Boosted MMI for Model and Feature-space Discriminative Training," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2008, pp. 4057–4060.

[8] "OpenKWS13 Keyword Search Evaluation Plan," http://www.nist.gov/itl/iad/mig/upload/OpenKWS13-EvalPlan.pdf, 2013.

[9] T. Ng, B. Zhang, S. Matsoukas, and L. Nguyen, "Region Dependent Transform on MLP Features for Speech Recognition," in *Proceedings of the INTERSPEECH*, 2011.

[10] D. Karakos et. al., "Score Normalization and System Combination for Improved Keyword Spotting in Speech," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013.

[11] J. Ma and R. Schwartz, "Unsupervised versus supervised training of acoustic models," in *Proceedings of the INTERSPEECH*, 2008.

[12] T. Ng, B. Zhang, L. Nguyen, S. Matsoukas, X. Zhou, N. Mesgarani, K. Veselý, and P. Matějka, "Developing a Speech Activity Detection System for the DARPA RATS Program," in *Proceedings of the INTERSPEECH*, 2012.

[13] G. Saon and D. Povey, "Penalty Function Maximization for Large Margin HMM Training," in *Proceedings of the INTERSPEECH*, 2008.

[14] R. Hsiao and T. Schultz, "Generalized Baum-Welch Algorithm and Its Implication to a New Extended Baum-Welch Algorithm," in *Proceedings of the INTERSPEECH*, 2011.

[15] D. Povey, P.C. Woodland, and M.J.F. Gales, "Discriminative MAP for Acoustic Model Adaptation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2003, vol. 1, pp. 312–315.

[16] M. Karafiát, F. Grézl, M. Hannemann, K. Veselý, and J. H. Černocký, "BUT BABEL System for Spontaneous Cantonese," in *Proceedings of the INTERSPEECH*, 2013.

[17] F. Grézl and M. Karafiát, "Semi-supervised bootstrapping approach for neural network feature extractor training," submitted to ASRU 2013.