INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Semi-supervised DNN training with word selection for ASR

Karel Veselý, Lukáš Burget, Jan "Honza" Černocký

Brno University of Technology, Speech@FIT and IT4I Center of Excellence, Czech Republic
iveselyk@fit.vutbr.cz

Abstract

Not all the questions related to the semi-supervised training of hybrid ASR system with DNN acoustic model were already deeply investigated. In this paper, we focus on the question of the granularity of confidences (per-sentence, per-word, per-frame), the question of how the data should be used (data-selection by masks, or in mini-batch SGD with confidences as weights). Then, we propose to re-tune the system with the manually transcribed data, both with the 'frame CE' training and 'sMBR' training.

Our preferred semi-supervised recipe which is both simple and efficient is following: we select words according to the word accuracy we obtain on the development set. Such recipe, which does not rely on a grid-search of the training hyper-parameter, generalized well for: Babel Vietnamese (transcribed 11h, untranscribed 74h), Babel Bengali (transcribed 11h, untranscribed 58h) and our custom Switchboard setup (transcribed 14h, untranscribed 95h). We obtained the absolute WER improvements 2.5% for Vietnamese, 2.3% for Bengali and 3.2% for Switchboard.

Index Terms: semi-supervised training, DNN, word selection, granularity of confidences

1. Introduction

The current ASR systems require relatively large data-sets to be trained on. These need to be recorded and manually transcribed, which is slow and costly. For some rare languages it might be even difficult to find native annotators.

On the other hand, we can save a lot of time and other resources, if only a part of the data is transcribed manually and a larger part is transcribed automatically by decoding. The decoding is done with a 'seed' ASR system trained with the manually transcribed data, while typically we also generate some confidences. Of course, the automatic transcripts are not perfect, but still, they can be used to improve the performance of the acoustic model by the semi-supervised training (i.e. training with the mixed data: manually transcribed and automatically transcribed). This is sometimes also referred as 'self-learning', as the ASR system is re-trained with its own outputs. The confidences express the certainty of the decoded labels, and we will use them to filter or assign weights to the training data.

The self-training was initially studied by NLP community for 'word-sense disambiguation' [1] or for 'syntactic parsing' [2]. Then, for the GMM-HMM systems it was studied in the journal article [3]. In the proposed scenario, the GMM-HMM model is trained, while the per-word confidences are used to select the frames for which the word-confidence was higher than a threshold. The word-boundaries are dynamic as the alignment is updated during the training.

Whereas in [4, 5] the GMM-HMM training data are selected per-sentence, choosing the sentences which are believed to have the WER smaller than the overall WER of the development set. An interesting observation was made that the self-training is more efficient if the automatic transcripts are generated with the larger language model.

The semi-supervised training of the NN-based bottleneck-feature extractors was studied in [6, 7, 8, 9], here the data were selected per-sentence. In the mismatched scenario of US English and European English, it was found helpful to reduce the number of NN-outputs [9]. Also, it is good to post-process the NN after the semi-supervised training by 're-tuning', in which we train only with the manually transcribed data. We can either train a new output layer [6] or re-train the network with a small learning rate [8].

In our earlier work with self-training of hybrid DNN-HMM system [10], we obtained good results with the frame selection done according to the per-frame 'lattice-posterior' confidences. The frames with confidence above a threshold were selected for the 'frame CE' training (mini-batch SGD training with cross-entropy loss), while the DNN was 're-tuned' with the manually transcribed data by the 'sMBR' training.

Eventually, instead of 're-tuning', we could use a topology with 2 output layers, where the 1st output layer is trained by the manually transcribed data and the 2nd output layer is trained with the automatic transcripts [11, 12].

In [12], the untranscribed data are used even for the sequence-discriminative training. The sMBR loss function is used for the manually transcribed data and the 'Negative Conditional Entropy (NCE)' is used for the untranscribed sentences. The NCE reduces the confusion of the lattice-paths by fostering the more likely paths. However, we did not achieve any improvement by adding NCE training to our setup.

Yet another possibility to improve the semi-supervised training is to use the multi-system transcripts from system combination [13], or for the 'agreement analysis' [14]. Also having the 'captions' available can be helpful [15, 16].

To our best knowledge, the literature does not compare the 'Data selection' strategies done on the level of a) sentences, b) words or c) frames. All the three approaches are possible. We previously compared the sentence-selection with frame-selection in [10]. However, the sentence confidences contained a bug and we present the updated results in table 1. In this article, we will also compare the training with the 'Data selection' or the 'Data weighting'. To make sure we get the best improvements from the self-training, we 're-tune' the models with the manually transcribed data before the final conclusions are formulated.

2. Confidences

2.1. Per-word confidence (MBR statistics)

We use the 'MBR confidence', which is calculated as the statistics $\gamma(q, s)$ from the Minimum Bayes Risk (MBR) decoding [17, section 7.1]. The quantity $\gamma(q, s)$ is the probability with which the word-symbol s is present at position q in the output word-sequence. We simply take the words from the best-path in lattice and calculate their γ's as their confidences. This

http://dx.doi.org/10.21437/Interspeech.2017-1385

MBR confidence is the default word confidence implemented in Kaldi. Yet another method to obtain word-confidences is the 'NN-posterior' confidence [9]. This is based on averaging of frame-by-frame log-posteriors of senone states along the state-sequence of the word. These were in [9] averaged into per-sentence confidences.

2.2. Per-sentence confidence (average word-confidence)

The per-sentence confidence c_{sent} is typically calculated as the average of the word confidences [4, 5, 6, 9]: $c_{sent} = \frac{1}{N}\sum_{i=1}^{N} c_{w_i}$ It is good to think about it as an estimate of the word accuracy in a sentence. For the analysis, we use both the 'MBR confidences' and the 'NN-posterior' confidences. For self-training experiments we use only 'MBR confidences'.

2.3. Per-frame confidence (lattice-posterior)

In our previous work [10], we advocated for using the frame-by-frame 'lattice-posterior' confidences and SGD training with frame-selection. The frame-level confidence c_{frame_i} is extracted from the lattice posteriors $\gamma(i, s)$, which express the probability of being in state s at time i. For each frame i, the confidence is $c_{frame_i} = \gamma(i, s_{1best,i})$, where the state $s_{1best,i}$ is taken from the best-path in lattice. The posteriors γ are computed using the forward-backward algorithm on the lattice.

2.4. Analysis of confidences

In figure 1, we compare the accuracy of the automatic transcripts, which are selected according to various confidences. We are selecting words starting from the high confidence (horizontal axis) and we measure the WER in the selected subset (vertical axis). As can be seen, the sentence selection (words are added per whole sentences) with 'NN-posterior' confidence is not ideal (yellow curve). In our system we have a DNN with 4599 outputs, which make it difficult to obtain good confidences from NN-posteriors. Much better results are obtained with the MBR confidences, which are either used for sentence-selection (red curve) or for word-selection (blue curve). We clearly see that the best subset is obtained by individual selection of words (blue curve). The dashed 'lower bound' curve is obtained by selecting all the correct words first.

3. Experimental setup

3.1. Dataset

In this paper, we report experiments on the Vietnamese dataset from the IARPA Babel program, release babel107b-v0.7. The training data consist of conversational telephone speech and a small part of prompted speech. The development set consists of conversational speech only. The data come from various telephone channels: landlines, different kinds of cellphones, or phones embedded in vehicles.

For the semi-supervised training we consider the Limited Language Pack (LimitedLP) scenario, in which 11 hours of data are transcribed, and 74 hours are 'untranscribed' (but we have the transcripts available for the analysis). The results are shown on the development set composed of 9.8 hours of data.

The Vietnamese phone set consists of 29 phonemes, which are marked with six different tones. For the triphone-tree clustering, we introduced a 'position in a word' feature and shared states across phonemes. The syllabic lexicon for LimitedLP condition contains 3k records, and the OOV rate on the dev-set is only 1.19%. We used a trigram language model with Kneser-Ney smoothing built on the training transcripts from the 11 hours, the model has 12k 3-grams and 47k 2-grams.

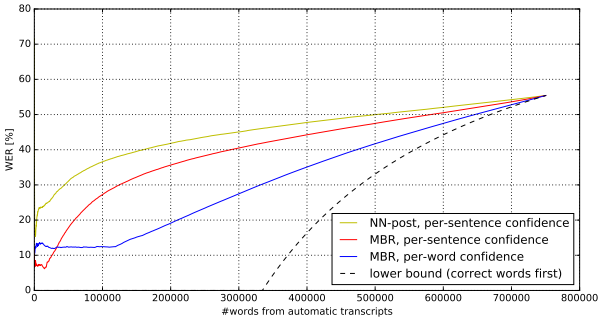

Figure 1: *WER in data selection according to per-sentence or per-word confidence (WER is computed without deletions and is normalized by #hyp-words. We can't select the 'deleted' word)*

3.2. The seed system

The DNN-HMM seed system is trained with the 11 hours of manually transcribed data. First, an auxiliary GMM system is used to generate the fMLLR features, which are the input of the main DNN model.

The GMM-HMM features are obtained by splicing +/- 4 frames of the 13-dimensional PLPs (includes C0) extended by 3 kaldi-pitch features [18]. All features are cepstral mean-variance normalized. The spliced features are projected to 40 dimensions with a global LDA+MLLT [19] linear transform and per-speaker fMLLR [20] linear transform. The auxiliary GMM-HMM system has 4599 cross-word triphone tied states and 5.6 Gaussians per state. It is used to prepare the initial DNN training targets.

The DNN has a standard feed-forward topology with 6 hidden layers of 2048 sigmoidal neurons. There is 440 dimensional input and 4599 dimensional softmax output. The 40 dimensional fMLLR features are spliced by +/- 5 frames and re-normalized to have zero mean and unit variance. We used RBM pre-training [21] to initialize the 6 hidden layers. Then, the 'frame CE' training, was done with mini-batch SGD (Stochastic Gradient Descent), in which the learning rate is halved from 3rd epoch till the convergence of the held-out loss. Finally, the network is re-trained by 4 epochs of sMBR training [22].

The seed system should be as good as possible to obtain the most accurate transcripts. The WER of our sMBR-DNN seed system is 59.6, measured on development set. This is a very competetive and fair baseline for the difficult Babel data.

4. Data selection

A new DNN is trained by 'frame CE' training with the mixed data: manually transcribed and automatically transcribed (decoded by the seed system). In the first set of experiments, we investigate into the question of the granularity of the confidences. We want to know, what is the ideal size of the 'data selection unit'.

Intuitively we expect that with smaller units, the data are selected more precisely. But, it might become more difficult to compute a reliable confidence value. In our 'Data selection' experiments, we use the weighted mini-batch SGD training with

Acknowledgements: The work was supported by Technology Agency of the Czech Republic project No. TA04011311 "MINT", European Union's Horizon 2020 project No. 645523 BISON, and by Czech Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project "IT4Innovations excellence in science - LQ1602".

Table 1: *Sentence selection (average MBR word-confidence)*

Added sentences	0%	30%	**50%**	**70%**	90%	100%
WER%	60.9	60.1	**59.8**	**59.8**	60.0	60.1

Table 2: *Word selection (per-word MBR confidence)*

Added words	0%	30%	**40%**	50%	100%	**Seed**
WER%	60.9	59.2	**59.1**	59.2	60.1	**59.6**

Table 3: *Frame selection (lattice-posterior confidence)*

Added frames	0%	50%	**60%**	**70%**	80%	100%
WER%	60.9	59.3	**59.1**	**59.1**	59.3	60.1

binary weights, which select the frames. The binary weights are multiplying the frame-by-frame gradients. We select the top N% of the units with the best confidence, while the selected frames have weight 1 and the rejected frames have weight 0.

4.1. Sentence selection

The most common approach in the literature is the selection of whole sentences [4, 5, 6, 9]. From table 1, we see that it is good to leave out 30-50% of sentences, which brings a 0.3% WER improvement compared to adding all the sentences.

4.2. Word selection

The word-selection can be found in GMM experiments from Wessel [3]. In our work, we select the top N% words. In table 2, we see that the word-selection leads to 0.7% better results than the sentence-selection in table 1. It is also interesting that the optimal amount of added words roughly corresponds to the word-accuracy of the seed system, which is $100 - 59.6 = 40.4$. Intuitively, this should select most of the correct words, and only some of the wrong words. Because this result might be by chance, we validate this 'simple word-selection' rule in section 7 with a very different experimental setup. In table 2, the WER of the seed system 59.6 is better than the training with 0% added words 60.9. This is because the seed system is trained with 'sMBR', while the other results are with 'frame CE' training.

4.3. Frame selection

The smallest possible unit for data-selection is the 'frame', the frames are produced with 10ms steps. In table 3 we select the frames according to the 'lattice-posterior' confidence. We see that the best frame-selection result is on-par with the best word-selection system in table 2. Nevertheless, it is more convenient to do the word-selection by word-confidences, as the word-confidences are represented more compactly than the frame-confidences.

Table 4: *'Data selection', re-tuning the initial models. Re-tuning is done with 11 hours of manually transcribed data, the initial model is built with mixed transcribed+untranscribed data.*

WER%	Initial model	Re-tuned	
		+ frame CE	+ sMBR
Sentence selection	59.8	58.7	57.5
Word selection	**59.1**	58.4	**57.1**
Frame selection	**59.1**	**58.3**	**57.1**
No confidence	60.1	58.7	57.6

Table 5: *Weighted sentences (average MBR word-confidnce)*

Scale α	1.0	2.0	2.5	3.0	**3.5**	4.0
WER%	59.8	59.6	59.6	59.5	**59.3**	59.5

Table 6: *Weighted words (per-word MBR confidence)*

Scale α	1.0	2.0	4.0	8.0	10.0	**12.0**	14.0
WER%	59.5	59.2	59.1	58.9	59.0	**58.8**	59.0

Table 7: *Weighted frames (lattice-posterior confidence)*

Scale α	1.0	2.0	3.0	4.0	**5.0**	6.0
WER%	59.4	59.1	59.0	59.1	**58.9**	59.0

5. Data weighting

Another possibility is to add all the untranscribed data, while the confidences are used as weights in the SGD training. The weights are used to scale the gradients from the individual frames. However, in this case, we need to be more careful about the actual values of the confidences. We found helpful to use the confidences c that reside in interval $(0, 1)$, while we tune the exponential scale α, that is applied as: $\hat{c} = c^{\alpha}$. The optimal α is found by training several NNs with different values α in a grid search. This may not be ideal in practical scenarios, but it is good for the analysis.

The α, which leads to the best results does not necessarily correspond to the 'ideally calibrated' confidences (i.e. the probability that the label of the unit is correct). For example, the ratio of the transcribed and untranscribed data might play an important role: In [10, table 5], it was helpful to repeat the manually transcribed sentences. However, the data repetition is no longer helpful after we start tuning the scale α. The results of weighting with per-sentence, per-word and per-frame confidences are in the tables 5, 6, 7. We see that 'weighted sentences' from table 5 are better than 'selected sentences' in table 1 (WER 59.8 $\rightarrow$ 59.3). Even better results are achieved with the per-frame or the per-word weights (WER 59.3 $\rightarrow$ 58.9 $\rightarrow$ 58.8).

The best result 58.8 was obtained with the per-word confidences, that were scaled by $\alpha = 12.0$. Such high value is leading to a 'soft' data-selection: a word with the original confidence 0.68 gets the training weight $w_i = 0.68^{12.0} \doteq 0.01$, which almost removed 43% words from the training by having the weight ≤ 0.01.

If we compare the results of the 'selected words' in table 2 with the 'weighted words' in table 6, the improvement is 59.1 $\rightarrow$ 58.8. Both approaches use a grid search. We either searched the N% words to add, or the optimal scale α. However, if we knew the correct N%, the word-selection would become more practical by avoiding the grid search, despite the 0.3% worse result. At the same time, it is unlikely for the best $\alpha = 12.0$ to

Table 8: *'Data weighting', re-tuning the initial models. Re-tuning is done with 11 hours of manually transcribed data, the initial model is built with mixed transcribed+untranscribed data.*

WER%	Initial model	Re-tuned	
		+ frame CE	+ sMBR
Sentence weighting	59.3	58.3	57.2
Word weighting	**58.8**	58.2	**56.9**
Frame weighting	58.9	**58.1**	57.0
No confidence	60.1	58.7	57.6

generalize for other datasets, which makes it difficult to remove the grid search from the word-weighting.

6. Re-tuning the systems

In literature, we can find that it is beneficial to 're-tune' the self-trained 'initial model'. In Thomas [6], the DNN output layer was discarded and trained again from random initialization with the manually transcribed data (in [6], the DNN was a bottleneck feature extractor).

We found that even better approach is to keep the output layer 'as-is' and continue training with the 11 hours of the manually transcribed data and a smaller initial learning rate (0.001 instead of original 0.008). This post-processing was described in Grézl [8] as 'fine-tuning' (we intentionally renamed it as 're-tuning' to avoid the confusion with the fine-tuning of RBMs that we used to build our seed system).

As can be seen in table 4 with 'Data selection' and table 8 with 'Data weighting', we repeated the same two-stage re-tuning, first with the 'frame CE' objective and then with 'sMBR' training. Both were done with the 11 hours of manually transcribed data. To verify that the confidences were helpful even after re-tuning, we also trained from the initial model (marked as 'No confidence'). Here we added all the automatically transcribed data, while we did not use any confidences. We clearly see that the confidences helped us get better results.

7. Finding a generic recipe

Ideally, we are interested in finding such semi-supervised training recipe, that will be effective for a broad range of scenarios. Until now, we explored the behavior for one language (Babel Vietnamese) and one scenario (11 transcribed hours, 74 hours are untranscribed).

We are searching for a universal recipe without the computationally expensive hyper-parameter tuning. In section 4.2, we saw that the best percentage of added words corresponds to the word accuracy of the seed system (in table 2 was good to add 40% words, while the word accuracy of the seed system was 40.4%). Surprisingly, this 'simple word-selection' rule generalized to other databases, see Babel Bengali in table 9 and Switchboard in table 10.

The initial Switchboard systems are re-tuned in table 11, starting from initial systems with: a) word-selection with optimal $N = 70\%$ words, b) word-weighting with optimal exponential scale $\alpha = 7.0$ or c) 'No confidence'. For Switchboard, the final performance of word-selection is 23.7, the same as if we did not use the confidences. But with word-weighting, the WER dropped to 23.5.

Table 9: *Word-selection with Babel Bengali, 11 transcribed hours, 58 untranscribed hours. Language model from 11 hours of transcripts. Word accuracy of seed system: 37.1%.*

Added words	0%	30%	**40%**	**50%**	60%	100%
WER%	64.2	62.5	**62.3**	**62.3**	62.4	63.2

Table 10: *Word-selection with modified Switchboard setup, 14 hours transcribed, 95 hours untranscribed. LM trained on Fisher transcripts. The results are for HUB5-2000 (Switchboard + CallHome). Word accuracy of seed system: 73.1%.*

Added words	0%	60%	**70%**	80%	90%	100%
hub5 WER%	28.0	25.1	**24.4**	24.7	24.5	24.8

Table 11: *Re-tuning the self-trained models (Switchboard). Re-tuning with 14 hours of manually transcribed data, the initial models are trained with mixed transcribed+untranscribed data.*

WER%	Initial model	Re-tuned	
		+ frame CE	+ sMBR
a) Word selection	**24.4**	24.2	23.7
b) Word weighting	**24.4**	**24.1**	**23.5**
c) No confidence	24.8	24.3	23.7

Table 12: *Final performance of the semi-supervised training based on 'simple word-selection'. The initial model is trained with the mixed transcribed+untranscribed data. The re-tuning is done with a smaller set of manually transcribed data.*

[WER%]	Vietnamese	Bengali	SWBD
Seed system (sMBR)	59.6	62.9	26.9
Initial model	59.1	62.3	24.4
+ re-tuned (frame CE)	58.4	61.6	24.2
+ re-tuned (sMBR)	57.1	60.6	23.7
Δ WER%	2.5	2.3	3.2

8. Conclusion

The overall WER improvements from the semi-supervised training become clear after re-tuning the 'simple word-selection' models for all the three databases (table 12). The absolute WER improvement between the seed sMBR system and the final sMBR systems is 2.5% for Babel Vietnamese, 2.3% for Babel Bengali and 3.2% for Switchboard. For Bengali and Vietnamese the seed WER is higher, so the absolute WER improvement from the semi-supervised training is smaller than in the case of Switchboard, which also had a larger language model from the Fisher transcripts. Our observations are summarized as follows:

- 'Data selection' is better done per words or frames, than per whole sentences ($\Delta = 0.4\%$ WER, see last column in table 4)
- 'Data weighting' leads to little better results than 'Data selection' ($\Delta = 0.2\%$ WER), while for weighting we tuned the exponential scale α (compare last columns in tables 4 and 8)
- A 'simple word-selection' setup without hyper-parameter tuning is as follows: We choose the amount (%) of the selected words with highest confidence according to the word accuracy on the development set. This simple rule selected the optimal amount of words for 3 databases: Babel Vietnamese, Babel Bengali, Switchboard
- Another experiment on Switchboard revealed that after the re-tuning, the system with 'simple word-selection' had the same performance as the 'no confidence' system (see table 11). For Babel Vietnamese the 'simple word-selection' was better by 0.5% than the 'no confidence' system (table 4)

Given this evidence, we conclude that the 'simple word-selection' is still a preferred technique. It does not involve any hyper-parameter tuning, and it is either helpful or causes no harm in the ASR system compared to adding all data with no use of confidences. We believe, that our findings are of high practical value. The untranscribed data are abundant and easy to obtain, while our proposed solution brings solid WER improvements and it is not difficult to replicate. $\diamond\diamond\diamond$

9. References

[1] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *33rd Annual Meeting of the Association for Computational Linguistics, 26-30 June 1995, MIT, Cambridge, Massachusetts, USA, Proceedings.*, 1995, pp. 189–196.

[2] D. McClosky, E. Charniak, and M. Johnson, "Effective self-training for parsing," in *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 4-9, 2006, New York, New York, USA*, 2006.

[3] 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.

[4] S. Novotney, R. M. Schwartz, and J. Z. Ma, "Unsupervised acoustic and language model training with small amounts of labelled data," in *Proc. of IEEE ICASSP*, 2009, pp. 4297–4300.

[5] S. Novotney and R. M. Schwartz, "Analysis of low-resource acoustic model self-training," in *Proc. of INTERSPEECH*, 2009, pp. 244–247.

[6] S. Thomas, M. L. Seltzer, K. Church, and H. Hermansky, "Deep neural network features and semi-supervised training for low resource speech recognition," in *Proceedings of ICASSP*, 2013, pp. 6704–6708.

[7] F. Grezl and M. Karafiat, "Semi-supervised bootstrapping approach for neural network feature extractor training," in *Proc. of ASRU*, 2013.

[8] F. Grézl and M. Karafiát, "Combination of multilingual and semi-supervised training for under-resourced languages," in *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*, 2014.

[9] P. Zhang, Y. Liu, and T. Hain, "Semi-supervised DNN training in meeting recognition," in *2014 IEEE Spoken Language Technology Workshop, SLT 2014, South Lake Tahoe, NV, USA, December 7-10, 2014*, 2014, pp. 141–146.

[10] K. Veselý, M. Hannemann, and L. Burget, "Semi-supervised training of deep neural networks," in *Proceedings of ASRU*, 2013, pp. 267–272.

[11] H. Su and H. Xu, "Multi-softmax deep neural network for semi-supervised training," in *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*, 2015, pp. 3239–3243.

[12] V. Manohar, D. Povey, and S. Khudanpur, "Semi-supervised maximum mutual information training of deep neural network acoustic models," in *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*, 2015, pp. 2630–2634.

[13] S. Li, Y. Akita, and T. Kawahara, "Semi-supervised acoustic model training by discriminative data selection from multiple ASR systems' hypotheses," *IEEE/ACM Trans. Audio, Speech & Language Processing*, vol. 24, no. 9, pp. 1524–1534, 2016.

[14] F. de Chaumont Quitry, A. Oines, P. J. Moreno, and E. Weinstein, "High quality agreement-based semi-supervised training data for acoustic modeling," in *2016 IEEE Spoken Language Technology Workshop, SLT 2016, San Diego, CA, USA, December 13-16, 2016*, 2016.

[15] B. Lecouteux, G. Linarès, P. Nocera, and J. Bonastre, "Imperfect transcript driven speech recognition," in *INTERSPEECH 2006 - ICSLP, Ninth International Conference on Spoken Language Processing, Pittsburgh, PA, USA, September 17-21, 2006*, 2006.

[16] H. Liao, E. McDermott, and A. W. Senior, "Large scale deep neural network acoustic modeling with semi-supervised training data for youtube video transcription," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, December 8-12, 2013*, 2013.

[17] H. Xu, D. Povey, L. Mangu, and J. Zhu, "Minimum bayes risk decoding and system combination based on a recursion for edit distance," *Computer Speech & Language*, vol. 25, no. 4, pp. 802–828, 2011.

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

[19] M. J. F. Gales, "Semi-tied covariance matrices for hidden Markov models," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 3, pp. 272–281, May 1999.

[20] D. Povey and G. Saon, "Feature and model space speaker adaptation with full covariance Gaussians," in *Proc. of INTERSPEECH*, 2006.

[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] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Proc. of INTERSPEECH'13*, 2013.