INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

PhonVoc: A Phonetic and Phonological Vocoding Toolkit

Milos Cernak, Philip N. Garner

Idiap Research Institute, Martigny, Switzerland
{Milos.Cernak,Phil.Garner}@idiap.ch

Abstract

We present the PhonVoc toolkit, a cascaded deep neural network (DNN) composed of speech analyser and synthesizer that use a shared phonetic and/or phonological speech representation. The free toolkit is distributed as open-source software under a BSD 3-Clause License, available at `https://github.com/idiap/phonvoc` with the pre-trained US English analysis and synthesis DNNs, and thus it is ready for immediate use.

In a broader context, the toolkit implements training and testing of the analysis by synthesis heuristic model. It is thus designed for the wider speech community working in acoustic phonetics, laboratory phonology, and parametric speech coding. The toolkit interprets the phonetic posterior probabilities as a sequential scheme, whereas the phonological posterior-class probabilities are considered as a parallel via K different phonological classes. A case study is presented on a LibriSpeech database and a LibriVox US English native female speaker. The phonetic and phonological vocoding yield comparable performance, improving speech quality by merging the phonetic and phonological speech representation.

Index Terms: speech vocoding, deep neural networks

1. Introduction

A speech signal conveys information on different semantic levels. For example, spectral features are used in the feature extraction step of conventional speech processing systems, and phonetic features are used in acoustic modelling. On the acoustic level, speech coders (for example the family of waveform-approximating coders originated in [1, 2]) operate at higher bit rates ($>$ 1000 bits-per-second). These coders are used in telephone, mobile and internet communications, so we experience them in daily life. On the parameter and higher semantic levels, so called parametric speech coders operate at lower bit rates ($<$ 1000 bits-per-second). To achieve transmission rates of the order of hundreds of bits-per-second, parametric speech coding composed of automatic phonetic speech recognition and synthesis has been proposed [3, 4, 5, 6, 7].

We are interested in very low bit rate speech coding. This necessitates our working at one of the higher semantic levels in Fig. 1. In some sense the easiest level is the word level as it is the usual level associated with speech recognition and synthesis. However, this work is rooted in the practicalities of the Swiss language scenario, which is not only multilingual but also dialectical. Rather than deal directly with the multiple phoneme sets and lexicons associated with such a scenario, we instead work at the phone level.

Aside from the practical benefit, the phone level has an appealing academic justification. The motor theory of Liberman et al. [8] suggests that speech sounds are stored in the brain as the motor commands required to reproduce such sounds. The

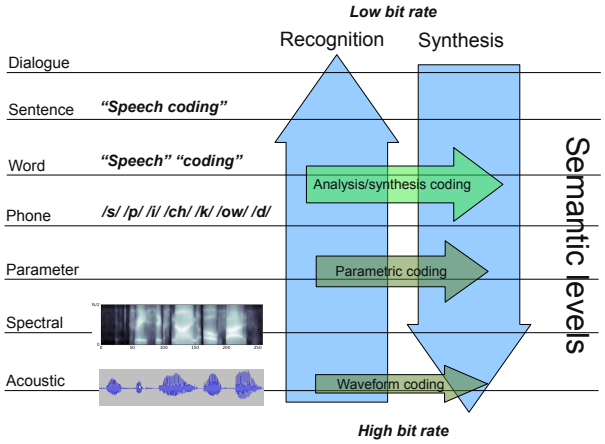

Figure 1: *Semantic levels of a speech signal.*

difference between motor commands, articulatory features and phonological features is just a (non-linear) mapping. Fig. 1, in keeping with convention in speech recognition literature, uses the word 'phone' to refer to any of these semantics; however, in this work we do distinguish them.

In previous work, we constructed a codec using the appealing symmetry of hidden Markov model (HMM)-based speech recognition and synthesis. In the last few years, however, neural networks have been used to great benefit in speech recognition; such benefit is also being shown in speech synthesis recently. This trend also leads to an appealing symmetry in the context of applications requiring both recognition and synthesis. In keeping with this symmetry, and subsuming the non-linear mapping in the motor / articulator / phonology sense, we choose to work with (deep) neural networks.

Concretely, we present the PhonVoc toolkit, a cascaded deep neural network composed of speech analyser and synthesizer that use shared phonetic and/or phonological speech representation. We provide theoretical and practical (implementation) details of the toolkit, followed by a case study investigating speech quality of phonetic and phonological vocoding. The source code is published with pre-trained English DNNs (trained on free LibriSpeech and LibriVox data), and thus the toolkit is ready for immediate use.

The structure of the paper is as follows. Section 2 describes the parametric vocoding consisting of the parametric LPC vocoder and phonetic and phonological vocoding. Section 3 provides implementation details. Section 4 describes a case study using the toolkit, and Section 5 concludes the work and points to further applications of PhonVoc: low bit rate speech coding, linguistic parsing, phonological speech synthesis and comparison of different phonological systems.

http://dx.doi.org/10.21437/Interspeech.2016-235

2. Parametric Vocoding

Moving higher on the semantic hierarchy (Fig. 1), a discretisation of the acoustic speech signal (represented by continuous speech parameters) to language symbols is performed. There are two widely divergent theories about the relation of speech to language [9]. The more conventional theory claims that the basic speech elements are speech sounds (i.e., context dependent/independent phones), whereas less conventional theory claims that the basic speech elements are articulatory gestures. Therefore, we implemented the neural network based phonetic vocoding to investigate the former claim, and the phonological vocoding to investigate the latter claim.

2.1. Parametric LPC Vocoding

Both phonetic and phonological vocoding are built on the top of a conventional parametric vocoding. The SSP vocoder[1] used by PhonVoc is a rather standard LPC codec with novel parametric mixed excitation [10]. For synthesis, harmonic and noise components are added according the harmonic to noise ratio (HNR) discerned during pitch estimation at the analysis stage. The harmonic part is shaped using a second order minimum phase linear prediction of the negative (maximum phase) part of the complex cepstrum. This effectively models the glottal formant, leading to an excitation signal with more harmonic power in the low frequencies and less at higher frequencies. The pitch estimation is continuous, rendering moot the need for a voicing detection. The modelled speech parameters are thus:

- $\boldsymbol{p}_n$: static Line Spectral Pairs (LSPs) of 24th order plus (log) gain,
- $\log(\boldsymbol{r}_n)$: a Harmonic-To-Noise (HNR) ratio,
- and $\boldsymbol{t}_n$, $\log(\boldsymbol{m}_n)$: two glottal model parameters – angle t and magnitude $\log(m)$ of a glottal pole.

2.2. Phonetic parameters

Vocoding analysis starts with speech analysis that converts speech samples into a sequence of acoustic feature observations $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \ldots, \boldsymbol{x}_N\}$ where N denotes the number of frames in the speech signal. Conventional cepstral coefficients can be used in this speech analysis step. Then, the analysis realised by DNN converts the acoustic feature observation sequence X into a sequence of vectors $Z = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n, \ldots, \boldsymbol{z}_N\}$.

Vocoding synthesis is realised as an another DNN that learns the highly-complex mapping of posteriors $\boldsymbol{z}_n$ to the speech parameters described. More specifically, it consists of two computational steps. The first step is a DNN forward pass that generates the speech parameters, and the second one is a conversion of the speech parameters into the speech samples. The generated speech parameter vectors – $\boldsymbol{p}_n$, $\boldsymbol{t}_n$, $\log(\boldsymbol{r}_n)$ and $\log(\boldsymbol{m}_n)$ for n-th frame – from the first computational step are smoothed using dynamic features and pre-computed (global) variances [11], and formant enhancement [12] is performed to compensate for over-smoothing of the formant frequencies. Parametric vocoding can be done either with synthesised or original pitch features.

The vector of the phonetic parameters $\boldsymbol{z}_n = [z_n^1, \ldots, z_n^p, \ldots, z_n^P]^\top$ for the n-th frame consists of posterior probabilities $z_n^p = p(c_p|x_n)$ of P classes (phonemes). The $.^\top$ stands for the transpose operator. The a posteriori estimates $p(c_p|x_n)$ are $0 \leq p(c_p|x_n) \leq 1, \forall p$ and $\sum_{p=1}^{P} p(c_p|x_n) = 1$. Because all the phonemes have to be recognised to access higher semantic levels, the phonetic posterior probabilities are considered as a sequential scheme.

2.3. Phonological parameters

The vector of phonological parameters $\boldsymbol{z}_n = [z_n^1, \ldots, z_n^k, \ldots, z_n^K]^\top$ consists of K phonological posterior probabilities of phonological features. The phonological posteriors are computed by a bank of parallel DNNs, each estimating the posteriors z_n^k as probabilities that the k-th phonological feature occurs (versus does not occur). The a posteriori estimates $p(c_k|x_n)$ are also $0 \leq p(c_k|x_n) \leq 1, \forall k$, but $\max \sum_{k=1}^{K} p(c_k|x_n) = K$. Only very few classes are active during a short term signal, $\sum_{k=1}^{K} p(c_k|x_n) \ll K$, that results in a sparse vector $\boldsymbol{z}_n$.

Using the phonological posterior probabilities can be considered as a parallel scheme via K different independent channels. While wrong phonetic posteriors estimation leads to a failure of the whole segment recognition, wrong phonological posteriors estimation leads to a failure only a sub-phonetic feature recognition, and this partial error does not inevitably leads to the whole segment misrecognition.

3. Implementation

Figure 2 shows the design of PhonVoc. It is split into training and the testing parts. The training consists of (i) an analysis module, speaker-independent phonetic and phonological DNNs, trained on standard ASR speech databases, and (ii) a synthesis module, a single speaker phonetic and phonological DNN, trained on audio-books available at `https://librivox.org/` as text-to-speech (TTS) databases. The trained DNNs are then used in the testing part to vocode any input speech. We implemented and tested training for:

1. the Wall Street Journal (WSJ0 and WSJ1) US English continuous speech recognition corpora [13],
2. the French radio broadcast news speech database ESTER [14], and
3. the Mandarin speech database from the EMIME project[2],

and the following three different phonological schemes:

1. the Government Phonology (GP) [15, 16],
2. the Sound Pattern of English (SPE) [17], and
3. the extended SPE system (eSPE) [18, 19],

Implementation of PhonVoc depends on the SSP vocoder and the Kaldi speech recognition toolkit [20] that we used for DNN training of both analysis and synthesis modules.

3.1. Training

Training of the analysis module consists of two steps. The first step, alignment, starts with forced aligned with cross-word triphones of an ASR database. The alignment can be obtained as a by-product of standard ASR system training.

The second step, training of the analysis DNNs, starts with mapping the phonemes from the alignment to phonological features (implemented as GP, SPE and eSPE maps). Training of the phonetic analysis DNN does not require this mapping. The

[1] `https://github.com/idiap/ssp`

[2] `http://www.emime.org/participate/emime-bilingual-database`

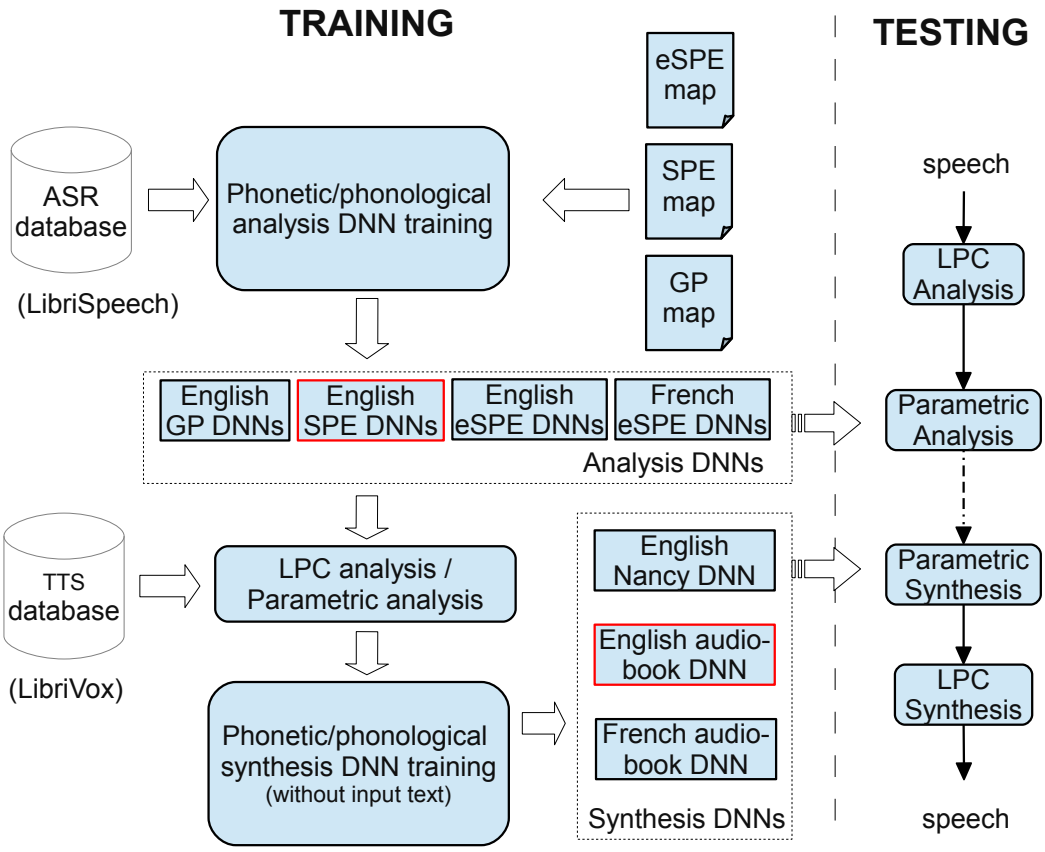

Figure 2: *Design and implementation of PhonVoc. The toolkit contains all blue models except of training databases. The trained DNNs (red boxes) are used in analysis and synthesis of testing audio. The toolkit is thus usable with the pre-trained models.*

number of outputs of the phonetic DNN is equal to the number of phonemes. The phonological analysis DNNs have two output labels, the particular phonological feature occurs for the aligned phoneme or not. The training is initialised by Deep Belief Network pre-training by contrastive divergence with 1 sampling step (CD1) [21]. The DNNs with the softmax output function are then trained using a mini-batch based stochastic gradient descent algorithm with the cross-entropy cost function.

Training of the synthesis module starts with preparing input features from the TTS database by performing the parametric (phonetic and/or phonological) analysis using the analysis DNNs. The output features – modelled speech parameters – are extracted by the LPC analysis. Cepstral mean normalisation of the output features is applied before DNN training. The DNN is also initialised by pre-training, and is trained with a linear output and the mean square error cost functions. Both analysis and synthesis DNNs are trained with the Kaldi train tool `nnet-train-frmshuff`. In the case of synthesis DNN training, the matrices of output features – training targets – are converted to posterior format with the `feat-to-post` tool.

3.2. Testing

Testing uses trained analysis and synthesis DNNs to parametrize and re-synthesize the speech signal. Users can set in the `Config.sh` file the type of parametrisation to be done. Three types are implemented:

1. `paramType=0` for phonetic speech parametrisation only,
2. `paramType=1` for phonological speech parametrisation only, and
3. `paramType=2` for joint phonetic and phonological speech parametrisation.

Optionally, the speech quality of vocoded speech signal can be evaluated by Mel Cepstral Distortion [22] (implemented in `cdist.sh`).

4. A Case Study

As a case study of using PhonVoc, we selected studying the relation of phonetic and phonological speech representations; asking, are they independent or rather complementary? For example, a recent study [23] found that they are rather complementary, e.g., using them both as a merged phonetic/phonological speech representation increased the performance of an ASR system.

4.1. Databases

We used the LibriSpeech database [24] for training the analysis part of the vocoder.

For the synthesis part, the recordings from the LibriSpeech were no usable, as they contain only 25 minutes from each speaker, in order to avoid major imbalances in per-speaker audio duration. For speech synthesis more recordings per speaker are required. Hence, we selected a full sized LibriVox audio-

book "Anna Karenina" of Leo Tolstoy[3], around 36 hours long. Recordings were organised into 238 sections, and we used sections 1–209 as a training set, 210–230 as a development set and 231–238 as a testing set. The development and testing sets were 3 hours and 1 hour long, respectively.

4.2. Training

The training of the analysis and synthesis parts followed the training described in Section 3.1. For LibriSpeech database, we used the training scripts available in Kaldi.

4.2.1. Analysis

Analysis DNNs were trained on the LibriSpeech `train-clean-100` training set, containing 100.6 hours of recordings, and cross-validation `dev-clean` set, containing 5.4 hours of recordings. The SPE phonological features were used for the phonological analysis. The 4×1024 DNNs were initialised by deep belief network pre-training, and trained by the Kaldi toolkit. Table 1 lists the features and the detection accuracy in detail. For phonetic DNN training, the accuracy of context-independent phoneme DNN was 83.7% on the training data, and 80.9% on the cross-validation data.

Table 1: *Classification accuracy (%) of the phonological analysis at frame level.*

Phonolog. features	Accuracy (%)		Phonolog. features	Accuracy (%)	
	train	cv		train	cv
vocalic	96.3	95.3	round	98.0	97.3
consonantal	96.9	93.4	tense	95.7	94.2
high	96.0	94.5	voice	95.7	94.4
back	95.0	93.3	continuant	96.1	94.9
low	97.4	96.5	nasal	98.6	98.0
anterior	95.6	94.2	strident	98.2	97.4
coronal	94.6	92.8	rising	98.1	97.2

4.2.2. Synthesis

The training and development parts of the Anna Karenina audio-book sampled at 16 kHz, framed by 25 ms windows with 10 ms frame shift, was used for training of synthesis DNN. The development set was used for cross-validation. Input features were prepared by analysis DNNs, and a temporal context of 11 successive frames resulted in an input feature vector of 165 (11×15, 14 phonological features plus silence) dimensions. Output features, LPC speech parameters, were extracted by SSP toolkit. Cepstral mean normalisation of the output features was applied before the training. Training of the 4×1024 synthesis DNN was then done as described in Section 3.1.

4.3. Results

Tab. 2 lists the quality evaluation of the different types of vocodings. LPC re-synthesis is the parametric synthesis described in Sec. 2.1. The three different types are phonetic and/or phonological vocoding done on the top of the LPC re-synthesis. We can conclude that:

1. The major degradation in vocoding quality comes from the parametric vocoding, and phonetic/phonology vocoding further degrades speech quality by about 1.4 dB.
2. We can expect that by using higher quality parametric vocoding, the phone/phonological vocoding would be improved significantly.
3. There are small but statistically significant ($p < 0.001$ of a t-test) differences in phonetic and phonological vocoding. The best quality results from the joint phonetic and phonological speech representation.

Table 2: *Objective quality evaluation of parametric and phone/phonological vocoding.*

Type	Name	MCD [dB]
–	LPC re-synthesis	4.2
0	Phone vocoding	5.6
1	Phonological vocoding	5.8
2	Phone+phonological voc.	5.5

5. Conclusion

In a broader context, techniques in analysis/synthesis vocoding can be applied to analysis-by-synthesis, a heuristic model emphasises a balance between bottom-up and top-down approaches in speech and language processing [25]. In our recent work we have already tried to apply these concepts also to the program of Laboratory Phonology [26]. Therefore, believing that such analysis/synthesis vocoding could be useful for a wider speech community, we have prepared an open-source release of the computational platform.

In this paper, we have presented the PhonVoc toolkit, a platform that implements training and testing of a cascaded deep neural network composed of speech analyser and synthesizer that use a shared phonetic and/or phonological speech representation. All the components of the PhonVoc toolkit are freely available, including pre-trained acoustic models for English language. Some examples of application of the toolkit are as follows:

- Low bit rate speech coding due to binary nature of phonological posteriors [7].
- Linguistic parsing due to structured sparsity of phonological posteriors [27].
- Phonological text-to-speech and computational phonology speech synthesis [26].
- Studying analysis and synthesis properties of the GP, SPE and eSPE phonological systems.

As a follow-up work, we recommend to a reader to train the analysis DNNs on all LibriSpeech data (about 1000 hours) and measure of their impact on quality of the parametric speech representation.

6. Acknowledgements

This work has been conducted with the support of the Swiss NSF under grant CRSII2 141903: Spoken Interaction with Interpretation in Switzerland (SIWIS), and under SP2: the SCOPES Project on Speech Prosody.

[3]https://librivox.org/anna-karenina-by-leo-tolstoy-2

7. References

[1] B. S. Atal and J. R. Remde, "A new model of LPC excitation for producing natural-sounding speech at low bit rates," in *Proc. of ICASSP*, vol. 7. IEEE, May 1982, pp. 614–617. [Online]. Available: http://dx.doi.org/10.1109/icassp.1982.1171649

[2] M. Schroeder and B. Atal, "Code-excited linear prediction(CELP): High-quality speech at very low bit rates," in *Proc. of ICASSP*, vol. 10. IEEE, Apr. 1985, pp. 937–940. [Online]. Available: http://dx.doi.org/10.1109/icassp.1985.1168147

[3] J. Picone and G. R. Doddington, "A phonetic vocoder," in *Proc. of ICASSP*. IEEE, May 1989, pp. 580–583 vol.1. [Online]. Available: http://dx.doi.org/10.1109/icassp.1989.266493

[4] K. Tokuda, T. Masuko, J. Hiroi, T. Kobayashi, and T. Kitamura, "A very low bit rate speech coder using HMM-based speech recognition/synthesis techniques," in *Proc. of ICASSP*, vol. 2. IEEE, May 1998, pp. 609–612 vol.2. [Online]. Available: http://dx.doi.org/10.1109/icassp.1998.675338

[5] K.-S. Lee and R. Cox, "A very low bit rate speech coder based on a recognition/synthesis paradigm," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 9, no. 5, pp. 482–491, Jul 2001.

[6] G. V. Baudoin and F. El Chami, "Corpus based very low bit rate speech coding," in *Proc. of ICASSP*, vol. 1. IEEE, Apr. 2003, pp. I–792–I–795 vol.1. [Online]. Available: http://dx.doi.org/10.1109/icassp.2003.1198900

[7] M. Cernak, B. Potard, and P. N. Garner, "Phonological vocoding using artificial neural networks," in *Proc. of ICASSP*. IEEE, Apr. 2015. [Online]. Available: https://publidiap.idiap.ch/index.php/publications/show/3070

[8] A. M. Liberman and I. G. Mattingley, "The motor theory of speech perception revised," *Cognition*, vol. 21, no. 1, pp. 1–36, October 1985.

[9] A. M. Liberman and D. H. Whalen, "On the relation of speech to language." *Trends in cognitive sciences*, vol. 4, no. 5, pp. 187–196, May 2000. [Online]. Available: http://view.ncbi.nlm.nih.gov/pubmed/10782105

[10] P. N. Garner, M. Cernak, and B. Potard, "A simple continuous excitation model for parametric vocoding," Idiap, Tech. Rep. Idiap-RR-03-2015, Jan. 2015. [Online]. Available: http://publications.idiap.ch/index.php/publications/show/2955

[11] K. Tokuda, T. Kobayashi, and S. Imai, "Speech parameter generation from HMM using dynamic features," in *Proc. of ICASSP*, vol. 1. IEEE, May 1995, pp. 660–663 vol.1. [Online]. Available: http://dx.doi.org/10.1109/icassp.1995.479684

[12] Z.-H. Ling, Y.-J. Wu, Y.-P. Wang, L. Qin, and R.-H. Wang, "USTC system for Blizzard Challenge 2006 - an improved HMM-based speech synthesis method," in *Proc. of Blizzard Challenge workshop*, 2006.

[13] D. B. Paul and J. M. Baker, "The design for the wall street journal-based CSR corpus," in *Proceedings of the workshop on Speech and Natural Language*, ser. HLT '91. Stroudsburg, PA, USA: Association for Computational Linguistics, 1992, pp. 357–362. [Online]. Available: http://dx.doi.org/10.3115/1075527.1075614

[14] S. Galliano, E. Geoffrois, G. Gravier, J. f. Bonastre, D. Mostefa, and K. Choukri, "Corpus description of the ester evaluation campaign for the rich transcription of french broadcast news," in *In Proceedings of the 5th international Conference on Language Resources and Evaluation (LREC 2006*, 2006, pp. 315–320.

[15] J. Harris, *English Sound Structure*, 1st ed. Wiley-Blackwell, Dec. 1994. [Online]. Available: http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0631187413

[16] J. Harris and G. Lindsey, *The elements of phonological representation*. Harlow, Essex: Longman, 1995, pp. 34–79.

[17] N. Chomsky and M. Halle, *The Sound Pattern of English*. New York, NY: Harper & Row, 1968.

[18] D. Yu, S. Siniscalchi, L. Deng, and C.-H. Lee, "Boosting attribute and phone estimation accuracies with deep neural networks for detection-based speech recognition," in *Proc. of ICASSP*. IEEE SPS, March 2012. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=157585

[19] S. M. Siniscalchi, D.-C. Lyu, T. Svendsen, and C.-H. Lee, "Experiments on Cross-Language Attribute Detection and Phone Recognition With Minimal Target-Specific Training Data," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 875–887, Mar. 2012. [Online]. Available: http://dx.doi.org/10.1109/tasl.2011.2167610

[20] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *Proc. of ASRU*. IEEE SPS, Dec. 2011, iEEE Catalog No.: CFP11SRW-USB.

[21] G. E. Hinton, S. Osindero, and Y. W. Teh, "A Fast Learning Algorithm for Deep Belief Nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006. [Online]. Available: http://dx.doi.org/10.1162/neco.2006.18.7.1527

[22] R. F. Kubichek, "Mel-cepstral distance measure for objective speech quality assessment," in *Proc. of ICASSP*, vol. 1. IEEE, May 1993, pp. 125–128 vol.1. [Online]. Available: http://dx.doi.org/10.1109/pacrim.1993.407206

[23] R. Rasipuram and Magimai, "Articulatory feature based continuous speech recognition using probabilistic lexical modeling," *Computer Speech & Language*, vol. 36, pp. 233–259, Mar. 2016. [Online]. Available: http://dx.doi.org/10.1016/j.csl.2015.04.003

[24] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *Proc. of ICASSP*. IEEE, Apr. 2015, pp. 5206–5210. [Online]. Available: http://dx.doi.org/10.1109/icassp.2015.7178964

[25] T. G. Bever and D. Poeppel, "Analysis by Synthesis: A (Re-)Emerging Program of Research for Language and Vision," *Biolinguistics*, vol. 4, no. 2-3, pp. 174–200, 2010.

[26] M. Cernak, S. Benus, and A. Lazaridis, "Speech vocoding for laboratory phonology," 2016. [Online]. Available: http://arxiv.org/abs/1601.05991

[27] M. Cernak, A. Asaei, and H. Bourlard, "On Structured Sparsity of Phonological Posteriors for Linguistic Parsing," 2016. [Online]. Available: http://arxiv.org/abs/1601.05647