Phoneme Boundary Detection using Deep Bidirectional LSTMs

Jörg Franke[1], *Markus Müller*[1], *Fatima Hamlaoui*[2], *Sebastian Stüker*[1], *Alex Waibel*[1]

[1]Karlsruhe Institute of Technology, Karlsruhe, [2]Zentrum für Allgemeine Sprachwissenschaft, Berlin
Email: joerg.franke@student.kit.edu, hamlaoui@zas.gwz-berlin.de,
{m.mueller|sebastian.stueker|alexander.waibel}@kit.edu
Web: http://isl.anthropomatik.kit.edu, http://www.zas-berlin.de/index.html

Abstract

In this paper we investigate the automatic detection of phoneme boundaries in audio recordings with the help of deep bidirectional LSTMs. This work is motivated by the needs of the project BULB which aims to support linguists in documenting unwritten languages. The automatic detection of phoneme boundaries in audio recordings of a new language is part of the technical requirements of the BULB project. For our first experiments with LSTMs for this task, we worked on TIMIT and BUCKEYE and measured the performance of our LSTMs using accuracy, precision, recall and F-measure. We then applied the trained networks crosslingually to Basaa, one of the Bantu languages addressed in BULB.

With the LSTMs trained for this paper we achieve a phoneme segmentation performance on TIMIT that, to the best of our knowledge, outperforms the systems reported in literature so far.

1 Introduction

Of the currently existing 7,000 living languages in the world [1], a large number are only spoken by a few speakers and are in danger of becoming extinct [2, 3].

Natural Language Processing (NLP) systems have been mainly created for the few languages with a large speaker base or great economic value, but are not available for the long tail of smaller, less well-resourced languages, especially not for those on the verge of extinction. Also, most of these endangered languages have not been properly documented yet.

Hereby, the number of endangered languages is so large that their comprehensive documentation by the community of documentary linguists will not be possible without support through NLP technology. Therefore it is the goal of the French-German ANR-DFG project *Breaking the Unwritten Language Barrier* (BULB) to develop tools to enable the efficient automatic processing of unwritten languages. BULB will validate its tools on three mostly unwritten African languages of the Bantu family: Basaa, Myene and Embosi [4].

One of the technologies needed to reach BULB's goals is the automatic segmentation of recordings of an unwritten language into phonemes. As we do not know the phoneme inventory of the target language, the use of multilingual phoneme recognizers for this task is not advisable, since the phoneme inventory of the multilingual recognizer might not sufficiently cover the target phoneme inventory. BULB therefore pursues a two step approach, inspired by work in speech synthesis, of first detecting phoneme boundaries, followed by classifying the detected segments into phonemes [5].

In this paper we address the first step, i.e. phoneme boundary detection, by using a deep bi-directional long short-term memory (DBLSTM) neural network. For our first experiments using BLSTMs, presented in this paper, we trained and tested first on the TIMIT corpus and then on the BUCKEYE corpus. This enables us to compare our results to other results reported in literature. Then we also applied the two English LSTMs to Basaa, one of the Bantu languages addressed, and compared the results to previous experiments on the same data using mono- and multilingual phoneme recognizers.

2 Related work

The task of either discovering the phoneme inventory of an unknown and under-resourced language or automatically segmenting recordings of it into phonemes has been studied by different research groups in the past. One common approach is to take a supervised, model based segmentation approach that has been trained either mono- or multilingually on known languages and to apply it to an unseen language. Using phoneme recognizers is a possible approach for this. E.g., [6] used HMMs for automatic segmentation and labeling of speech. Current HMM-based models with extended postprocessing reaches boundary accuracies up to 96,8 % with a 20ms tolerance on segmenting TIMIT [7]. [8] presented an HMM/SVM approach for automatic phoneme segmentation that imitates the human phoneme segmentation process.

Other approaches use features derived directly from the audio signal, to identify phone or phoneme boundaries. E.g., [9] estimated phoneme boundaries by analyzing the acoustic change of audio signals. Their method is a two step approach in which the information derived from the speech signal is enriched by additional cues. [10] describes an approach to discovering a proper set of subword-like units, which in addition to segmenting the audio, also utilizes a Dirichlet process mixture model for representing individual acoustic units. [11] performed phoneme segmentation based on acoustic clues on Mandarin. [12] examined phoneme-level segmentation of speech based on a perceptual representation —the Spectro Temporal Excitation Pattern (STEP)—and a dimensionality reduction technique—the t-Distributed Stochastic Neighbour Embedding (t-SNE). This method searches for the true phonetic boundaries in the vicinity of those produced by an HMM-based segmentation.

[13] has investigated algorithms and metrics for the task of unsupervised phoneme segmentation. Recently different kind of works have been done in the context of the Zero Speech Challenge [14], which focuses on the unsupervised discovery of subword units from raw speech. The organizers provide a unified and open suite of evaluation metrics.

With respect to cross-lingual experiments our general approach is based on [5] and [15]. In these publications,

the authors were using an English phoneme recognizer for determining phoneme boundaries as part of their work. In [16] we applied mono- and cross-lingual HMM based phoneme recognizers to unknown languages, especially to Basaa, for phoneme boundary detection. In this work we perform the phoneme boundary detection using an LSTM based approach.

Another approach for founding phoneme boundaries is unsupervised lexicon discovery. [17] reached with this method a F1-score of 77 with a 20ms tolerance on TIMIT.

3 DBSLTMs for Phoneme Boundary Detection

Recurrent neural networks (RNN) are widely used in sequence classification tasks for example in speech recognition [18], machine translation [19] or segmentation of DNA [20]. RNN's are a special kind of artificial neural network which are extended with a link in time [21]. The output of nodes from a hidden layer in the past time step are concatenated to the input of the nodes in the current time step.

Unfortunately vanilla RNNs are not good in modeling longer dependencies because of the vanishing gradient problem [22]. A common solution are Long-Short Term Memories (LSTM) first described in [23] and extended by peephole connection in [24].

They replace the ordinary activation function of an RNN through a LSTM block. While in vanilla RNNs all hidden states are fleeting, the LSTM stores hidden states steadily. This hidden cell state is controlled by three gates which adjust how much information gets preserved from the past hidden cell state (forget gate), how much new information enters (input gate) and how much information gets out (output gate), see Figure 1. The incoming signal and the outgoing signal pass a nonlinear activation function. This is usually a tanh-function as opposed to the gates which are sigmoid functions. All gates and the input activation receive the same input signals. These are the output from the layer below and the recurrent output from the same layer one time step earlier. Because the hidden state passes no activation function it is not affected by the vanishing gradient problem.

Furthermore, bidirectional Long-Short Term Memories (BLSTM) network training improves the performance further [25, 26]. In this setup layers are recurrent in both directions in time, forward and backward, and are fed forward together to the output layer. Another improvement is to stack BLSTM layers to deep bidirectional LSTMs (DBLSTM) [27]. A schematic illustration of this is pictured in Figure 2.

In this work we detect phoneme boundaries in a novel fashion with the use of DBLSTM. For this use case we found that a network architecture with two hidden layers works best. The first layer contains 200 LSTM bocks and the second 50 LSTM blocks. The whole network contains barely 150k weights and is trainable in a feasible time. Networks with more layers or more blocks per layer did not lead to better results and tended to overfit. Networks with less weights showed poorer performance. As output we use a softmax layer of size two, one for boundary and another for no boundary. Thereby we are able to calculate the network loss using a weighted cross entropy function.

Because of the rarer occurrence of boundaries the features are highly imbalanced. We correct this imbalance

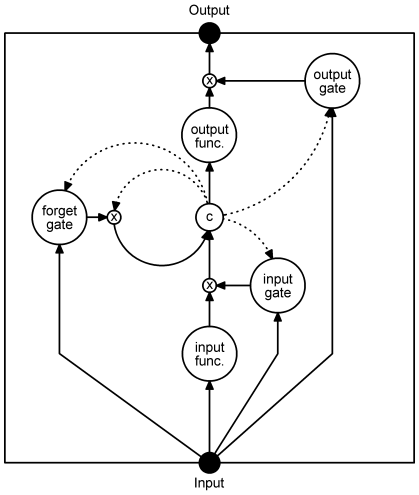

Figure 1: A LSTM block; the gates are implemented with use of a sigmoid function and the input and output function with tanh. The dotted lines are recurrent signals from the previous state. The 'c' in the middle symbolizes the internal cell state which gets information from the previous cell state and the input signal and supplies the output function.

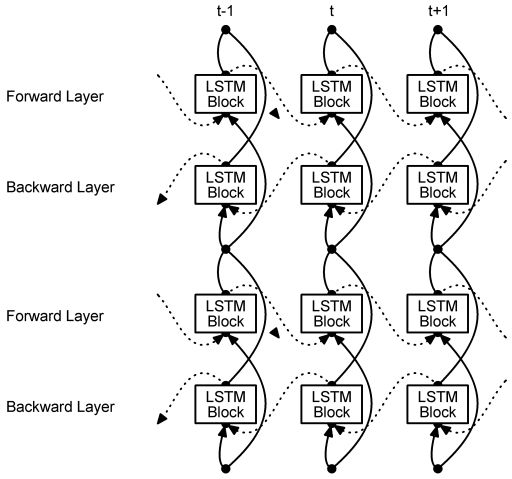

Figure 2: Deep bidirectional recurrent neural network over three time steps.

with the higher weight of the boundary feature loss, see Equation 1. The intuitive expectation that the weight should be the inverse value of the boundary / no-boundary ratio was wrong. This ratio is in the TIMIT corpus nearly 1/8 but experiments show that weights between 3 and 7 lead to better results. In this range a higher weight leads to a faster convergence. Therefore we use a weight of 7 in our experiments.

$$L(t,h) = \frac{1}{N}\sum_{n=1}^{N} -(t_n^{NB} \cdot \log(h_n^{NB}) + t_n^{B} \cdot \log(h_n^{B}) \cdot w) \tag{1}$$

For network training we use back propagation through time [28] with mini batch stochastic gradient decent. For optimization we deploy AdaDelta [29]. Its an extension of AdaGrad [30] with the same robustness but the benefit of a smoother gradient adaption and no more need for manual

tuning of the learning rate.

We implement our model with use of the Theano framework [31].

4 Data

We used three speech corpora to evaluate our networks. To train our model and to compare our work with related work we used two common phonetic speech corpora TIMIT [32] and BUCKEYE [33]. To test our model on a less-resourced language in a cross-lingual set-up we used a Basaa speech corpus form BULB.

4.1 TIMIT Corpus

TIMIT is a phonemic speech corpus which contains recordings of 630 speakers of eight major dialects of American English, each speaking 10 phonetically-rich sentences [32]. The corpus is divided into a training and test set with a ratio of roughly 60/40. We divided the data on a per-speaker basis into different sets. Although we selected the speakers randomly, we ensured that each set contains speakers of both genders as well as from each dialect. We divided the TIMIT-TRAIN set at speaker level into a training and validation set with a ratio of 90/10. The training set consists of 416 speakers, roughly 3.5h in length, divided in 4,160 unique sequences with a mean length of 3.1s and averaging 37.3 boundaries. The validation set includes 46 speakers, totaling 0.4 hours of speech, 460 sequences with a mean length of 3.1s and roughly 37.6 boundaries per sequence. The test set consists of 168 speakers with 1.4 hours of speech, 168 sequences with a mean length of 3.1s and roughly 37.2 boundaries per sequence.

4.2 BUCKEYE Corpus

Like TIMIT this corpus is an American English phonemic speech corpus with recorded interviews from 40 different speakers from Central Ohio, USA. We divided at speaker level into training, validation and test sets with an approximate ratio of 70/20/10. For simpler training we split the long records (some are over 10 minutes) into smaller sequences with a minimum length of 100 frames and cut during noises. Thus, each sequence starts and ends with maximum of 20 frames of noise and contains at minimum 100 frames of speech. Our sets do not contain non-speech-only sequences. So the BUCKEYE training set contains 28 speakers speaking 24 hours of speech divided into 25,533 sequences with a mean length of 3.4s and roughly 28.3 boundaries per sequence. The validation set includes 4 speakers speaking 2.8 hours of speech divided into 2,681 speech sequences with a mean length of 3.7s and roughly 28.0 boundaries per sequence. The test set contains 8 speakers, has a length of 6.16h, divided into 6,001 speech sequences with a mean length of 3.7s and roughly 30.4 boundaries per sequence. The reason for less boundaries per sequences at nearly the same mean sequence length compared to the TIMIT corpus is that this corpus contains more non-speech portions.

4.3 Basaa Corpus

Basaa (A43 in Guthrie's classification [34]) is one of the three Bantu languages of the BULB project. It is spoken by approximately 300,000 speakers from the Centre and Littoral regions in southern Cameroon [1]. The dialects of Basaa listed by Ethnologue [1] are the following: Bakem, Bon, Bibeng, Diboum, Log, Mpo, Mbang, Ndokama, Basso, Ndokbele, Ndokpenda, Nyamtam.

From a tonal perspective, Basaa is similar to other Bantu languages such as Kinande (JD42), Sukuma (F21) or Tiriki (E40) in that it displays a tone system that is both equipollent and privative (see [35] and references therein), as H, L and ∅ tone bearing units are underlyingly distinguished. On the surface, and as a result of a number of tonal processes, Basaa displays a 5-way opposition between H, L, LH, HL and $^{\downarrow}$H tones.

Morphologically speaking, Basaa is in several respects a typical member of the Bantu family: it displays a rich inventory of noun class markers and verb extensions. A number of phonological processes have however affected the verb stem morphology and generally made it less Bantu-like [36].

The Basaa data used in the present experiment are respoken radio broadcasts. The original audio files were obtained from the radio station CRTV-Centre and feature a male native speaker of Basaa. His speech was phonetically transcribed by a linguist and later carefully re-spoken by a female native speaker of Basaa. The re-speaking sessions were conducted in a quiet environment, using the Voice Memo application of a smartphone.

The test set used in our experiments contains 0.4 hours of speech divided into 345 sequences with a mean length of 4.1s and roughly 31.8 boundaries per sequence.

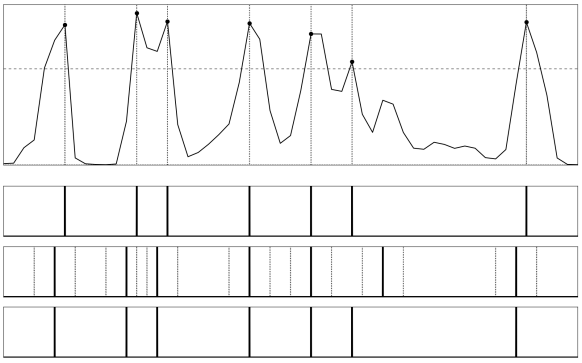

Figure 3: The upper plot is an example network output of the boundary feature with a threshold of 0.6 and peak detection. Below are the detected boundaries and in the third plot shows the correct target boundaries with a 20ms tolerance. The bottom plot displays the corrected boundary detection which is used for measurement.

5 Experiments

We conducted two sets of experiments. First we trained our model on TIMIT and evaluated on the TIMIT test set and then the Basaa corpus. Second we trained our model on BUCKEYE and evaluated on the BUCKEYE test set and again on Basaa.

5.1 Preprocessing

We oriented the preprocessing on related work and used the same procedure for both corpora. We extract 12 Mel-Frequency Cepstrum Coefficients and augment them with the log-energy and an approximation of the first derivative.

TIMIT TRAINING	Tolerance	Accuracy	Precision	Recall	F1-Score
Valid TIMIT	10ms	96.5 ± 0.03	87.2 ± 0.51	83.5 ± 0.47	85.1 ± 0.10
	20ms	97.6 ± 0.04	91.5 ± 0.42	88.1 ± 0.64	89.7 ± 0.20
Test TIMIT	10ms	96.3 ± 0.03	86.0 ± 0.44	83.2 ± 0.49	84.6 ± 0.14
	20ms	97.5 ± 0.04	91.1 ± 0.39	88.1 ± 0.60	89.6 ± 0.18
Test Basaa	10ms	88.9 ± 0.06	30.4 ± 0.22	33.4 ± 0.79	31.9 ± 0.43
	20ms	91.3 ± 0.06	44.2 ± 0.29	48.6 ± 0.87	46.3 ± 0.39

Table 1: Results of training on TIMIT, postprocessing with threshold of 0.6, optimized on the validation set for F1-score. For better reading all values are multiplied by 100.

BUCKEYE TRAINING	Tolerance	Accuracy	Precision	Recall	F1-Score
Valid BUCKEYE	10ms	97.2 ± 0.03	83.6 ± 0.76	81.9 ± 0.61	82.5 ± 0.07
	20ms	98.0 ± 0.02	88.7 ± 0.76	86.4 ± 0.68	87.5 ± 0.07
Test BUCKEYE	10ms	96.5 ± 0.04	81.9 ± 0.89	77.7 ± 0.70	79.7 ± 0.11
	20ms	97.5 ± 0.03	87.8 ± 0.88	83.3 ± 0.82	85.5 ± 0.12
Test Basaa	10ms	89.7 ± 0.04	33.0 ± 0.23	31.9 ± 0.62	32.4 ± 0.40
	20ms	91.9 ± 0.05	47.7 ± 0.33	46.2 ± 0.66	46.9 ± 0.38

Table 2: Results of training on BUCKEYE corpus, measurement with a threshold of 0.6 in respect to the best F1-score on the validation set. For a better reading all values multiplied by 100.

This gives a vector of 26 features per time step. We choose a time step size of 10ms and a window length of 25ms. Experiments have shown that smaller step sizes as well as more filter coefficients lead to better cross entropy losses but not to better F1-scores. For the targets of the features we unify all non-speech labels to noise. For training we split sequences when they are longer than 5 seconds. In doing so we cut only in noises and let each sequence start and end with noise. If no noise occurs within 5 seconds then we accept longer sequences. To improve the training time we sort all the training sequences by length and construct minibatches with sequences of nearly the same length. We use minibatches with a size of 10. To train minibatches with samples of different length we use masks.

5.2 Postprocessing Network Output

Based on [37] we used a peak picking method to derive the actually predicted phoneme boundaries from the frame wise peaks in a posteriori probability for frame boundaries from the DBSLTM. Additionally we used a variable threshold to adjust when we accept a peak as boundary. In a second step we adapted the output with respect to the commonly used tolerance of 10ms and 20ms of the true boundaries [9], as shown in Figure 3.

As quality measures we use accuracy, precision, recall and F1-score. The often used correct detection rate is the same as recall.

5.3 Experiments on TIMIT

We trained six models with different random initializations about 20 epochs on the TIMIT training set. After training we used these six models to determine the postprocessing threshold on the validation as to optimize F1-score. Table 1 shows the mean and standard deviation of the quality measures of the six models for both, a tolerance of 10ms and 20ms. With a higher threshold precision goes up to 0.98 with an F1-score of about 0.70 and a 20ms tolerance. In contrast, lowering the threshold brings recall up to 0.93 with an F1-score of about 0.84 and a 20ms tolerance.

5.4 Experiments on BUCKEYE

Like in the TIMIT experiment we trained six models with about 20 epochs and determined the postprocessing threshold on the validation set of BUCKEYE with respect to the maximum F1-score. Table 2 shows mean and standard error of our quality measures for the six models with both, a tolerance of 10ms and 20ms. By lifting the threshold a precisions of up to 0.95 with an F1-score of 0.80 and a 20ms tolerance could be obtained. Otherwise by lowering the threshold a recall of up to 0.93 was possible at an F1-score of 0.80 with a 20ms tolerance.

6 Conclusion

In this paper we presented the use of deep bidirectional LSTMs for detecting phoneme boundaries. We trained separate systems on TIMIT and BUCKEYE and tested them on their respective tasks, but also applied them cross-lingually to Basaa, one of the languages of interest in the BULB project.

Our experiments outperform our previous method that is based on phoneme recognizers and shows promise in cross-lingual application. The results on TIMIT that we present in this paper are, to the best of our knowledge, the best phoneme segmentation results reported in literature so far.

7 Acknowledgement

This work was realized in the framework of the ANR-DFG project BULB (STU 593/2-1 and ANR-14-CE35-002) and also supported by the French Investissements d'Avenir - Labex EFL program (ANR-10-LABX-0083)

References

[1] R. G. G. Jr. and B. F. G. (Eds.), *Ethnologue: Languages of the World*. Dallas, Texas, USA: SIL International, 2005.

[2] D. Nettle and S. Romaine, *Vanishing Voices*. New York, NY, USA: Oxford University Press Inc., 2000.

[3] D. Crystal, *Language Death*. Cambridge, UK: Cambridge University Press, 2000.

[4] S. Stüker, G. Adda, M. Adda-Decker, O. Ambouroue, L. Besacier, D. Blachon, H. Bonneau-Maynard, P. Godard, F. Hamlaoui, D. Idiatov, G.-N. Kouarata, L. Lamel, E.-M. Makasso, M. Müller, A. Rialland, M. V. de Velde, F. Yvon, and S. Zerbian, "Innovative technologies for under-resourced language documentation: the bulb project," in *2nd Workshop Collaboration and Computing for Under-Resourced Languages (CCURL 2016)*, 2016.

[5] P. K. Muthukumar and A. W. Black, "Automatic discovery of a phonetic inventory for unwritten languages for statistical speech synthesis," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 2613–2617, IEEE, 2014.

[6] F. Brugnara, D. Falavigna, and M. Omologo, "Automatic segmentation and labeling of speech based on hidden markov models," *Speech Communication*, vol. 12, no. 4, pp. 357–370, 1993.

[7] A. Stolcke, N. Ryant, V. Mitra, J. Yuan, W. Wang, and M. Liberman, "Highly accurate phonetic segmentation using boundary correction models and system fusion," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 5552–5556, IEEE, 2014.

[8] J. wei Kuo, H. yi Lo, and H. min Wang, "Improved hmm/svm methods for automatic phoneme segmentation," in *in Proc. Interspeech*, pp. 2057–2060, 2007.

[9] O. Scharenborg, V. Wan, and M. Ernestus, "Unsupervised speech segmentation: An analysis of the hypothesized phone boundaries," *The Journal of the Acoustical Society of America*, vol. 127, no. 2, pp. 1084–1095, 2010.

[10] C. ying Lee and J. Glass, "A nonparametric bayesian approach to acoustic model discovery," in *Association for Computational Linguistics, 50th Annual Meeting of the*, pp. 40–49, Association for Computational Linguistics, 2012.

[11] J. Yuan, N. Ryant, and M. Liberman, "Automatic phonetic segmentation in Mandarin Chinese: boundary models, glottal features and tone," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 2539–2543, IEEE, 2014.

[12] A. Stan, C. Valentini-Botinhao, M. Giurgiu, and S. King, "Phonetic segmentation of speech using step and t-sne," in *Speech Technology and Human-Computer Dialogue (SpeD), 2015 International Conference on*, pp. 1–6, IEEE, 2015.

[13] Y. Qiao, N. Shimomura, and N. Minematsu, "Unsupervised optimal phoneme segmentation: objectives, algorithm and comparisons," in *Acoustics, Speech and Signal Processing (ICASSP), 2008 IEEE International Conference on*, pp. 3989–3992, IEEE, 2008.

[14] M. Versteegh, R. Thiolliere, T. Schatz, X. N. Cao, X. Anguera, A. Jansen, and E. Dupoux, "The zero resource speech challenge 2015," in *Proc. of INTERSPEECH*, 2015.

[15] P. Baljekar, S. Sitaram, P. K. Muthukumar, and A. W. Black, "Using articulatory features and inferred phonological segments in zero resource speech processing," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[16] M. Vetter, M. Müller, F. Hamlaoui, G. Neubig, S. Nakamura, S. Stüker, and A. Waibel, "Unsupervised phoneme segmentation of previously unseen languages," in *submitted to INTERSPEECH*, 2016.

[17] C.-y. Lee, T. J. O'Donnell, and J. Glass, "Unsupervised lexicon discovery from acoustic input," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 389–403, 2015.

[18] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 1764–1772, 2014.

[19] S. Liu, N. Yang, M. Li, and M. Zhou, "A recursive recurrent neural network for statistical machine translation," in *Proceedings of ACL*, pp. 1491–1500, 2014.

[20] W.-C. Cheng, J.-C. Huang, and C.-Y. Liou, "Segmentation of DNA using simple recurrent neural network," *Knowledge-Based Systems*, vol. 26, pp. 271–280, Feb. 2012.

[21] J. L. Elman, "Finding structure in time," *Cognitive science*, vol. 14, no. 2, pp. 179–211, 1990.

[22] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," 2001.

[23] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[24] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber, "Learning precise timing with lstm recurrent networks," *The Journal of Machine Learning Research*, vol. 3, pp. 115–143, 2003.

[25] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *Signal Processing, IEEE Transactions on*, vol. 45, no. 11, pp. 2673–2681, 1997.

[26] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm networks," in *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, vol. 4, pp. 2047–2052, IEEE, 2005.

[27] A. Graves, N. Jaitly, and A.-r. Mohamed, "Hybrid speech recognition with deep bidirectional lstm," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pp. 273–278, IEEE, 2013.

[28] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[29] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[30] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *The Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, 2011.

[31] Theano Development Team, "Theano: A Python framework for fast computation of mathematical expressions," *arXiv e-prints*, vol. abs/1605.02688, May 2016.

[32] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "Timit acoustic-phonetic continuous speech corpus ldc93s1. web download," 1993.

[33] M. A. Pitt, L. Dilley, K. Johnson, S. Kiesling, W. Raymond, E. Hume, and Fosler-Lussier, "Buckeye Corpus of Conversational Speech (2nd release)," 2007.

[34] M. Guthrie, *The classification of the Bantu languages*. Oxford University Press for the International African Institute, 1948.

[35] L. Hyman, "Markedness, faithfulness, and two-height tone systems," in *Proceedings from the Montreal-Ottawa-Toronto (MOT) Phonology Workshop 2011: Phonology in the 21st Century: In Honour of Glyne Piggot. McGill Working Papers in Linguistics*, vol. 22, pp. 1–13, 2012.

[36] L. Hyman, *The Bantu languages*, ch. Bàsàa (A43), pp. 257–282. Routledge, 2003.

[37] S. Dusan and L. R. Rabiner, "On the relation between maximum spectral transition positions and phone boundaries," in *INTERSPEECH*, 2006.