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

Formant Estimation and Tracking using Deep Learning

Yehoshua Dissen and Joseph Keshet

Department of Computer Science Bar-Ilan University, Ramat-Gan, Israel
disseny1@cs.biu.ac.il, joseph.keshet@biu.ac.il

Abstract

Formant frequency estimation and tracking are among the most fundamental problems in speech processing. In the former task the input is a stationary speech segment such as the middle part of a vowel and the goal is to estimate the formant frequencies, whereas in the latter task the input is a series of speech frames and the goal is to track the trajectory of the formant frequencies throughout the signal. Traditionally, formant estimation and tracking is done using ad-hoc signal processing methods. In this paper we propose using machine learning techniques trained on an annotated corpus of read speech for these tasks. Our feature set is composed of LPC-based cepstral coefficients with a range of model orders and pitch-synchronous cepstral coefficients. Two deep network architectures are used as learning algorithms: a deep feed-forward network for the estimation task and a recurrent neural network for the tracking task. The performance of our methods compares favorably with mainstream LPC-based implementations and state-of-the-art tracking algorithms.

Index Terms: formant estimation, formant tracking, deep neural networks, recurrent neural networks

1. Introduction

Formants are considered to be resonances of the vocal tract during speech production. There are 3 to 5 formants, each at a different frequency, roughly one in each 1 kHz band. They play a key role in the perception of speech and they are useful in the coding, synthesis and enhancement of speech, as they can express important aspects of the signal using a very limited set of parameters [1]. An accurate estimate of these frequencies is also desired in many phonological experiments in the fields of laboratory phonology, sociolinguistics, and bilingualism (see examples [2, 3]).

The problem of formant estimation has received considerable attention in speech recognition research as formant frequencies are known to be important in determining the phonetic content as well as articulatory information about the speech signal. They can either be used as additional acoustic features or can be utilized as hidden dynamic variables as part of the speech recognition model [4].

The formant frequencies approximately correspond to the peaks of the spectrum of the vocal tract. These peaks cannot be easily extracted from the spectrum, since the spectrum is also tainted with pitch harmonics. Most commonly, the spectral envelope is estimated using a time-invariant all-pole linear system, and the formants are estimated by finding the peaks of the spectral envelope [1, 5]. While this method is very simple and efficient it lacks the accuracy required by some systems.

Most algorithms for tracking are based on traditional peak picking from Linear Predictive Coding (LPC) spectral analysis or cross-channel correlation methods coupled with continuity constraints [1, 5, 6]. More elaborate methods used dynamic programming and HMMs to force continuity [7, 8, 9]. Other algorithms for formant tracking are based on Kalman filtering [10, 11] and extended in [12]. Other authors [13, 14] have used autocorrelation sequence for representing speech in a noisy speech recognition system and [15, 16, 17] use LPC of the zero phase version of the signal and the peaks of its group delay function.

Recently a publicly available corpus of manually-annotated formant frequencies of read speech was released [18]. The corpus is based on the TIMIT corpus, and includes around 30 min of transcribed read speech. The release of this database enables researchers to develop and evaluate new algorithms for formant estimation.

In this paper we present a method called *DeepFormants* for estimating and tracking formant frequencies using deep networks trained on the aforementioned annotated corpus. In the task of formant estimation the input is a stationary speech segment (such as the middle of a vowel) and the goal is to estimate the first 3 formants. In the task of formant tracking the input is a sequence of speech frames and the goal is to predict the sequence of the first 3 formants corresponding to the input sequence. In both tasks the signal is represented using two sets of acoustic features. The first set is composed of LPC cepstral coefficients extracted from a range of LPC model orders, while the second set is composed of cepstral coefficients derived from quasi-pitch-synchronous spectrum.

We use a feed-forward network architecture for the task of estimation and a recurrent neural network architecture for the task of tracking. RNN is a type of neural network that is a powerful sequence learner. In particular, the Long Short-Term Memory (LSTM) architecture has shown to provide excellent modeling of sequential data such as speech [19].

The paper is organized as follows. The next section describes the two sets of features. Section 3 presents the deep network architectures for each tasks. Section 4 evaluates the proposed method by comparing it to state of the art LPC implementations, namely WaveSurfer [20] and Praat [21], and to two state of the art tracking algorithms: MSR [10] and KARMA [12]. We conclude the paper in Section 5.

2. Acoustic Features

A key assumption is that in the task of estimation the whole segment is considered stationary, which mainly holds for monophthongs (pure vowels). In the task of tracking, the speech signal is considered stationary over roughly a couple dozen milliseconds. In the former case the features are extracted from the whole segment, while in the latter case the input signal is divided into frames, and the acoustic features are extracted from each frame. The spacing between frames is 10 msec, and frames are overlapping with analysis windows of 30 msec. As with

http://dx.doi.org/10.21437/Interspeech.2016-490

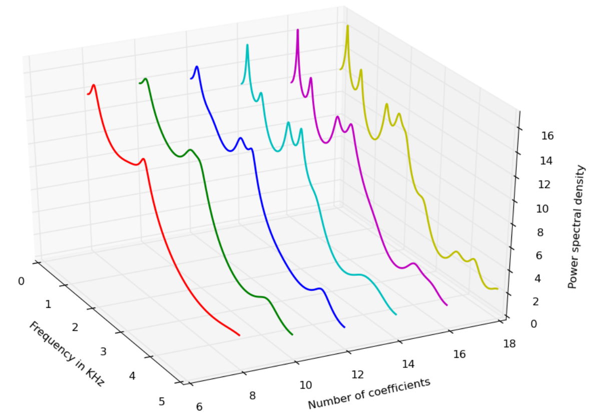

Figure 1: *LPC spectrum of the vowel /uw/ produced for 262 msec for values of p 8,10,12,14,16, and 18.*

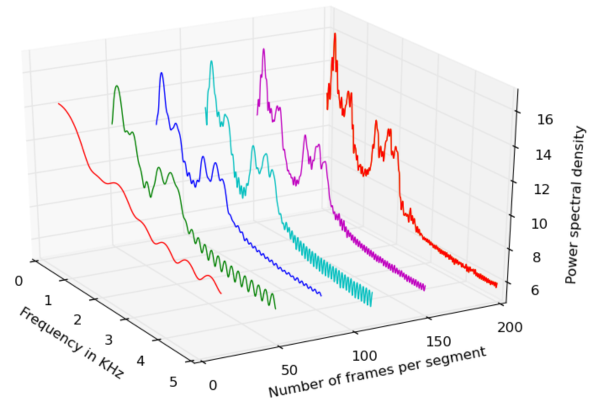

Figure 2: *Quasi pitch-synchronous sepctra of the vowel /uw/ produced for 262 msec with different values of pitch. The true value of the pitch was 123.4 frames.*

all processing with this type, we apply a pre-emphasis filter, $H(z) = 1 - 0.97\,z^{-1}$, to the input speech signal, and a Hamming window to each frame.

At this phase, two sets of spectral features are extracted. The goal of each of the sets is to parametrize the envelop of the short-time Fourier transform (STFT). The first set is based on Linear Predictive Coding (LPC) analysis, while the second is based on the pitch-synchronous spectra. We now describe in detail and motivate each set of features.

2.1. LPC-based features

LPC model determines the coefficients of a forward linear predictor by minimizing the prediction error in the least squares sense.

Consider a frame of speech of length N denoted by $\bar{s} = (s_1, \ldots, s_N)$, where s_n the n-th sample. The LPC model assumes that the speech signal can be approximated as a linear combination of the past p samples:

$$\hat{s}_n = \sum_{k=1}^{p} a_k s_{n-k} \tag{1}$$

where $\boldsymbol{a} = (a_1, \ldots, a_p)$ is a vector of p coefficients. The values of the coefficients $\boldsymbol{a}$ are estimated so as to minimize the mean square error between the signal $\bar{s}$ and the predicted signal $\hat{s} = (\hat{s}_1, \ldots, \hat{s}_N)$,

$$\boldsymbol{a} = \arg\min_{\boldsymbol{a}} \frac{1}{N} \sum_{n=1}^{N} (s_n - \hat{s}_n)^2. \tag{2}$$

Plugging Eq. (1) into Eq. (2), this optimization problem can be solved by a linear equation system.

The spectrum of the LPC model can be interpreted as the envelop of the speech spectrum. The model order p determines how smooth the spectral envelop will be. Low values of p represent the coarse properties of the spectrum, and as p increases, more of the detailed properties are preserved. Beyond some value of p, the details of the spectrum do not reflect only the spectral resonances of the sound, but also the pitch and some noise. Figure 1 illustrates this concept, by showing the spectrum of the all-pole filter with values of p ranging from 8 to 18. A disadvantage of this method is that if p is not well chosen (i.e., to match the number of resonance present in the speech), then the resulted LPC spectrum is not as accurate as desired [22].

Our first set of acoustic features are based on the LPC model. Instead of using a single value of the number of LPC coefficients, we used a range of values between 8 and 17. This way the classifier can combine or filter out information from different model resolutions. More specifically, in our setting after applying pre-emphasize and windowing, the LPC coefficients for each value of p were extracted using the autocorrelation method, where the Levinson-Durbin recursion was used for the autocorrelation matrix inversion, and the FFT for the autocorrelation computation.

The final processing stage is to convert the LPC spectra to cepstral coefficients. This is done efficiently by the method proposed in [23]. Denoted by $\boldsymbol{c} = (c_1, \ldots, c_n)$ is the vector of the cepstral coefficients where $n > p$:

$$c_m = \begin{cases} a_m + \displaystyle\sum_{k=1}^{m-1} \left(1 - \frac{k}{m}\right) a_k c_{m-k} & 1 \leq m \leq p \\ \displaystyle\sum_{k=1}^{p} \left(1 - \frac{k}{m}\right) a_k c_{m-k} & p < m \leq n \end{cases}.$$

We tried different values for n and found that $n = 30$ gave reasonable results.

2.2. Pitch-synchronous spectrum-based features

The spectrum of a periodic speech signal is known to exhibit a impulse train structure located at multiples of the pitch period. A major concern when using the spectrum directly for locating the formants is that the resonance peaks might fall between two pitch lines, and then they are not "visible". The LPC model estimates the spectrum envelop to overcome this problem. Another method to estimate the spectrum while eliminating the pitch impulse train is using the *pitch synchronous spectrum* [24]. According to this method the DFT is taken over frames the size of the instantaneous pitch.

One of the main problem of this method is the need of a very accurate pitch estimator. Another issue is how to implement the method in the case of formant estimation, when the input is a speech segment that represents a single vowel, which typically spans a few pitch periods, and the pitch in not fixed along the segment. We found out that using a pitch period which is close enough to its exact value is good enough in our application. This can be observed in Figure 2, where the quasi pitch-synchronous FFT for different values of pitch periods are depicted. It can be seen that except for extreme cases, the peaks of the spectrums are well-smoothed and clearly defined.

In out implementation we extract quasi-pitch synchronous spectrum similar to [24]. For the task of formant estimation we use the median pitch computed in frames of 10 msec along the input segment, and use the average spectra.

At the final stage, the resulting quasi pitch-synchronous spectrum is converted to cepstral coefficients by applying log compression and then Discrete Cosine transform (DCT). We use the first 100 DCT coefficients as our second set of features.

3. Deep Learning Architectures

In this section we describe the two network architectures that are used for formant estimation and formant tracking. In the former the input is a speech segment representing a single vowel and the goal is to extract the first three formants, and in the latter the input is a series of speech frames and the goal is to extract the corresponding series of values of the first three formants.

3.1. Network architecture for estimation

The method chosen to classify the data was a standard feed forward neural network. The the input of the network is a vector of 400 features (30 DCT features for each of the 10 LPC model sizes plus 100 features of the quasi pitch-synchronous spectrum), and the output is a vector of the three annotated formants.

The network has three hidden layers with 1024, 512 and 256 neurons respectively and all of them are fully connected. The activations for said layers are sigmoid functions. The network was trained using adagrad [25] to minimize the mean absolute error or the absolute difference between the predicted and true formant frequencies with weights randomly initialized. The training of the networks weights was done as regression rather than classification. The network predicts all 3 formants simultaneously to exploit interformant constraints.

3.2. Network architecture for tracking

For tracking we use a Recurrent Neural Network (RNN) consisting of an input layer with 400 features as in the estimation task. In addition to these features extracted from the current segment of speech on account of the fact that this is an RNN the predictions and features of the previous speech segment (i.e. temporal context) are taken into account when predicting the current segments formants. Next are two Long Short Term Memory (LSTM) [26] layers with 512 and 256 neurons respectively, a time distributed fully connected layer with 256 neurons and an output layer consisting of the 3 formant frequencies. As in the estimation network the activations were all sigmoid, the optimizer was adagrad and the function to minimize was mean absolute error.

4. Evaluation

For the training and validating our model we used the Vocal Tract Resonance (VTR) corpus [18]. This corpus is composed of 538 utterances selected as a representative subset of the well-known and widely-used TIMIT corpus. These were split into 346 utterances for the training set and 192 utterances for the test set. These utterances were manually annotated for the first 3 formants and their bandwidths for every 10 msec frame. The fourth formant was annotated by the automatic tracking algorithm described in [10], and it is not used here for evaluation.

4.1. Estimation

We will begin by presenting the results for our estimation algorithm. The estimation algorithm applies only to vowels (monophthongs and diphthongs). We used the whole vowel segments of the VTR corpus. Their corresponding annotation were taken to be the average formants along the segments.

Table 1 shows the influence of our different feature sets. The loss is the mean absolute difference between predicted values and their manually annotated counterparts measured in Hz. It can be seen that using different LPC model orders improves the performance on F_2 and F_3, and the performance on F_1 improves with the quasi-pitch-synchronous feature set.

Table 1: *The influence of different feature sets on the estimation of formant frequencies of whole vowels using deep learning.*

Feature set	F_1	F_2	F_3
LPC, $p = 12$	59	123	179
LPC, $p = \{8 - 17\}$	60	86	110
quasi-pitch-sync	51	115	164
LPC, $p = \{8 - 17\}$ + quasi-pitch-sync	**48**	**83**	**109**

As a baseline we compared our results to those of Praat, a popular tool in phonetic research [21]. Formants were extracted from Praat using Burg's method with a maximum formant value of 5.5 kHz, a window length of 30 msec and a pre-emphasis from 50 Hz. The results of our system and of Praat's on the test set are shown in Table 2, where the loss is the mean absolute difference in Hz. As seen in the table, we have achieved better results across the board over Praat when comparing our respective estimations to the manually annotated reference.

Table 2: *Estimation of formant frequencies of whole vowels using deep learning and Praat.*

	Method	F_1	F_2	F_3
Mean	DeepFormants	**48**	**83**	**109**
	Praat	75	115	151
Median	DeepFormants	**38**	**62**	**75**
	Praat	48	78	88
Max	DeepFormants	**528**	**716**	**1509**
	Praat	1611	1711	1633

In addition, the observed mean differences between our automated measurements and the manually annotated measurements are comparable in size to the generally-acknowledged uncertainty in formant frequency estimation demonstrated on our dataset by the degree of inconsistency between different labelers in Table 3 and to the perceptual difference limens found in [27]. Such that it is doubtful that higher accuracy can be achieved with automated tools seeing as manual annotation cannot.

Analysis of the predictions with the largest inaccuracies show that they broadly fall into 3 categories, either they are annotation errors and the system indeed did classify them accurately, the vowel segment was very short (less than 35 ms) and

Table 3: *Tracking errors of on broad phone classes measured by mean absolute difference in Hz.*

	inter-labler			WaveSurfer			Praat			MSR [10]			DeepFormants		
	F_1	F_2	F_3	F_1	F_2	F_3	F_1	F_2	F_3	F_1	F_2	F_3	F_1	F_2	F_3
vowels	55	69	84	70	94	154	130	230	267	64	105	125	**54**	**81**	**112**
semivowels	68	80	103	89	126	222	136	295	334	83	122	**154**	**67**	**114**	168
nasal	75	112	106	96	229	239	219	409	381	67	**120**	**112**	**66**	175	151
fricatives	91	113	125	209	263	439	564	593	700	**129**	**108**	**131**	131	135	159
affricates	89	118	135	292	407	390	730	515	583	**141**	**129**	**149**	164	162	189
stops	91	110	116	168	210	286	258	270	351	**130**	**113**	**119**	131	135	168

Table 4: *Same as for Table 3 except for the focus on temporal regions of CV transitions and VC transitions.*

	WaveSurfer			Praat			MSR [10]			DeepFormants		
	F_1	F_2	F_3	F_1	F_2	F_3	F_1	F_2	F_3	F_1	F_2	F_3
CV transitions	156	192	273	169	225	261	**106**	**101**	**119**	110	142	165
VC transitions	59	88	157	344	355	495	**48**	92	120	53	**80**	**111**

ambiguous spectrograms where both the manual annotation and the predicted value can be correct.

4.2. Tracking

We now present the results for our tracking model. We evaluated the model on whole spoken utterances of VTR. We compared our results to Praat, to the results obtained in [18] from WaveSurfer and from the MSR tracking algorithm. Table 3 shows the accuracy in mean absolute difference in Hz for each broad phonetic class. The inter-labeler variation is also presented in this table for reference (from [18]).

Our method outperforms Praat and WaveSurfer in every category, and compared to MSR our model shows higher precision with vowels and semivowels while MSR reports higher precision with nasals, fricatives, affricates and stops. It's worth mentioning though that the phone class where formants are most indicative of speech phenomena is vowels. The higher precision reported by MSR in consonant phone classes is most likely due to the fact that the database abtained its initial trajectory labels from MSR and was then manualy corrected [18] so in phonemes without clear formants (i.e. consonants) there is a natural bias towards the trajectories labled by MSR.

We also examined the errors of the algorithms when limiting the error-counting regions to only the consonant-to-vowel (CV) and vowel-to-consonant (VC) transitions. The transition regions are fixed to be 6 frames, with 3 frames to the left and 3 frames to the right of CV or VC boundaries defined in the TIMIT database. The detailed results are listed in Table 4.

Results from other works on the VTR dataset include [12] and compared to his results seen in Table 5 our precision is on par for the first formant but greatly improved for the second and third formants. Error is measured in root mean squared error (RMSE).

Table 5: *Formant tracking performance of KARMA, and deep learning in terms of root-mean-square error (RMSE) per formant. RMSE is only computed over speech-labeled frames.*

Method	F_1	F_2	F_3	Overall
KARMA [12]	**114**	226	320	220
DeepFormants	118	**169**	**204**	**163**

5. Conclusions

Accurate models for formant tracking and estimation were presented with the former surpassing existing automated systems accuracy and the latter within the margins of human inconsistencies. Deep learning has proved to be a viable option for automated formant estimation tasks and if more annotated data is introduced, we project higher accuracy models can be trained as analysis of the phonemes with the least accuracy on average seems to show that they were the ones that were represented the least in the database.

In this paper we have demonstrated automated formant tracking and estimation tools that are ready to be added to the methods that sociolinguists use to analyze acoustic data. The tools will be publicly available at `https://github.com/MLSpeech/DeepFormants`.

In future work we will consider the formant bandwidths estimation. Moreover, we would like to evaluate our method on noisy environments, as well as reproducing phonological experiments such as [28].

6. References

[1] D. O'Shaughnessy, "Formant estimation and tracking," in *Springer handbook of speech processing*, J. Benesty, M. M. Sondhi, and Y. Huang, Eds. Springer, 2007.

[2] B. Munson and N. P. Solomon, "The effect of phonological neighborhood density on vowel articulation," *Journal of speech, language, and hearing research*, vol. 47, no. 5, pp. 1048–1058, 2004.

[3] C. G. Clopper and T. N. Tamati, "Effects of local lexical competition and regional dialect on vowel production," *The Journal of the Acoustical Society of America*, vol. 136, no. 1, pp. 1–4, 2014.

[4] L. Deng and J. Ma, "Spontaneous speech recognition using a statistical coarticulatory model for the vocal-tract-resonance dynamics," *The Journal of the Acoustical Society of America*, vol. 108, no. 6, pp. 3036–3048, 2000.

[5] S. S. McCandless, "An algorithm for automatic formant extraction using linear prediction spectra," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 22, no. 2, pp. 135–141, 1974.

[6] L. Deng and C. D. Geisler, "A composite auditory model for processing speech sounds," *The Journal of the Acoustical Society of America*, vol. 82, no. 6, pp. 2001–2012, 1987.

[7] G. E. Kopec, "Formant tracking using hidden markov models and vector quantization," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 34, no. 4, pp. 709–729, 1986.

[8] M. Lee, J. Van Santen, B. Möbius, and J. Olive, "Formant tracking using context-dependent phonemic information," *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 5, pp. 741–750, 2005.

[9] D. T. Toledano, J. G. Villardebó, and L. H. Gómez, "Initialization, training, and context-dependency in hmm-based formant tracking," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 2, pp. 511–523, 2006.

[10] L. Deng, L. J. Lee, H. Attias, and A. Acero, "A structured speech model with continuous hidden dynamics and prediction-residual training for tracking vocal tract resonances," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, vol. 1. IEEE, 2004, pp. I–557.

[11] ——, "Adaptive kalman filtering and smoothing for tracking vocal tract resonances using a continuous-valued hidden dynamic model," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 1, pp. 13–23, 2007.

[12] D. D. Mehta, D. Rudoy, and P. J. Wolfe, "Kalman-based autoregressive moving average modeling and inference for formant and antiformant trackinga)," *The Journal of the Acoustical Society of America*, vol. 132, no. 3, pp. 1732–1746, 2012.

[13] J. Hernando, C. Nadeu, and J. Mariño, "Speech recognition in a noisy car environment based on lp of the one-sided autocorrelation sequence and robust similarity measuring techniques," *Speech Communication*, vol. 21, no. 1, pp. 17–31, 1997.

[14] J. A. Cadzow, "Spectral estimation: An overdetermined rational model equation approach," *Proceedings of the IEEE*, vol. 70, no. 9, pp. 907–939, 1982.

[15] D. Ribas Gonzalez, E. Lleida Solano, C. de Lara, and R. Jose, "Zero phase speech representation for robust formant tracking," in *Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European*. IEEE, 2014, pp. 1462–1466.

[16] M. Anand Joseph, S. Guruprasad, and B. Yegnanarayana, "Extracting formants from short segments of speech using group delay functions," in *Proceeding of Interspeech*, 2006.

[17] H. A. Murthy and B. Yegnanarayana, "Group delay functions and its applications in speech technology," *Sadhana*, vol. 36, no. 5, pp. 745–782, 2011.

[18] L. Deng, X. Cui, R. Pruvenok, Y. Chen, S. Momen, and A. Alwan, "A database of vocal tract resonance trajectories for research in speech processing," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 1. IEEE, 2006, pp. I–I.

[19] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6645–6649.

[20] K. Sjölander and J. Beskow, "Wavesurfer-an open source speech tool." in *Interspeech*, 2000, pp. 464–467.

[21] P. Boersma and D. Weenink, "Praat, a system for doing phonetics by computer," *Glot international*, vol. 5, no. 9/10, pp. 341–345, 2002.

[22] G. E. Birch, P. Lawrence, J. C. Lind, and R. D. Hare, "Application of prewhitening to ar spectral estimation of EEG," *Biomedical Engineering, IEEE Transactions on*, vol. 35, no. 8, pp. 640–645, 1988.

[23] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *the Journal of the Acoustical Society of America*, vol. 55, no. 6, pp. 1304–1312, 1974.

[24] Y. Medan and E. Yair, "Pitch synchronous spectral analysis scheme for voiced speech," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 37, no. 9, pp. 1321–1328, 1989.

[25] 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.

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

[27] P. Mermelstein, "Difference limens for formant frequencies of steady-state and consonant-bound vowels," *The Journal of the Acoustical Society of America*, vol. 63, no. 2, pp. 572–580, 1978.

[28] S. Reddy and J. N. Stanford, "Toward completely automated vowel extraction: Introducing darla," *Linguistics Vanguard*, 2015.