INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Predicting Perceptual Centers Located at Vowel Onset in German Speech Using Long Short-Term Memory Networks

Felicia Schulz[12], *Mirella De Sisto*[1], *M. Paula Roncaglia-Denissen*[1], *Peter Hendrix*[1]

[1]Tilburg University, The Netherlands
[2]Lund University, Sweden

fe5502sc-s@student.lu.se, m.desisto@tilburguniversity.edu,
m.p.roncaglia@tilburguniversity.edu, p.h.g.hendrix@tilburguniversity.edu

Abstract

Perceptual centers (p-centers) can be defined as the perceived centers of a syllable. Previous research regarding the location of p-centers in speech relied on experimental methods, and among the suggested acoustic features contributing to the location of p-centers in Germanic languages is the transition of the consonant to the vowel onset. The current study investigates the prediction of the location of p-centers in German, by means of machine learning. Machine learning is a promising tool to capture possible non-linear relationships that may occur among the acoustic features used in the complexity that is the human perception. Therefore, an LSTM neural network approach was used for the identification of p-centers in a set of spoken German sentences, with input data features being Mel Frequency Cepstral Coefficients (MFCC), amplitude envelope and root mean squared energy. The model was able to achieve a balanced accuracy of 84% with MFCCs being the best predictor of p-center location.

Index Terms: perceptual centers, Long Short-Term Memory, Mel Frequency Cepstral Coefficients, deep learning

1. Introduction

Perceptual centers (p-centers) can be defined as the perceived moment a sound occurs, and this seems to be mostly constant among humans [1]. However, there is no agreement regarding the precise acoustic features involved in determining the exact location of p-centers in a sound (see section 2). There seem to be universal points of attraction of p-centers in each syllable, but research has not yet succeeded in finding significant, distinctive predictors.

This research aims to predict the p-centers in German speech using a deep learning neural network. More specifically, a supervised recurrent neural network, the Long-Short Term Memory (LSTM) network, will be applied. As input for the LSTM, several different acoustic and phonetic features will be extracted from the data, such as Mel Frequency Cepstral Coefficients, amplitude envelope and root mean squared energy.

This methodology can be very beneficial for future p-center research: the automation of p-center locations could bring new insight to the field, even though the relevant acoustic features contributing to the p-center location might not be known. Furthermore, it has the potential of being applied on different data and of providing an easier and less time-consuming method for p-center prediction. In addition to that, the model could bring new insights to the question of which features actually affect p-center location. With a more in-depth understanding of p-centers, one can understand better how human perception of sounds occurs, which brings valuable insights on language acquisition and understanding in general [2, 3].

2. Related Work

2.1. Existing research on p-centers

P-centers were first recognised as a phenomenon worth investigating by Morton and associates (1976) when coming across difficulties in the attempt to produce stimuli at regular intervals [1]. The question of what exactly in the syllable determines the regularity arose, which led them to study p-centers further. In this research, they also found that p-centers occur independently of their surrounding syllables [1, 4, 5].

In order to measure p-centers, several different experimental methods have been applied, and the most frequently used methods are the rhythm adjustment method, the phase-correction response (PCR) and the tap-asynchrony method [5]. In the rhythm adjustment method, two sounds are played in a cyclic pattern, one is a base sound and the other is a test sound. The base-base interval is constant but the base-test interval can vary. Participants adjust the timing of the test sound until the "point of subjective isochrony" [5, p. 1616]. The p-centers can then be determined from the distances between the consecutive sounds. In PCR, a sequence of sounds is played with regular intervals and occasional event onset shifts or phase shifts between the sounds. While listening, the participants are instructed to tap along to the rhythm of the sound, and if there is a shift in the timing of one of the sounds, they adjust their tap in order to adapt to the perceived new rhythm. This adjustment of the timing of the participants' taps then identifies the p-center of the next sound. Finally, in the tap-asynchrony method, the participants are asked to tap along to each sound that is played.

Although research is generally in agreement about the context independence of p-centers, there is still no consensus on the exact determiners of p-centers within each syllable. De Jong (1994) investigated a number of acoustic and articulatory kinematic features in two separate experiments, but no significant correlates were found [6]. Moreover, amplitude envelope, i.e., the distribution of energy in the sound, has also been proposed as a candidate for predicting p-center location [7]. Some research suggests that the p-center is likely to occur close to the energy peak in a syllable [7], for a different view, please see [8, 9, 4]. P-center location is also suggested to be influenced by acoustic makeup and the durational features of syllables [4], such as vowel onset [9]. Using a speech-metronome synchronisation approach, Barbosa and associates [10] found that in Germanic languages, the vowel onset correlates with the p-center if the onset has a large amount of energy. Overall, the general consensus is that the p-center location is not predicted by any single acoustic event but rather must occur due to a number of features, the precise interactions of which have not been discovered yet [1, 4, 6, 9].

10.21437/Interspeech.2023-2154

2.2. Feature extraction in modern Audio Signal Processing

Feature extraction is a crucial part of audio signal analysis, especially when the research includes a machine learning component. One example of a type of feature in the time domain used in speech analysis is root mean square energy (RMSE) [11]. It represents the loudness of the sound by calculating the square root of the mean squared amplitude of the signal [12].

Mel Frequency Cepstral Coefficients (MFCCs) are a feature which is frequently used for vowel detection in audio signal processing [13]. They also hold information about time and frequency of the audio signal, simulating sound perception in a similar fashion to the human ear [14]. The resulting coefficients have shown to possess great explanatory and predictive power regarding music and speech processing and analysis [11].

Finally, another feature that is often used in modern audio analysis is energy amplitude envelope [15], and a correlation between amplitude envelope and p-center location has been found in previous research [16].

2.3. Machine Leaning models for Audio Signal Processing

For time-series data, Recurrent Neural Networks (RNNs) are a type of neural network that is often used for time-series data as for each time step they have a separate internal state [17]. The Long Short-Term Memory (LSTM) network is a type of RNN which overcomes the vanishing gradient problem that classical RNNs have [17], which means that over time, information from previous time steps "vanishes" and the impact of their data becomes insignificant [18].

3. Methodology

3.1. Data set description

The original data set consists of 88 sound files which are a subset of the stimulus material created and used by Roncaglia-Denissen and associates [19, 20] for the purpose of studying the role of rhythmic regularity during syntactic ambiguity processing in German. The sentences presented a rhythmically regular stress pattern (i.e., with a constant interstress interval of three syllables). [For additional information about the dataset, please see [19].]

The sentences were spoken by a German female professional speaker and recorded. Since previous research on p-centers made use of rhythmically regular stimulus material (i.e., mostly word lists), this dataset allows for the investigation of p-centers in a more natural, and ecologically valid speech context.

3.2. Data pre-processing

Each sound file was analysed and p-centers corresponding to each syllable were extracted and manually labelled. This was done using the software Praat [21]. The vowel onset was identified in Praat by investigating markers such as the waveforms, the intensity, the pitch, formants of the sound and patterns on the spectrogram.

The data was zero-padded and signal duration was increased up to a common length so that no valuable data is lost (please see Table 1 for the comparison between original and edited data). For this procedure, the Python library Pydub was used [22].

For processing the audio signal, first, the sampling rate of 22050 samples per second was chosen as it preserves enough data to represent the original, continuous signal sufficiently without being too computationally expensive [23]. For framing and windowing of the now discrete signal, a frame length of 2048 and a hop size of 512 was selected, as these are common parameters to choose in audio signal processing [24, 25, 26].

Table 1: *Min, max and mean durations and frame numbers of files before and after editing.*

	Original files	**Edited files**
Min	3.1576 secs, 132 fr	3.9098 secs, 169 fr
Max	3.9093 secs, 165 fr	3.9099 secs, 169 fr
Mean	3.5551 secs, 150 fr	3.9098 secs, 169 fr

In the final step in the data pre-processing, the target data had to be converted into a format that would fit the LSTM network. In the processed data, the p-centers were denoted by occurrence per frame using binary numbers. This means that for each of the 169 frames, either a 0 (no p-center) or a 1 (p-center) was assigned. There was an average of 17 frames containing p-centers for each signal, so an average of 10% 1s and 90% 0s. If a p-center occurred at a point in time where two frames were overlapping on the signal, the p-center was allocated to the frame for which it was closer to the center point.

3.3. Audio feature extraction

For the extraction of the audio features, the Python package Librosa for audio and music analysis was used [27]. In addition to that, for visualisation of the data, the Python package Matplotlib was used [28].

In this project, MFCCs were used as features of the model, since they are often used for vowel detection [13], of which the onset has been linked to p-center location [10]. Here, 20 MFCCs were extracted for each frame, because a high amount of information content and possible predictive power was desired while still maintaining low computational complexity.

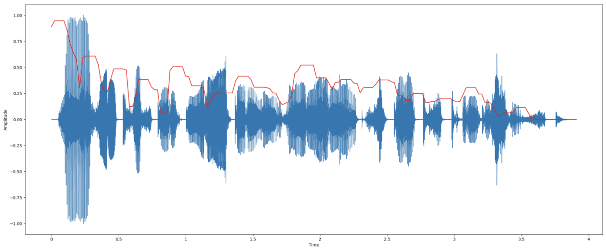

Figure 1: *Plot showing amplitude envelope for one sample signal and its waveform over time.*

The second feature that was extracted from the audio data was the amplitude envelope, since previous studies suggest its relevance in p-center location [16]. Figure 1 shows the amplitude envelope mapped onto a waveform of a sample audio signal. The sentence in the signal is: "*Maren trifft den Diener den Lorena mal gestört hat im Geschäft*". The same sentence is represented in Figure 2 which shows the RMSE, the third feature that was extracted. This feature was used in order to investigate the potential effect of energy in the signal on the location of p-centers as this was also found to be a potential predictor in Germanic languages [10].

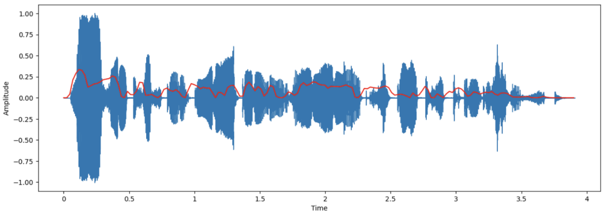

Figure 2: *Plot showing RMSE for one sample signal and its waveform over time.*

3.4. LSTM network implementation

For this research, an LSTM network was chosen because of its recursive nature and its ability to overcome the vanishing gradient problem. The LSTM itself was implemented using the Python library Keras [29]. Due to the zero-padding that was required for increasing the duration of some of the signals, first, a masking layer was added to the model to ensure that the output does not become distorted because of the padding. Initially, three LSTM layers were implemented.

For this model, the hyperbolic tangent and the Rectified Linear Unit (ReLU) activation functions were chosen as they are both suitable for binary classification and showed the best overall model performance. The loss function in this LSTM was sparse categorical crossentropy loss. It was chosen because it is appropriate for categorical classification, which is carried out by this model with this data. Other loss functions were tested but resulted in a decrease in model performance. For this specific loss function, the activation function of the last, dense layer, must be the Softmax function. Other activation functions were also experimented with, but resulted in a decrease in performance of around 10 percentage points. The Adam optimizer and the evaluation metrics accuracy were selected for this network.

The input data was split up into 15% testing data, 15% validation data and 70% training data. The LSTM was run using the same settings separately for each input data. In addition to that, it was compared to a baseline of randomly generated numbers between 0 and 1.

In order to avoid overfitting, the complexity of the model needed to be reduced to only two LSTM layers and sample weights which assign different weights to each class were set.

3.5. Model evaluation

The performance of the LSTM was evaluated using balanced accuracy on the test set, due to the unbalanced nature of the data. Each variation of the model with different input data was run ten times and the means and standard deviations of the balanced accuracy outputs were compared. This and the following statistical analysis were carried out using the programming language R on the software RStudio [30, 31]. A one-directional paired student's t-test was conducted on each of the model outputs with different feature inputs in order to see whether they perform significantly better than the baseline with an α value of .05. In addition to that, confusion matrices were computed for each model variant, and sensitivity and specificity were calculated.

4. Results

The baseline input data with random values performed as expected with a mean balanced accuracy of 51% in the ten runs of the LSTM (*SD* = .01) as seen in Table 2. The MFCC data performed best in predicting the target data as it showed a mean balanced accuracy of 83% (*SD* = .01) and a maximum balanced accuracy of 84%. The RMSE input had a mean balanced accuracy of 61% (*SD* = .05) and the mean balanced accuracy of the amplitude envelope data was 69% (*SD* = .05).

Table 2: *Balanced accuracy means and standard deviations (SD) for each different input feature and the baseline.*

	Baseline	**MFCC**	**RMSE**	**AE**
Mean	.51	.83	.61	.69
SD	.01	.01	.05	.05

The results of the one-sided, paired student's t-test showed that the balanced accuracy of the MFCC model is significantly greater than the baseline, because the *t* value is large ($t = 46.74$) and there is a 95% confidence interval of a mean difference of 31% (see Table 3).

Furthermore, the balanced accuracy of the amplitude envelope model is significantly greater than that of the baseline *t* α ($t = 13.24$, $p < .001$). The 95% confidence interval of the mean difference is 16%. The null hypothesis can also be rejected, thus, the amplitude envelope model's balanced accuracy is significantly greater than the baseline. Similarly, the balance accuracy for the RMSE model was also significantly greater than the baseline with a 95% confidence interval of a mean difference of 7% ($t = 5.87$, $p < .001$).

Table 3: *Results of the one-sided, paired student's t-tests.*

	MFCC	**AE**	**RMSE**
t	46.74	13.24	5.87
95% Confidence Interval	.31	.16	.07
p-value	<.001	<.001	<.001

Confusion matrices were computed in which a positive value is equal to 1 and a negative value is 0. Moreover, the specificity and sensitivity of the models were calculated.

Table 4: *Confusion matrix for the LSTM with the MFCC input data.*

		Target	
		Positive	Negative
Model	Positive	207	374
	Negative	31	1754
		Sensitivity	**Specificity**
		.87	.82

The confusion matrix of the MFCC model in Table 4 shows that the number of true positives is 207, and there were 31 false positive cases. The sensitivity is therefore high (87%). There are many true negative cases and not many false negatives, which is why specificity is 82%.

In the confusion matrix of the model which used amplitude envelopes as input in Table 5, the sensitivity is 63%, because 88 out of 238 actual positive values were incorrectly labelled as negative. Moreover, 439 cases that the model predicted to be positive were in fact negative, so the specificity of this model is 79%.

Table 5: *Confusion matrix for the LSTM with the AE input data.*

		Target	
		Positive	Negative
Model	Positive	150	439
	Negative	88	1689
		Sensitivity	**Specificity**
		.63	.79

Furthermore, Table 6 shows that in the LSTM with the RMSE data as input, sensitivity is 76%. The specificity is only 45%, because there were 1177 cases in which the true value was negative but the LSTM predicted them to be positive.

Table 6: *Confusion matrix for the LSTM with the RMSE input data.*

		Target	
		Positive	Negative
Model	Positive	180	1177
	Negative	58	951
		Sensitivity	**Specificity**
		.76	.45

In order to show the baseline performance of the LSTM, a confusion matrix for the random inputs has also been calculated and can be seen in Table 7. This matrix shows the low performance of the baseline.

Table 7: *Confusion matrix for the LSTM with the baseline.*

		Target	
		Positive	Negative
Model	Positive	44	281
	Negative	194	1847
		Sensitivity	**Specificity**
		.18	.86

5. Discussion

The goal of this study was to predict p-center location estimated at vowel onset in German speech using an LSTM. The input features which were extracted from the audio data were MFCCs, amplitude envelopes, and the RMSE.

The LSTM was able to predict the p-centers in the unseen data with a balanced accuracy of up to 84%. The performance of the LSTM was dependent on the input data, and balanced accuracy scores varied with different predictors. The MFCCs had the highest predictive power, scoring the maximum mean balanced accuracy of 83%. Amplitude envelope was able to predict p-center location with the second-highest mean balanced accuracy of 69%, and the RMSE input led to a 61% mean balanced accuracy of the model. The LSTM showed a low standard deviation for each of the different features (.01 < *SD* <.06), which means that its performance was constant across all runs. The model always performed significantly better than the baseline.

With the highest performing predictor, the MFCCs, which also had the highest sensitivity and specificity scores, the model was able to predict both the negative and positive classes similarly well. With the amplitude envelopes as input, the sensitivity of the LSTM was 63% and the specificity was 79%. Therefore, the model had similar predictive power as the MFCC model over the negative classes but performed slightly worse in predicting the positive class. This may be because the positive class is much less frequent due to the data being unbalanced. Finally, the confusion matrix for the LSTM which used the RMSE input features presented a sensitivity of 76% and a specificity of 45%. Hence, the model was better at predicting the positive class than the negative class. This behaviour is surprising due to the unbalanced nature of the data, which would usually cause the model to be more likely to predict the more frequent class in case of ambiguity.

The findings on MFCCs are in line with the existing literature, as MFCCs are some of the most used features in speech processing research. Amplitude envelope and RMSEs both extract features related to the energy of the signal at a certain point in time, but amplitude envelope performed better than RMSEs. This may be because the amplitude envelope reflects the change in amplitude over a certain amount of time, rather than describing an aspect of the sound at one specific point in time like RMSE does. This could be an interesting causality to further research. Overall, as suggested in previous research [10], the energy of the sound does seem to be a significant predictor for p-centers in German speech.

Compared to the existing literature, this study adds a new method, the use of LSTM networks, to the research of locating p-centers. In general, the promising results suggest that using machine learning in research regarding similar phenomena is a valuable approach with the potential of being employed increasingly in future research.

6. Conclusion

This research aimed to find how well a supervised learning algorithm can predict p-centers located at vowel onset from acoustic and phonetic features in German speech.

The supervised learning algorithm in the LSTM neural network was able to predict the p-center location with a balanced accuracy of up to 84% from the MFCC features. The model performed significantly better than the baseline. In addition to that, the amplitude envelope and the RMSE features which were also extracted from the German audio signals were significant p-center predictors as well. The amplitude envelope features showed the second highest predictive power with a balanced accuracy of 69%, followed by the RMSEs with 61%.

In future research, it would be interesting to test this model on a different data set that has less rhythmic regularity and is closer to free speech. Furthermore, the methodology in this project could be combined with experimental methods in classical p-center identification models to gain more knowledge about this phenomenon of human perception. With the use of these new, efficient methods, researchers may consider revisiting the debate about p-center attractors and potentially open up the topic to many new discussions and conclusions.

7. References

[1] J. Morton, S. Marcus, and C. Frankish, “Perceptual centers (p-centers).” *Psychological review*, vol. 83, no. 5, p. 405, 1976.

[2] I. Chow, M. Belyk, V. Tran, and S. Brown, “Syllable synchronization and the p-center in cantonese,” *Journal of Phonetics*, vol. 49, pp. 55–66, 2015.

[3] H. Chung, B. Munson, and J. Edwards, “Cross-linguistic perceptual categorization of the three corner vowels: Effects of listener language and talker age,” *Language and speech*, vol. 64, no. 3, pp. 558–575, 2021.

[4] S. M. Marcus, “Acoustic determinants of perceptual center (p-center) location,” *Perception & psychophysics*, vol. 30, no. 3, pp. 247–256, 1981.

[5] R. C. Villing, B. H. Repp, T. E. Ward, and J. M. Timoney, “Measuring perceptual centers using the phase correction response,” *Attention, Perception, & Psychophysics*, vol. 73, no. 5, pp. 1614–1629, 2011.

[6] K. J. De Jong, “The correlation of p-center adjustments with articulatory and acoustic events,” *Perception & Psychophysics*, vol. 56, no. 4, pp. 447–460, 1994.

[7] P. Howell, “An acoustic determinant of perceived and produced anisochrony,” in *Proceedings of the 10th international Congress of Phonetic Sciences*. Foris Dordrecht, Holland, 1984, pp. 429–433.

[8] B. Tuller and C. A. Fowler, “The contribution of amplitude to the perception of isochrony,” *Haskins Laboratories Status Report on Speech Research*, pp. 245–250, 1981.

[9] A. M. Cooper, D. Whalen, and C. A. Fowler, “P-centers are unaffected by phonetic categorization,” *Perception & Psychophysics*, vol. 39, no. 3, pp. 187–196, 1986.

[10] P. A. Barbosa, P. Arantes, A. R. Meireles, and J. M. Vieira, “Abstractness in speech-metronome synchronisation: P-centres as cyclic attractors.” in *Interspeech*. Citeseer, 2005, pp. 1441–1444.

[11] G. Sharma, K. Umapathy, and S. Krishnan, “Trends in audio signal feature extraction methods,” *Applied Acoustics*, vol. 158, p. 107020, 2020.

[12] M. B. Er, “A novel approach for classification of speech emotions based on deep and acoustic features,” *IEEE Access*, vol. 8, pp. 221 640–221 653, 2020.

[13] A. Kumar, S. Shahnawazuddin, and G. Pradhan, “Improvements in the detection of vowel onset and offset points in a speech sequence,” *Circuits, systems, and signal processing*, vol. 36, no. 6, pp. 2315–2340, 2017.

[14] L. Muda, M. Begam, and I. Elamvazuthi, “Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques,” *arXiv preprint arXiv:1003.4083*, 2010.

[15] L. He and V. Dellwo, “Amplitude envelope kinematics of speech: Parameter extraction and applications,” *The Journal of the Acoustical Society of America*, vol. 141, no. 5, pp. 3582–3582, 2017.

[16] P. Howell, “Prediction of p-center location from the distribution of energy in the amplitude envelope: I,” *Perception & Psychophysics*, vol. 43, no. 1, pp. 90–93, 1988.

[17] R. C. Staudemeyer and E. R. Morris, “Understanding lstm–a tutorial into long short-term memory recurrent neural networks,” *arXiv preprint arXiv:1909.09586*, 2019.

[18] J. J. Hopfield, “Neurons with graded response have collective computational properties like those of two-state neurons,” *Proceedings of the national academy of sciences*, vol. 81, no. 10, pp. 3088–3092, 1984.

[19] M. P. Roncaglia-Denissen, M. Schmidt-Kassow, and S. A. Kotz, “Speech rhythm facilitates syntactic ambiguity resolution: Erp evidence,” *PloS one*, vol. 8, no. 2, p. e56000, 2013.

[20] M. P. Roncaglia-Denissen, M. Schmidt-Kassow, A. Heine, and S. A. Kotz, “On the impact of l2 speech rhythm on syntactic ambiguity resolution,” *Second Language Research*, vol. 31, no. 2, pp. 157–178, 2015.

[21] P. Boersma and D. Weenink, “Praat,” 2022. [Online]. Available: http://www.praat.org/

[22] J. Robert, M. Webbie *et al.*, “Pydub,” 2018. [Online]. Available: http://pydub.com/

[23] M. Müller, *Fundamentals of Music Processing: Using Python and Jupyter Notebooks*. Springer Nature, 2021.

[24] H. Wang, D. Chong, and Y. Zou, “Acoustic scene classification with multiple decision schemes,” *Tech. Rep., DCASE2020 Challenge*, 2020.

[25] S. Panwar, A. Das, M. Roopaei, and P. Rad, “A deep learning approach for mapping music genres,” in *2017 12th System of Systems Engineering Conference (SoSE)*. IEEE, 2017, pp. 1–5.

[26] K. W. Cheuk, K. Agres, and D. Herremans, “The impact of audio input representations on neural network based music transcription,” in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–6.

[27] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, “librosa: Audio and music signal analysis in python,” in *Proceedings of the 14th python in science conference*, vol. 8, 2015.

[28] J. D. Hunter, “Matplotlib: A 2d graphics environment,” *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[29] F. Chollet *et al.*, “Keras,” https://keras.io, 2015.

[30] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2020. [Online]. Available: https://www.R-project.org/

[31] RStudio Team, *RStudio: Integrated Development Environment for R*, RStudio, Inc., Boston, MA, 2019. [Online]. Available: http://www.rstudio.com/