A Deep Learning Approach to Automatic Prosodic Segmentation in Untranscribed Discourse

Nathan Roll
University of California, Santa Barbara
nroll@ucsb.edu

Dr. Calbert Graham
University of Cambridge
crg29@cam.ac.uk

***Abstract*—Information in spoken language is determined by both the semantic meaning of words and the manner in which they are spoken, known as prosody. Prosodic units, also known as intonation units (IUs), are short phrases, each bound by a single tune, that play a role in both cognitive processing and speech synthesis. Despite applications in a variety of domains ranging from discourse analysis to synthetic speech generation, automatic IU identification remains unsolved. We propose a completely automated approach to the identification of prosodic boundaries utilizing a heuristic potential boundary reduction algorithm, input space compression, and an ensemble of CNN and RNN models, resulting in an F-score of 0.74.**

I. Introduction

Humans are exceptional at recognizing patterns, even when they may be elusive computationally. Identification of intonation units (IUs) is no exception, as dozens of empirically abstract cues collectively comprise our perception of prosodic segments. Some of the most important are tune gestalt (unity of pitch/intonation contour), reset (return to a baseline pitch), silent pauses, anacrusis (rapid, unstressed syllables), and lag (lengthening of speech) [1]. We hypothesize that sufficiently complex computational models may be able to recognize some of these cues without the need for an accompanying lexical transcription.

Some attempts have already been made towards automatic segmentation and boundary detection in speech using a variety of methods. Semi-supervised Learning for Automatic Prosodic Event Detection Using Co-training Algorithm [2] achieves an F-score of 0.77 on a supervised labeled prosodic break detection task. In Automatic Detection of Prosodic Boundaries in Spontaneous Speech [3], a variety of heuristic and theoretical models were able to achieve an F-score of 0.65 without the need for supervised training. Variances in segment definitions, input features (syntactic and/or acoustic), and corpus content (number of speakers, scripted or unscripted, etc.), however, make comparisons of results difficult.

We define boundaries as timestamps which occur at the beginning or end of an IU. A dynamic number of speakers means that multiple initial or end boundaries may occur sequentially in simultaneous speech examples. Successful identification of these boundaries lays the foundations for an automatic segmentation system while also helping us better understand prosody. Applications also exist in other domains, notably fluency assessment and synthetic speech generation [4] [5].

Convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are utilized for their distinct specializations. CNNs are designed for image recognition tasks– passing convolutional filters over groups of pixels or other filters repeatedly to determine which ones hold predictive power [6]. In contrast, RNNs excel at modeling temporal sequences. They attempt to generalize sequential relationships which may extend across an entire sample. Such an architecture may be able to identify segment-spanning features like tune, while underperforming CNNs in short-term signatures like pitch reset or vocal quality changes.

The framework proposed has the following advantages:

- Inferences can be made from audio files alone; transcript-reliant forced alignment is replaced by a heuristic algorithm.
- Predictions are near-instantaneous.
- Multiple relationships are considered in parallel.

For training and testing purposes, we use the Santa Barbara Corpus of Spoken American English (SBC), a collection of 60 discourse events spanning about 20 hours [7]. Included are stereo audio files (22,050 Hz) and manually generated IU-segmented transcripts with timestamps accurate to the nearest tenth of a second. A variety of demographic backgrounds (age, race, sex) are represented, with sensitive information removed in a manner which preserves pitch.

II. Methods

A. Preprocessing

The audio files in the Santa Barbara Corpus were reduced to a single channel through left-right averaging and downsampled to 8kHz. Given that boundaries must occur directly preceding or following a word, as IUs do not encompass peripheral pauses, mid-word or mid-pause samples did not need to be considered in the data. Also, a lack of precision (0.05s at 8kHz yields a potential error of 400 samples) must be accounted for. Forced alignment is an option, but this requires prior transcription and is relatively expensive computationally. [8]. Instead, we implement a maximum-recall algorithm to identify all potential IU boundaries [1]:

1) Take the absolute value of given waveform

[1]All code, including examples and descriptions, can be found in the project github: https://github.com/Nathan-Roll1/ProsodPy

2) Compute the moving average over n samples (n=0.025s=200)
3) Subset every nth value
4) Index local minima
5) Potential boundaries exist where minima separate speech above given threshold

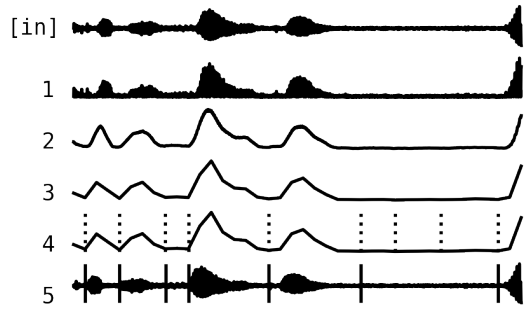

Fig. 1. Boundary Heuristic Algorithm on SBC 32: 1.0s to 2.0s

Mel-frequency cepstral coefficients (MFCCs) are spectral transformations which maintain some pitch, intentity, and temporal information [9]. For each manually transcribed boundary and an equal number of randomly selected non-boundaries (identified by the heuristic), the first fifteen MFCCs were extracted with a hop length of 16 samples (0.002s) and a window size of 624 samples (0.078s) with a buffer of 512 frames (0.624s) on either side. To avoid precision loss due to padding, a minimal number of boundaries occurring at the beginning or end of the audio files were discarded. Each MFCC was normalized on its own axis by subtracting the mean and dividing by the standard deviation. A balanced dataset of 128,508 MFCC matrices and associated labels (true/false boundaries) form our train (80%), test (10%), and validation (10%) sets.

The prior (preboundary), surrounding (boundary), and posterior (postboundary) featuresets within each MFCC are considered separately to explicitly match input sequences to boundaries. An additional 50% subset of each is also considered (with the prefix "h_").

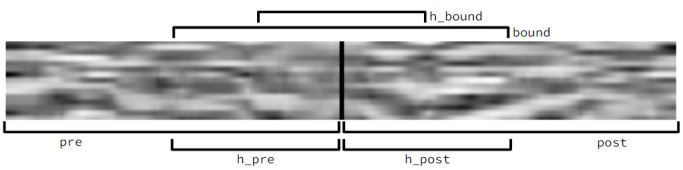

Fig. 2. MFCC-Extracted Featuresets

Postboundary matrices are inverted on the temporal axis so that the sequence terminus corresponds to the boundary in question.

B. Neural Networks

The CNNs and RNNs are both trained using MSE loss, minimizing the squared distance between the floating point sigmoid outputs and the binary boundary classes. Due to inconsistencies during training, the CNN models utilize the stochastic gradient descent (SGD) optimizer while the RNNs use Adam [10]. The models which perform best on the validation data during training are checkpointed with a max epoch of 45.

CNN architecture:

Conv2d: 128 3x3 Filters; ReLU
Conv2d: 64 3x3 Filters; ReLU
2x2 Max Pooling
Flattening Layer
Dense: 8 Nodes; Tanh
20% Dropout
Dense: 1 Node; Sigmoid

RNN architecture:

LSTM: 128 Units; Sigmoid
LSTM: 64 Units; Sigmoid
Dense: 32 Nodes; Tanh
20% Dropout
Dense: 1 Node; Sigmoid

C. Ensemble

CNN and RNN outputs on the training data are regressed on boundary labels using an ordinary least squares (OLS) regression (fig. 3). This determines which models are significant, how they should be weighted, and acts as its own predictive model. [2]

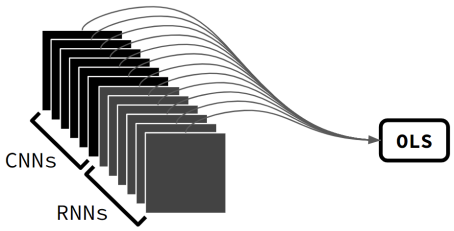

Fig. 3. Ensemble Diagram

III. Results

A. Metrics

We find that MFCC-based neural networks provide significant predictive power in the determination of IU boundaries. Furthermore, linear combinations of their outputs outperform any one individually.

The regression problem is transformed into one of classification by rounding the continuous outputs off at certain thresholds (t). Intermediate steps were evaluated using MSE and r^2, while the binary categorical results of rounding are measured in terms of F-score and accuracy. F-score is particularly informative in the realm of cross-model evaluation and precision/recall balancing.

[2]Models are trained on a remote Python 3.7.13 environment using tensorflow [11] and scikit-learn [12]

TABLE I: Performance Metrics (t=0.5)

	F1	Precision	Recall	Accuracy ⇓
OLS	**0.719**	**0.728**	**0.709**	**0.724**
pre_rnn	0.683	0.707	0.660	0.695
h_pre_rnn	0.679	0.693	0.666	0.687
h_bound_rnn	0.680	0.660	0.701	0.671
bound_cnn	0.648	0.676	0.622	0.663
h_bound_cnn	0.657	0.655	0.659	0.658
h_pre_cnn	0.645	0.636	0.655	0.642
pre_cnn	0.628	0.641	0.615	0.637
post_rnn	0.614	0.650	0.582	0.636
post_cnn	0.591	0.602	0.580	0.600
bound_rnn	0.556	0.562	0.551	0.562
h_post_rnn	0.530	0.566	0.499	0.560
h_post_cnn	0.537	0.553	0.522	0.552
baseline	*0.665*	*0.498*	*1.000*	*0.498*

TABLE II: Performance Metrics (t=0.37)

	F1 ⇓	Precision	Recall	Accuracy
OLS	**0.739**	**0.650**	**0.855**	**0.699**
pre_rnn	0.717	0.622	0.845	0.668
h_pre_rnn	0.709	0.605	0.854	0.650
h_bound_rnn	0.703	0.595	0.859	0.639
bound_cnn	0.694	0.607	0.812	0.644
h_bound_cnn	0.693	0.586	0.849	0.627
h_pre_cnn	0.683	0.566	0.859	0.603
post_rnn	0.681	0.565	0.857	0.601
pre_cnn	0.679	0.573	0.833	0.608
post_cnn	0.666	0.536	0.879	0.561
h_post_rnn	0.665	0.498	1.000	0.498
bound_rnn	0.665	0.498	1.000	0.498
baseline	*0.665*	*0.498*	*1.000*	*0.498*
h_post_cnn	0.663	0.499	0.989	0.500

Accuracy and F-score reach their optima at different thresholds (fig. 4), with t=0.5 achieving the maximum accuracy of 72.36% (Table I), and t=0.37 achieving the maximum F-score of 73.88% (Table II) on unseen data. A baseline of all positive predictions is included for reference. Preboundary features, specifically MFCCs one second leading up to a potential boundary, are the most valuable individually at predicting whether or not it truly marks the start/end of an IU.

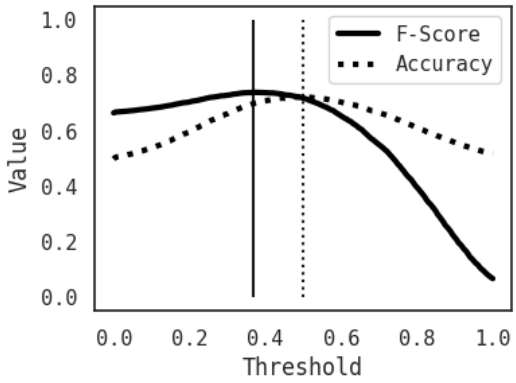

Fig. 4. F-Score reaches its maximum at a threshold of 0.37

Higher predictions lead to a higher likelihood of a true boundary and vice-versa (fig. 5).

B. Implementation

Processing time through the proposed framework is linear [O(n)], requiring 0.052 seconds per second of input audio on our system.[3]Given an imbalance of true and false boundaries in naturally occurring discourse (there are many more of the latter), we suggest using a threshold of 0.37 instead of 0.5

[3]This does not include environment initialization or resampling and may take longer without access to GPU resources.

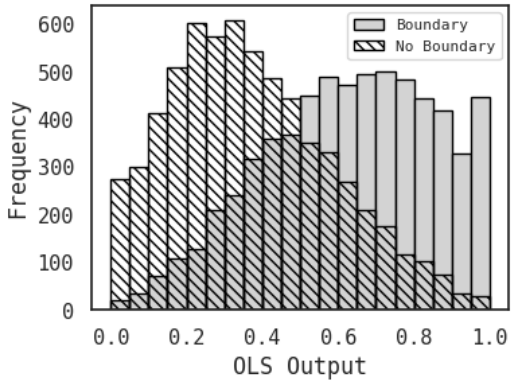

Fig. 5. OLS-predicted values versus binary truth

(fig. 6). Please see our github[4] for specific implementation instructions.

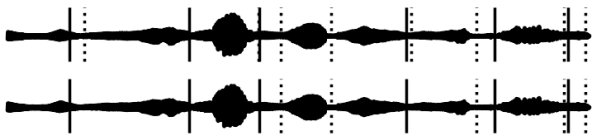

Fig. 6. Output of entire framework (dotted) versus actual transcription (solid) top: t=0.37, bottom: t=0.5 (SBC 32: 2s to 4s)

IV. Conclusion

MFCC transformations of audio data are powerful predictors of intonation unit boundaries. There still, however, remains unexplained variation. A theoretical accuracy limit of 80% (the threshold of inter-human agreement) [3] leaves much to be desired from an automatic prosodic segmentation system. We expect that hyperparameter tuning, expanded input data, and

[4]https://github.com/Nathan-Roll1/ProsodPy

more powerful models will improve the results reported in this work.

ACKNOWLEDGMENTS

We thank Dr. John Du Bois (University of California, Santa Barbara), Dr. Tirza Biron (Weizmann Institute of Science), and the University of Cambridge.

REFERENCES

[1] Du Bois JW, Cumming S, Schuetze-Coburn S, Paolino D. Discourse transcription, Santa Barbara Papers in Linguistics vol. 4. 1992.

[2] Jeon, Je Hun & Liu, Yang. (2009). Semi-supervised Learning for Automatic Prosodic Event Detection Using Co-training Algorithm. 540-548. 10.3115/1690219.1690222.

[3] Biron T, Baum D, Freche D, Matalon N, Ehrmann N, Weinreb E, et al. (2021) Automatic detection of prosodic boundaries in spontaneous speech. PLoS ONE 16(5): e0250969. https://doi.org/10.1371/journal.pone.0250969

[4] Kuhn, Melanie & Schwanenflugel, Paula & Meisinger, Elizabeth. (2010). Aligning theory and assessment of reading fluency: Automaticity, prosody, and definitions of fluency. Reading Research Quarterly. 45. 232-253.

[5] Julia Hirschberg, Communication and prosody: Functional aspects of prosody, Speech Communication, Volume 36, Issues 1–2, 2002, Pages 31-43,ISSN 0167-6393, https://doi.org/10.1016/S0167-6393(01)00024-3.

[6] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn and D. Yu, "Convolutional Neural Networks for Speech Recognition," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 22, no. 10, pp. 1533-1545, Oct. 2014, doi: 10.1109/TASLP.2014.2339736.

[7] Du Bois, John W., Wallace L. Chafe, Charles Meyer, Sandra A. Thompson, Robert Englebretson, and Nii Martey. 2000-2005. Santa Barbara corpus of spoken American English, Parts 1-4. Philadelphia: Linguistic Data Consortium.

[8] Moreno, Pedro & Joerg, Christopher & Thong, Jean-Manuel & Glickman, Oren. (1998). A recursive algorithm for the forced alignment of very long audio segments. 10.21437/ICSLP.1998-603.

[9] P. Mermelstein (1976), "Distance measures for speech recognition, psychological and instrumental," in Pattern Recognition and Artificial Intelligence, C. H. Chen, Ed., pp. 374–388. Academic, New York.

[10] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).

[11] Martín Abadi et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[12] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011