Phoneme Boundary Detection using Learnable Segmental Features

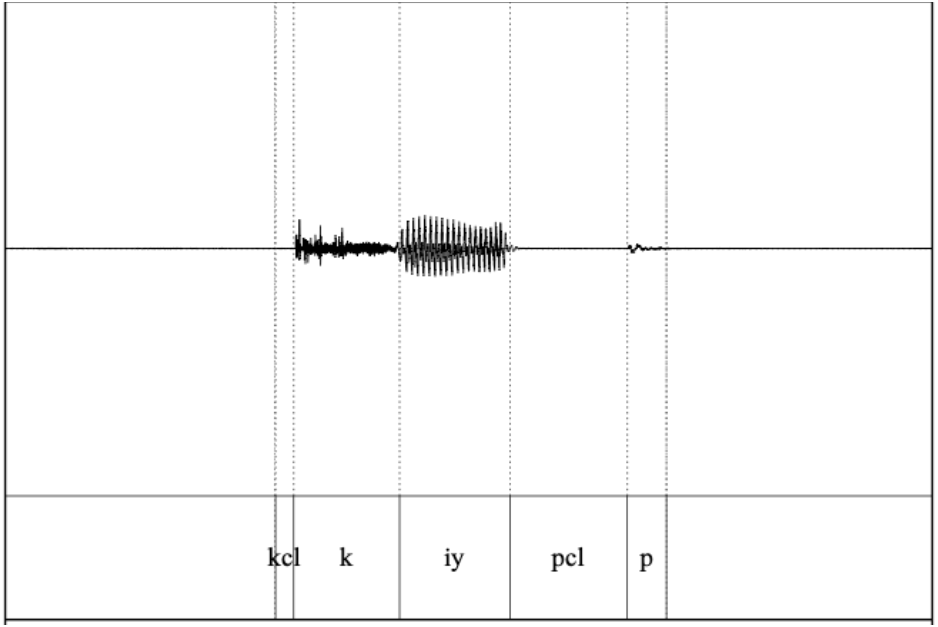
Felix Kreuk, Yaniv Sheena, Joseph Keshet, Yossi Adi

ICASSP 2020

Introduction

- **Phoneme Boundary Detection** or **Phoneme Segmentation** plays an essential first step for a variety of speech processing applications (Automatic Speech Recognition, Speech Diarization, etc)
- Supervision Types:
  - Unsupervised -- Audio only
  - Supervised -- Audio +
    - Phoneme boundaries and presumed phonemes -- **Forced Alignment**
    - Phoneme boundaries alone -- **Text-Independent Phoneme Segmentation**

Example



Introduction

- We suggest learning **segmental representation** for both **phoneme boundaries** and **phoneme segments** to detect phoneme boundaries accurately
- We do this by jointly optimizing a **Recurrent Neural Network (RNN)** with **structured loss parameters**
- We evaluate our approach using TIMIT and Buckeye datasets. The proposed method reaches state-of-the-art performance
- We additionally experiment with leveraging phoneme information as **an additional supervision** and show this to be beneficial for **performance** and **convergence speed**
- Finally, we demonstrate that such phonetic supervision does not make the proposed model language specific

Related Work

- Traditionally, in the unsupervised setting, signal processing techniques were used to find **spectral changes** in the singal, such changes are **candidates for a phoneme boundary location** [Estevan et. al 2007, Rasanen et. al 2011, Hoang and Wang 2015]
- In the supervised setting, the common approach is the **Forced Alignment** setup. Models that follow this approach involve with **HMM** and **Structured Prediction** algorithms [Keshet et. al 2005, McAuliffe et. al 2017]
- In the text-independent setting, most previous work consider the task of segmentation as a binary classification problem (one label for boundaries, one for the rest) [King and Hasegawa-Johnson 2013, Franke et. al 2016]

Model

- We denote by $\bar{\mathbf{x}} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$ a speech utterance represented by acoustic features
- Each utterance is associated with a timing sequence denoted by $\bar{\mathbf{y}} = (y_1, \ldots, y_k)$, where k is the number of segments

Model

- We denote by $\bar{\mathbf{x}} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$ a speech utterance represented by acoustic features
- Each utterance is associated with a timing sequence denoted by $\bar{\mathbf{y}} = (y_1, \ldots, y_k)$, where k is the number of segments
- Consider the following prediction rule:

$$\bar{\mathbf{y}}'_{\boldsymbol{w}}(\bar{\mathbf{x}}) = \underset{\bar{\mathbf{y}} \in \mathcal{Y}^*}{\operatorname{argmax}} \ \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{\mathbf{y}})$$

  Where $\boldsymbol{w} \in \mathbb{R}^d$ and $\boldsymbol{\phi}$ is a mapping function from the set of input objects to a real vector in $\mathbb{R}^d$

Model

- We assume the score for a segmentation can be decomposed as a sum of segmental scores:

$$\bar{\mathbf{y}}'_{\boldsymbol{w}}(\bar{\mathbf{x}}) = \underset{\bar{\mathbf{y}} \in \mathcal{Y}^*}{\operatorname{argmax}} \ \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = \ \sum_{i=1}^{k} \boldsymbol{\phi}'(\bar{\mathbf{x}}, y_i)$$

Model

- We assume the score for a segmentation can be decomposed as a sum of segmental scores:

$$\bar{\mathbf{y}}'_{\boldsymbol{w}}(\bar{\mathbf{x}}) = \underset{\bar{\mathbf{y}} \in \mathcal{Y}^*}{\operatorname{argmax}} \ \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = \ \sum_{i=1}^{k} \boldsymbol{\phi}'(\bar{\mathbf{x}}, y_i)$$

- Notice, such decomposition assumes conditional independence between boundaries

Model

- We assume the score for a segmentation can be decomposed as a sum of segmental scores:

$$\bar{\mathbf{y}}'_{\boldsymbol{w}}(\bar{\mathbf{x}}) = \underset{\bar{\mathbf{y}} \in \mathcal{Y}^*}{\operatorname{argmax}} \ \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = \ \textstyle\sum_{i=1}^{k} \boldsymbol{\phi}'(\bar{\mathbf{x}}, y_i)$$

- Notice, such decomposition assumes conditional independence between boundaries
- Practically, **information about the previous boundary can provide insight about the next one**:

$$= \underset{\bar{\mathbf{y}} \in \mathcal{Y}^*}{\operatorname{argmax}} \ \boldsymbol{w}^\top \left(\sum_{i=1}^{k} \boldsymbol{\phi}'_u(\bar{\mathbf{x}}, y_i) + \sum_{j=1}^{k-1} \boldsymbol{\phi}'_{bi}(\bar{\mathbf{x}}, y_j, y_{j+1}) \right)$$

Model

- During training, we optimize the hinge loss function as follows:

$$\ell(\boldsymbol{w}, \bar{\mathbf{x}}, \bar{\mathbf{y}}) = \max_{\bar{\mathbf{y}}' \in \mathcal{Y}^*} \left[1 - \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) + \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{\mathbf{y}}'_{\boldsymbol{w}})\right]$$

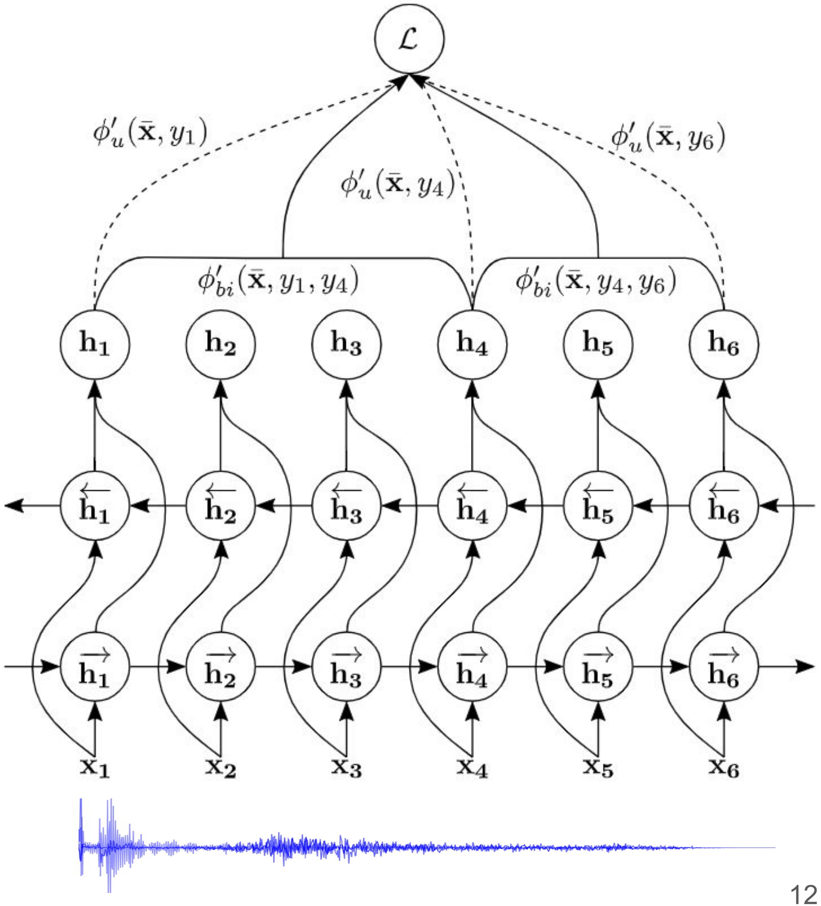
Model

Prediction rule:

$$\bar{\mathbf{y}}'_{\boldsymbol{w}}(\bar{\mathbf{x}}) = \underset{\bar{\mathbf{y}} \in \mathcal{Y}^*}{\operatorname{argmax}} \;\; \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{\mathbf{y}})$$

Loss function:

$$\ell(\boldsymbol{w}, \bar{\mathbf{x}}, \bar{\mathbf{y}}) = \max_{\bar{\mathbf{y}}' \in \mathcal{Y}^*} \;\; \left[1 - \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) + \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{\mathbf{y}}'_{\boldsymbol{w}})\right]$$



Results: Performance

Table 1. Comparison of phoneme segmentation models. Precision (P) and recall (R) are calculated with tolerance value of 20 ms

	Model	P	R	F1	R-val
TIMIT	King *et al.*[22]	87.0	84.8	85.9	87.8
	Franke *et al.*[23]	91.1	88.1	89.6	90.8
	SEGFEAT	**94.03**	**90.46**	**92.22**	**92.79**
Buckeye	Franke *et al.*[23]	**87.8**	83.3	85.5	87.17
	SEGFEAT	85.4	**89.12**	**87.23**	**88.76**

Results: Loss Ablation

Table 2. Models performance on TIMIT using different sets of loss function.

Loss	P	R	F1	R-val

Results: Loss Ablation

Table 2. Models performance on TIMIT using different sets of loss function.

Loss	P	R	F1	R-val
BIN	91.1	88.1	89.6	90.8

Results: Loss Ablation

Table 2. Models performance on TIMIT using different sets of loss function.

Loss	P	R	F1	R-val
BIN	91.1	88.1	89.6	90.8
SEGFEAT	94.03	90.46	92.22	92.79

Results: Loss Ablation

Table 2. Models performance on TIMIT using different sets of loss function.

Loss	P	R	F1	R-val
BIN	91.1	88.1	89.6	90.8
SEGFEAT	94.03	90.46	92.22	92.79
SEGFEAT +PHN	92.98	92.33	92.66	93.69

Results: Loss Ablation

Table 2. Models performance on TIMIT using different sets of loss function.

Loss	P	R	F1	R-val
BIN	91.1	88.1	89.6	90.8
BIN + PHN	**96.6**	85.0	90.04	89.33
SEGFEAT	94.03	90.46	92.22	92.79
SEGFEAT +PHN	92.98	92.33	92.66	93.69
SEGFEAT +PHN +BIN	92.67	**93.03**	**92.85**	**93.91**

Results: Loss Ablation

Table 4. An ablation study on the effect of the PHN loss on Hebrew language.

Model	P	R	F1	R-val
SEGFEAT w/o PHN LOSS	**83.58**	79.2	81.24	83.67
SEGFEAT w PHN LOSS	83.11	**81.66**	**82.38**	**84.92**

Results: Loss Ablation

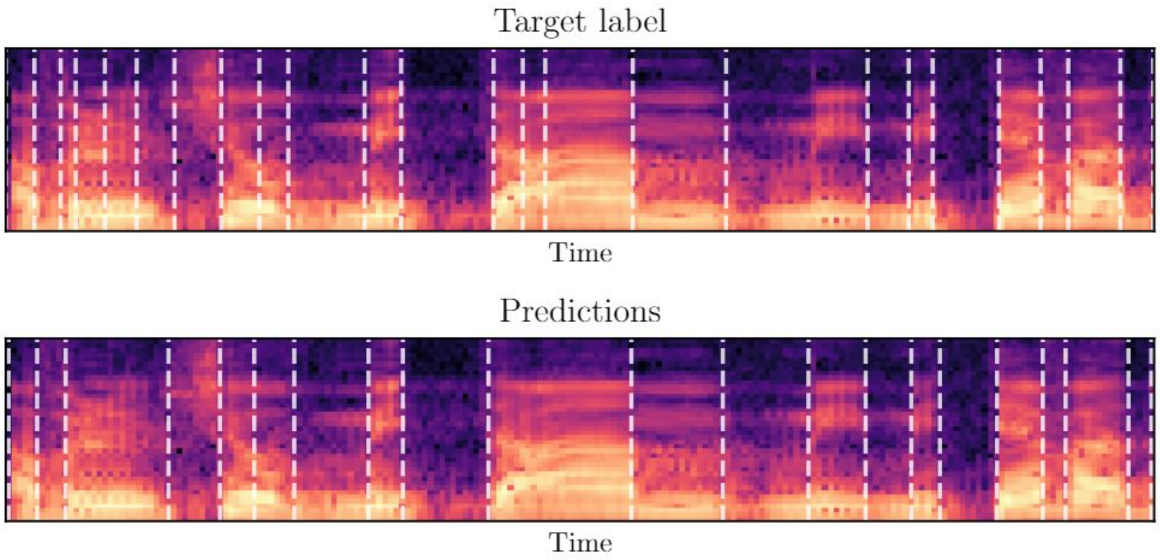


Fig. 3. Example of segmentation result on an Hebrew utterance using an English trained model.

Results: Comparison to Forced Alignment

Table 3. Comparison of the proposed model against forced-alignment algorithms.

Model	P	R	F1	R-val
McAuliffe (unsup.) [21]	83.9	81.6	82.7	85.16
Keshet (sup.) [20]	90	82.2	85.9	79.51
SegFeat	**94.03**	**90.46**	**92.22**	**92.79**

Summary

- Moving from point scores to segmental scores
- Additional phoneme supervision gains (performance, convergence)
- Generalization to multilingual setup

Future Work

- Unsupervised Phoneme Segmentation
- Systematic comparison in a multilingual setting

Thank you!

felix.kreuk@gmail.com