SEQUENCE SEGMENTATION USING JOINT RNN AND STRUCTURED PREDICTION MODELS

Yossi Adi, Joseph Keshet, Emily Cibelli, Matt Goldrick

Department of Computer Science, Bar-Ilan University, Ramat-Gan, Israel | Department of Linguistics, Northwestern University, Evanston, IL, USA

Abstract

We describe and analyze a simple and effective algorithm for sequence segmentation applied to speech processing tasks. We propose a neural architecture that is composed of two modules trained jointly: a recurrent neural network (RNN) module and a structured prediction model. The RNN outputs are considered as feature functions to the structured model. The overall model is trained with a structured loss function which can be designed to the given segmentation task. We demonstrate the effectiveness of our method by applying it to two simple tasks commonly used in phonetic studies: word segmentation and voice onset time segmentation. Results suggest the proposed model is superior to previous methods, obtaining state-of-the-art results on the tested datasets.

Proposed Model

We would like to minimize the following surrogate loss function:

$$F(\boldsymbol{w}, \bar{\mathbf{x}}, \bar{\mathbf{y}}) = \frac{1}{m} \sum_{i=1}^{m} \bar{\ell}(\boldsymbol{w}, \bar{\mathbf{x}}, \bar{\mathbf{y}})$$

where,

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

Usually, phi is manually chosen. In this work we consider RNN outputs as feature functions to the segmentation task. It enables us optimizing in an end-to-end fashion.
Our inference function can be formulated as follows:

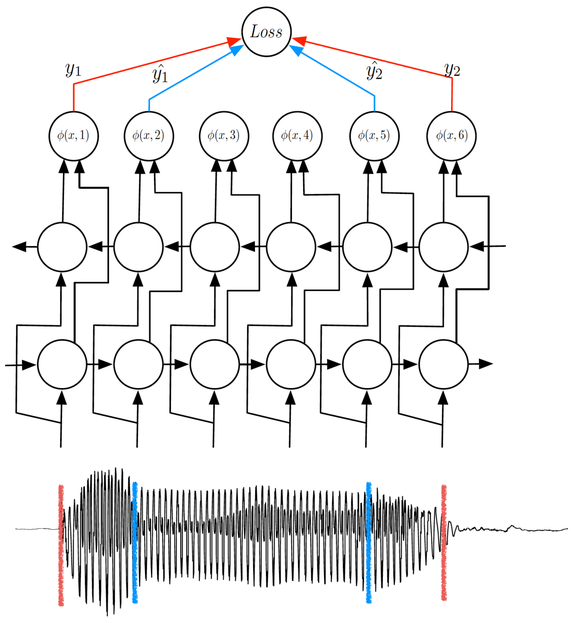
$$\begin{aligned} \bar{\mathbf{y}}'_{\boldsymbol{w}}(\bar{\mathbf{x}}) &= \operatorname*{argmax}_{\bar{\mathbf{y}} \in \mathcal{Y}^p} \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) \\ &= \operatorname*{argmax}_{\bar{\mathbf{y}} \in \mathcal{Y}^p} \boldsymbol{w}^\top \sum_{i=1}^{p} \boldsymbol{\phi}'(\bar{\mathbf{x}}, y_i) \\ &= \operatorname*{argmax}_{\bar{\mathbf{y}} \in \mathcal{Y}^p} \boldsymbol{w}^\top \sum_{i=1}^{p} \mathrm{RNN}(\bar{\mathbf{x}}, y_i), \end{aligned}$$

In order to plug it into a neural setting all we need to do is to compute the derivatives w.r.t the parameters and inputs.

The task loss function we use in all of the segmentation experiments is:

$$\ell(\bar{\mathbf{y}}, \bar{\mathbf{y}}') = \left[|y_1 - y'_1| - \tau \right]_+ + \left[|y_2 - y'_2| - \tau \right]_+,$$

where τ is a tolerance value.



Experiments

We compare our model to both a deep model with an NLL loss function and to a linear model with structured loss function. We investigate two segmentation problems; **Word Segmentation** and **Voice Onset Time (VOT) Segmentation**. Those are roughly simple problems, but still demonstrate the efficiency of the proposed model. We leave the extension to more complex problems for future work.

Word Segmentation

Results are reported for the mean onset and offset difference between the manual and automatic measurement. The loss function was measured using the task loss (with τ=0) in frames of 10ms.

	RNN	2-RNN	Bi-RNN	Bi-2-RNN	DeepSeg.
Onset	6.0	5.84	2.88	3.48	**2.02**
Offset	9.43	8.92	4.46	**3.75**	3.96

VOT Segmentation

Proportion of differences between automatic and manual measures falling at or below a given tolerance value (in msec) for the AutoVOT - Linear structured model, DeepVOT - Deep model with NLL loss function and DeepSeg. - Deep model with structured loss.
Left value are results for PGWORDS dataset and results in the right value are for BB dataset.

	t ≤ 2	t ≤ 5	t ≤ 10	t ≤ 15	t ≤ 25
AutoVOT	49.1 / 59.1	81.3 / 80.5	93.9 / 89.9	96.0 / 94.3	97.2 / 96.8
DeepVOT	53.8 / 60.3	91.6 / 84.2	**97.6 / 94.3**	**98.7** / 94.9	**99.6 / 98.1**
DeepSeg.	**78.2 / 64.8**	**94.1 / 85.5**	97.1 / **94.3**	98.6 / **95.0**	99.1 / 96.2