BLSTM-CRF Based End-to-End Prosodic Boundary Prediction with Context Sensitive Embeddings in A Text-to-Speech Front-End

Yibin Zheng[1,2], *Jianhua Tao*[1,2], *Zhengqi Wen*[1], *Ya Li*[1]

[1]National Laboratory of Pattern Recognition, Institute of Automation, CAS, China
[2]School of Artificial Intelligence, University of Chinese Academy of Science, China
yibin.zheng, jhtao, zqwen, yli@nlpr.ia.ac.cn

Abstract

In this paper, we propose a language-independent end-to-end architecture for prosodic boundary prediction based on BLSTM-CRF. The proposed architecture has three components, word embedding layer, BLSTM layer and CRF layer. The word embedding layer is employed to learn the task-specific embeddings for prosodic boundary prediction. The BLSTM layer can efficiently use both past and future input features, while the CRF layer can efficiently use sentence level information. We integrate these three components and learn the whole process end-to-end. In addition, we investigate both character-level embeddings and context sensitive embeddings to this model, and employ an attention mechanism for combining alternative word-level embeddings. By using an attention mechanism, the model is able to decide how much information to use from each level of embeddings. Objective evaluation results show the proposed BLSTM-CRF architecture achieves the best results on both Mandarin and English datasets, with an absolute improvement of 3.21% and 3.74% in F1 score, respectively, for intonational phrase prediction, compared to previous state-of-the-art method (BLSTM). The subjective evaluation results further indicate the effectiveness of the proposed methods.

Index Terms: prosodic boundary prediction, BLSTM-CRF, attention, context sensitive embeddings, end-to-end

1. Introduction

Prosody structure plays an important role in both naturalness and intelligibility of speech [1]. It splits an utterance into prosodic units which can be easily understood by people. Even the newly developed speech synthesis architecture, WaveNet [2], still required the prosodic features derived from the text. Therefore, identifying the phrase boundaries of different prosodic units from text is crucial in speech synthesis.

Previous researches on automatic prediction of prosodic boundaries could be classified into two main categories. One focuses on the aspect of feature engineering. These studies investigate a great number of features and their relevance to prosodic boundaries prediction [3–7]. Recently, some syntactic features [8, 9] and embedding features [10–12] have been employed to augment or replace the traditional linguistic features (like POS, word-terminal syllables etc.). The other category focuses on the aspect of modeling methods. Some statistical machine learning methods, like maximum entropy, conditional random field (CRF) and deep recurrent neural network (RNN) [13–18], have been investigated. Among these methods, the best reported results in shallow and deep model were achieved with CRF [15] and bi-directional long-short term memory (BLSTM) recurrent neural network [10–12], respectively.

With the previous researches on prosodic boundary prediction, this task has been accomplished by dividing it into two stages. The first stage is to extract rich features from raw texts [3–12], while the second stage takes as input the rich features and predicts corresponding boundary for each word. Even the newly developed methods [10–12] had to transform the raw texts to word embeddings first before feeding them into the prediction model. This two-stage procedure is often time-consuming and language-specific, since it requires much expert linguistic knowledge to define linguistic features [7–9]. In this paper, we integrate these two stages and learn the whole process end-to-end, which removes a major bottleneck in modeling prosodic boundary for new languages.

Different from [10–12] (which directly took as input the pre-trained word embeddings), we add an embedding layer in our proposed architecture to induce task-specific word embeddings for prosodic boundary prediction. Besides that, [10–12] still treated words as atomic units and ignored any surface or morphological similarities between different words. However, in many languages such as Chinese, the meaning of a word is also related to its composing characters. To take advantage of this regularity, we investigate character-level extension to this model and use an attention mechanism for combining alternative word-level embeddings. In addition, the word- or character-level embeddings captures only the semantic and syntactic information of a word. However, in many natural language processing (NLP) tasks [19], it's essential to represent not only the meaning of a word, but also the word in context. Therefore, we use a bi-directional language model (LM), pre-trained on a large, unlabeled corpus to compute the embeddings of context at each position in the sequence (hereafter context sensitive embeddings) and use it in the prosodic boundary prediction.

In our previous research [12], we trained CRF- and BLSTM-based model separately and then made fusion at decision-level. This greatly increased the complexity of the training process. Therefore, a complementary research we focus on here is to combine BLSTM and CRF to form a BLSTM-CRF model. This model can take advantages of both: a BLSTM layer can efficiently use both past and future input features; while a CRF layer can efficiently use sentence level information to make the boundary prediction.

In this paper, we propose a language-independent end-to-end architecture for prosodic boundary prediction based on BLSTM-CRF. Our contributions can be summarized as follows. (1) We propose a novel BLSTM-CRF based architecture for prosodic boundary prediction. The architecture combines feature induction and prosodic boundary prediction in a unified framework to learn the whole process end-to-end. (2) We apply an attention mechanism to combine both character embeddings and context sensitive embeddings with the word-level embeddings features. (3) Our proposed method is language-independent, which can be extended to other languages without any expert linguistic knowledge to define linguistic features.

10.21437/Interspeech.2018-1472

2. BLSTM-CRF based end-to-end model

Fig.1 shows the general architecture of the word-level BLSTM-CRF based end-to-end model for prosodic boundary prediction. The first layer of the architecture is an embedding layer, which maps the raw input words into word embeddings for processing by subsequent layers. After the embedding layer, there is a BLSTM-CRF based layer. The layer receives a sequence of word embeddings as inputs, and predicts a label (Break or No Break) corresponding to each of the input words. The whole architecture is learned jointly.

2.1. Embedding layer

Unlike [10–12], which directly took as input the pre-trained word embeddings (The word embeddings wouldn't be updated during model training.), we add an embedding layer to induce task-specific word representation, i.e., the pre-trained word embeddings would be fine-tuned for prosodic boundary prediction task. For simplicity, we consider words as indices in finite dictionary D of N unique word tokens. An input sentence $w_1, ..., w_T$ of T words is thus transformed by the first embedding layer into a sequence of word embeddings $x_1, ..., x_T$, by applying the lookup table operation. In this work, a pre-trained word embedding model presented in Section 5.2 is employed to generate the initial dictionary D.

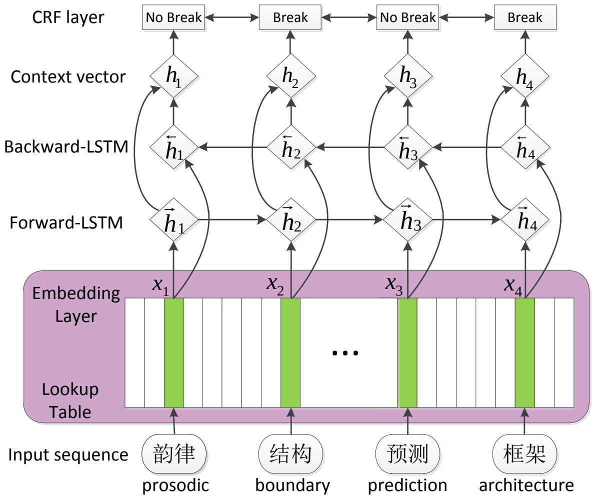

Figure 1: *The general architecture of word-level BLSTM-CRF based end-to-end model for prosodic boundary prediction.*

2.2. BLSTM-CRF model

Inspired by the successful use of BLSTM [10–12] and CRF [15] in prosodic boundary prediction, we employ a hybrid architecture (BLSTM-CRF) to take the advantages of both. The BLSTM-CRF based architecture was used in many NLP tasks and achieved the state-of-art performance [20–22].

2.2.1. BLSTM

As shown in Fig.1, the word embeddings generated by the embedding layer are given as the inputs to two LSTM [23] components moving in opposite directions through the text. For each time step t, each LSTM takes as the input the hidden state from previous time step, along with the word embedding from current step x_t, and outputs a new hidden state. The final representations of a word by BLSTM [24] are obtained by concatenating the hidden representation from both directions, resulting in representations that are conditioned on the whole sequence:

$$\overrightarrow{h}_t = LSTM(\overrightarrow{h}_{t-1}, x_t); \overleftarrow{h}_t = LSTM(\overleftarrow{h}_{t+1}, x_t); \quad h_t = [\overrightarrow{h}_t, \overleftarrow{h}_t] \tag{1}$$

where *LSTM* is the implementation of LSTM [25].

2.2.2. CRF

To produce the label predictions, a simple and common way is to use the hidden representation h_t to make independent prediction for each word, such as [11]. But for prosodic boundary prediction task, there are dependencies between successive tags and thus it is beneficial to model and decode each sentence jointly. Accordingly, we add a CRF layer at the output of the proposed architecture (Fig.1), which allows the network to look for the most optimal path through all possible sequences. During training, the model is then optimized by maximizing the score of the correct tag sequence $y = (y_1, y_2, ..., y_T)$, while minimizing the scores for all other sequences:

$$E = -s(y) + log \sum_{\widetilde{y} \in \widetilde{Y}} e^{s(\widetilde{y})} \tag{2}$$

where $s(y)$ is the CRF score for a sequence y, and $\widetilde{Y}$ represents all possible tag sequences.

3. Character-level representation

In [12], we have compared the performance of character-level models with word-level models for prosodic boundary prediction, and found models that operate exclusively on characters were not yet competitive to word-level models. Therefore, instead of fully replacing word embeddings, we employ an attention mechanism to allow the model to take advantage of information at both granularity levels in this paper.

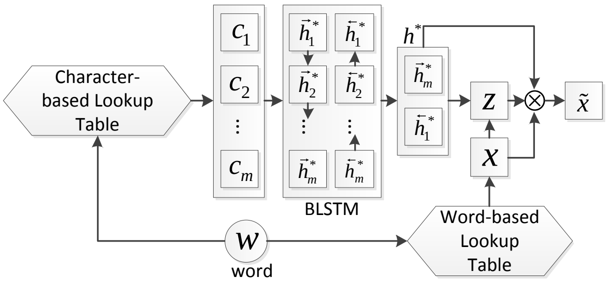

Figure 2: *Combining character-level embeddings with word-level embeddings using an attention mechanism.*

3.1. Character-level components

The general flowchart of the method to combining character embeddings with word embeddings is shown in Fig.2. The individual characters of a word are mapped to character embeddings $(c_1, ..., c_m)$ by applying the lookup table operation, then they are encoded by a BLSTM:

$$\overrightarrow{h}^*_i = LSTM(\overrightarrow{h}^*_i, c_t); \overleftarrow{h}^*_i = LSTM(\overleftarrow{h}^*_{i+1}, x_t) \tag{3}$$

The last hidden state from both direction are then concatenated to form an alternative representation h^* for each word that built from individual characters:

$$h^* = [\overrightarrow{h}^*_m, \overleftarrow{h}^*_1] \tag{4}$$

3.2. Combining with regular word-level embeddings

We use an attention mechanism [22] to adaptively control the balance between word-level (x) and character-level (h^*) embeddings. As shown in Fig.2, these two vectors (x and h^*) are added together using a weighted sum, where the weights are predicted by a two-layer network:

$$z = \sigma(W_z^{(3)} tanh(W_z^{(1)} x + W_z^{(2)} h^*) \tag{5}$$

$$\widetilde{x} = z \cdot x + (1 - z) \cdot h^* \tag{6}$$

where $W_z^{(1)}$, $W_z^{(2)}$ and $W_z^{(3)}$ are weight matrices for calculating z, $\sigma()$ is the logistic function with values in the range [0, 1], and $\widetilde{x}$ is the new word representation. The model could dynamically learn how much information to use from the character-level or word-level component by different vector value of z.

4. Context sensitive embeddings from LM

In many NLP tasks including the prosodic boundary prediction, it's essential to represent not only the meaning of a word, but also the word in context. Therefore, we explore to use an unsupervised method to learn context sensitive embeddings from a LM, which requires no additional annotated training data.

4.1. Language model (LM)

Given a sequence of text $w = (w_1, ..., w_T)$, a language model (LM) computes the probability of w as:

$$p(w_1, ..., w_T) = \prod_{t=1}^{T} p(w_t | (w_1, ..., w_{T-1})) \tag{7}$$

We use bi-directional LSTM LM [26] to capture both past and future context here. At time step t, a forward LSTM LM receives input w_t and predicts w_{t+1}. Using Eq.8, it first computes representation x_t of w_t. Given this representation and previous state $\overrightarrow{h}_{t-1}^{LM}$, it produces a new state $\overrightarrow{h}_{t}^{LM}$ and predicts $w_{(t+1)}$:

$$\overrightarrow{h}_{t}^{LM} = LSTM(\overrightarrow{h}_{t-1}^{LM}, x_t) \tag{8}$$

$$w_{t+1} = g(D^T \cdot \overrightarrow{h}_{t}^{LM}) \tag{9}$$

where g is a softmax function over the dictionary D. A backward LSTM LM could be implemented in an analogous way to a forward LM and produces backward context sensitive embeddings $\overleftarrow{h}_{t}^{LM}$. After pre-training the forward and backwards LM separately, we remove the top layer softmax and concatenate the forward and backward context sensitive embeddings to form bi-directional context sensitive embeddings:

$$h_t^{LM} = [\overleftarrow{h}_{t}^{LM}; \overrightarrow{h}_{t}^{LM}] \tag{10}$$

Since the context sensitive embeddings are used to compute the probability of next words in a bi-dirctional LM, they are likely to encode both semantic and syntactic informatic roles of word in context.

4.2. Integration into prosodic boundary prediction

Our final system uses the bi-directional context sensitive embeddings h_t^{LM} as additional inputs for the prosodic boundary prediction. Instead of simply concatenating h_t^{LM} with the new word representation $\widetilde{x}$, we found that using an attention mechanism that weights all the embeddings in a sentence gains more benefits. Therefore, the context sensitive embeddings h_t^{LM} are combined with the new word representations $\widetilde{x}$ using a dynamic weighting mechanism which have been introduced in Section 3.2.

5. Experiments and result analysis

5.1. Dataset

We evaluate the proposed methods on both Mandarin and English dataset. These two datasets are both recorded for speech synthesis task. The prosodic boundaries were labelled by three expert annotators by both listening the utterances and reading the transcriptions. Before annotation, they were trained several times to achieve most of the annotations. The average label consistency between all three annotators is 87.6%. The training/validation/test split is 8:1:1 for all the experiments. The description of each dataset and their preprocessing are as follows:

Mandarin: a dataset with 60,000 sentences. Prosodic boundaries (prosodic word (PW), prosodic phrase (PPH) and intonational phrase (IPH)) were then labelled. This hierarchical prosodic structure [5] is widely employed in Mandarin.

English: a dataset with 10,000 sentences. Only the intonational phrase (IPH) boundary was labelled in this dataset.

5.2. Experimental setting

For Mandarin, 15 GBytes Mandarin text corpus [27] is collected to pre-train embeddings using word2vec [28]. Both character and word embeddings dimension are set to 100 for Mandarin. For English, we use 300-dimensional pre-trained word embedding trained on Google News [29]. The dimension of character embeddings is set to 50 and initialized randomly for English. These two text corpora are also used to pre-train the LM.

All the BLSTM-related architectures (including the LM) have two hidden layers; each layer contains 160 memory blocks in each direction. Parameters of the proposed models are optimized using AdaDelta [30] with default learning rate at 0.001. To mitigate overfitting, the dropout method [31] is employed to regularize our model and the dropout rate is set at 0.5 through all the experiments.

For Mandarin, different models are trained to predict different level of prosodic boundaries, and the predicted boundary y_l rom the lower level is used as an input feature (concatnated to the final word representation, ie., $[\widetilde{x}; y_l]$ for the current boundary prediction. Based on these, the following systems are built. All the systems are trained by Theano [32] toolkit.

1. **CRF**: Traditional linguistic features are used for CRF based prosodic boundaries prediction. These features include POS tags, the length of words etc. that were presented in [12] .
2. **BL**: Pre-trained word embeddings are directly used as input features (like [10–12]) for BLSTM based prosodic boundaries prediction without an embedding layer.
3. **BC**: Adding a **CRF** layer on the top of the system **BL**.
4. **WB**: Adding **an embedding layer** to initialize and fine-tune the word embeddings at the input of system **BC**.
5. **CC**: Adding **character-level components** by an attention mechanism from Section 3 to system **WB**.
6. **CA**: Adding **context sensitive embeddings** by an attention mechanism from Section 4 to system **CC**.

5.3. Results in Mandarin dataset

5.3.1. Evaluation of the BLSTM-CRF model

To evaluate the effectiveness of the proposed BLSTM-CRF model, we compare the results of system BC with two baseline systems CRF and BL. System BL serves as a strong baseline system here as it achieved state-of-the-art performance in previous researches. System BC significantly outperforms these two baseline systems on all three prosodic boundaries, especially in higher boundaries (with an absolute gain of 1.69% in F1 score at the IPH level prediction). Even we find this improvement is more obvious and with a less model complexity than that of in our previous work [12] (1.14% in F1 score) where we trained CRF- (system CRF) and BLSTM-based (system BL) models separately and then made linear fusion at decision-level. It can be explained by linear fusion method is just a linear combination of two single models, while the proposed BLSTM-CRF model can take full advantages of BLSTM and CRF, with BLSTM could use both past and future input features and CRF could use sentence level sequence information.

Table 1: *F1 score for different systems in Mandarin.*

Systems	CRF	BL	BC	WB	CC	CA
PW	95.46	95.60	96.01	96.26	96.49	**96.87**
PPH	79.39	80.15	81.49	81.84	82.18	**82.95**
IPH	77.54	78.88	80.57	81.00	81.39	**82.09**

5.3.2. Evaluation of embedding layer

An observation of the results for system BC and WB, shows that there is an improvement in F1 score when adding an embedding layer for the prosodic boundary prediction, rather than using pre-trained embeddings as inputs directly. This validates our hypothesis that adding an embedding layer could induce task-specific embeddings for the prosodic boundary prediction, since the parameters in the embedding layer are fine-tuned to predict the prosoidc boundaries during training. More importantly, after adding embedding layer (system WB), we can combine feature induction and prosodic boundary prediction in a unified end-to-end framework and thus avoid the two-stage processes that were required in system BC.

5.3.3. Evaluation of character-based components

Tab.1 shows that system CC achieves performance superior to system WB. This indicates the necessity for taking into account of the character components. This can be explained by the fact that character components may also carry some semantic information, which is useful for prosodic boundary prediction, since a Chinese word with the similar characters may have similar meaning, such as ”危险” (danger) and ”惊险” (thrill).

5.3.4. Evaluation of context sensitive embeddings

As shown in Tab.1, compared to system CC, the system CA (which considers context sensitive embeddings) outperforms the former on all evaluations. And this is a statistically significant increase over state-of-the-art method (BL), with an absolute increase of 3.21% at the IPH level prediction. This indicates that the added context sensitive embeddings could provide much richer representation for prosodic boundary prediction. Meanwhile, the attention mechanism for dynamically deciding how much context sensitive embeddings information to use allows the model to better control the balance between the word representations, giving it an advantage in the experiments.

5.4. Extension to other language

To demonstrate our proposed methods' ability to generalize to different languages, we test our methods on English dataset and the results are presented in Tab.2. A similar pattern that shown in Mandarin could be seen in English as well. Especially, it's noticed that adding character-based components (CC) can bring substantial gains in English, with an absolute gain of 0.94% in F1 score (compared to system WB). Such improvement is much more pronounced than that in Mandarin (0.39%). This can be explained by morphemes (root, prefix or suffix) in English carrying much semantic information, and the character-level components in English have the potential of capturing morpheme patterns, thereby improving generalization representation of words. Compared to previous state-of-the-art methods (system BL), system CA shows an absolute increase of 3.74% at the IPH level prediction. The substantial gains gotten in English dataset can in a way show our proposed methods are language-independent, which can be extended to other languages without any expert linguistic knowledge to define linguistic features.

Table 2: *F1 score for different systems in English.*

Systems	CRF	BL	BC	WB	CC	CA
IPH	75.24	75.78	77.44	77.79	78.73	**79.52**

5.5. Subjective evaluation results

We further conducted an AB preference test on the naturalness of the synthesized speech. We compared system CA (that achieves the best performance in F1 score) with system BL (the previous state-of-the-art methods). A set of 20 sentences was randomly selected from test set with different prosodic boundary prediction results and speech was generated through a typical BLSTM-based TTS system. A group of 14 subjects were asked to choose which one was better in terms of the naturalness of synthesis speech. The percentage preference is shown in Fig.3. We can clearly see that the proposed method (system CA) can achieve better naturalness of synthesis speech as compared to the baseline system BL.

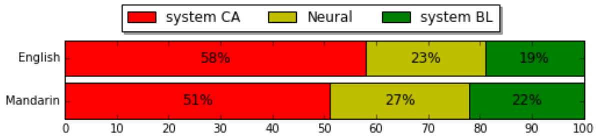

Figure 3: *The preference of AB test on Mandarin and English dataset, with confidence level of 95% and p-value* < 0.0001.

6. Conlusions

In this paper, we present a BLSTM-CRF based language-independent end-to-end architecture for prosodic boundary prediction. We first show that the embedding layer is able to learn task-specific embeddings for prosodic boundary prediction. Then, we show that the BLSTM-CRF model can efficiently use both past and future input features thanks to the BLSTM layer, and can also use sentence level information thanks to a CRF layer. Finally, we investigate both character-level embeddings and context sensitive embeddings to our models and employ an attention mechanism for combining alternative word-level embeddings. In future, we wish to explore the use of our proposed methods for other aspects of prosody prediction such as sentential stress prediction for speech synthesis.

7. Acknowledgements

This work is supported by the National Natural Science Foundation of China (NSFC) (No.61425017, No. 61773379, No. 61603390, No. 61771472), the National Key Research & Development Plan of China (No. 2017YFC0820602) and Inria-CAS Joint Research Project (173211KYSB20170061).

8. References

[1] Z. Chen, G. Hu, and W. Jiang, “Improving prosodic phrase prediction by unsupervised adaptation and syntactic features extraction,” in *INTERSPEECH 2010, Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September*, 2010, pp. 1421–1424.

[2] A. V. D. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “Wavenet: A generative model for raw audio,” 2016.

[3] Y. Qian, Z. Wu, X. Ma, and F. Soong, “Automatic prosody prediction and detection with conditional random field (crf) models,” in *International Symposium on Chinese Spoken Language Processing*, 2011, pp. 135–138.

[4] S. Ananthakrishnan and S. S. Narayanan, “An automatic prosody recognizer using a coupled multi-stream acoustic model and a syntactic-prosodic language model,” in *ICASSP ’05). IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2005, pp. 269–272.

[5] M. Chu and Y. Qian, “Locating boundaries for prosodic constituents in unrestricted mandarin texts,” *Computational Linguistics and Chinese Language Processing*, vol. 6, no. 1, pp. 61–82, 2008.

[6] O. Watts, J. Yamagishi, and S. King, “Unsupervised continuous-valued word features for phrase-break prediction without a part-of-speech tagger,” in *INTERSPEECH 2011, Conference of the International Speech Communication Association, Florence, Italy, August*, 2011, pp. 2157–2160.

[7] N. Dehé, I. Feldhausen, and S. Ishihara, “The prosody–syntax interface: Focus, phrasing, language evolution,” *Lingua*, vol. 121, no. 13, pp. 1863–1869, 2011.

[8] H. Che, Y. Li, J. Tao, and Z. Wen, “Investigating effect of rich syntactic features on mandarin prosodic boundaries prediction,” *Journal of Signal Processing Systems*, vol. 82, no. 2, pp. 263–271, 2016.

[9] G. Bailly, “Integration of rhythmic and syntactic constraints in a model of generation of french prosody,” *Speech Communication*, vol. 8, no. 2, pp. 137–146, 1989.

[10] C. Ding, L. Xie, J. Yan, W. Zhang, and Y. Liu, “Automatic prosody prediction for chinese speech synthesis using blstm-rnn and embedding features,” in *Automatic Speech Recognition and Understanding*, 2016, pp. 98–102.

[11] A. Rendel, R. Fernandez, R. Hoory, and B. Ramabhadran, “Using continuous lexical embeddings to improve symbolic-prosody prediction in a text-to-speech front-end,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 5655–5659.

[12] Y. Zheng, Y. Li, Z. Wen, X. Ding, and J. Tao, “Improving prosodic boundaries prediction for mandarin speech synthesis by using enhanced embedding feature and model fusion approach,” in *INTERSPEECH*, 2016, pp. 3201–3205.

[13] B. Busser, W. Daelemans, and A. V. D. Bosch, “Predicting phrase breaks with memory-based learning,” *Proceedings of Isca Tutorial and Research Workshop on Speech Synthesis Edimburgh*, pp. 29–34, 2001.

[14] V. K. Rangarajan Sridhar, S. Bangalore, and S. S. Narayanan, “Exploiting acoustic and syntactic features for automatic prosody labeling in a maximum entropy framework,” *IEEE Transactions on Audio Speech and Language Processing*, vol. 16, no. 4, pp. 797–811, 2008.

[15] A. Rosenberg, R. Fernandez, and B. Ramabhadran, “Phrase boundary assignment from text in multiple domains,” in *Interspeech*, 2012.

[16] R. Fernandez and B. Ramabhadran, “Discriminative training and unsupervised adaptation for labeling prosodic events with limited training data,” in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[17] I. Read and S. Cox, “Automatic pitch accent prediction for text-to-speech synthesis,” 2007.

[18] A. Rosenberg, R. Fernandez, and B. Ramabhadran, “Modeling phrasing and prominence using deep recurrent learning,” in *Interspeech*, 2015.

[19] C. Vania and A. Lopez, “From characters to words to in between: Do we capture morphology?” pp. 2016–2027, 2017.

[20] Z. Huang, W. Xu, and K. Yu, “Bidirectional lstm-crf models for sequence tagging,” *Computer Science*, 2015.

[21] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, “Neural architectures for named entity recognition,” *arXiv preprint arXiv:1603.01360*, 2016.

[22] M. Rei, G. K. O. Crichton, and S. Pyysalo, “Attending to characters in neural sequence labeling models,” 2016.

[23] A. Graves, *Long Short-Term Memory*. Springer Berlin Heidelberg, 2012.

[24] R. K. Ando and T. Zhang, *A Framework for Learning Predictive Structures from Multiple Tasks and Unlabeled Data*. JMLR.org, 2005.

[25] W. Zaremba, I. Sutskever, and O. Vinyals, “Recurrent neural network regularization,” *Eprint Arxiv*, 2014.

[26] M. Rei, “Semi-supervised multitask learning for sequence labeling,” in *Meeting of the Association for Computational Linguistics*, 2017, pp. 2121–2130.

[27] S. Lai, K. Liu, S. He, and J. Zhao, “How to generate a good word embedding,” *IEEE Intelligent Systems*, vol. 31, no. 6, pp. 5–14, 2016.

[28] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *Computer Science*, 2013.

[29] https://code.google.com/archive/p/word2vec/.

[30] M. D. Zeiler, “Adadelta: An adaptive learning rate method,” *Computer Science*, 2012.

[31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[32] T. D. Team, R. Alrfou, G. Alain, A. Almahairi, C. Angermueller, D. Bahdanau, N. Ballas, F. Bastien, J. Bayer, and A. Belikov, “Theano: A python framework for fast computation of mathematical expressions,” 2017.