Speech Prosody 2016
31 May - 3 Jun 2106, Boston, USA

A Study on BLSTM-RNN-based Chinese Prosodic Structure Prediction in a Unified Framework with Character-level Features

Yi Zhao[1], *Chuang Ding*[2], *Nobuaki Minematsu*[1], *Daisuke Saito*[1]

[1]The University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan
[2] School of Computer Science, Northwestern Polytechnical University, Xi'an, 710068, China
{zhaoyi,mine,dsk_saito}@gavo.t.u-tokyo.ac.jp, cding@nwpu-aslp.org

Abstract

In Text-to-Speech system, prosodic attributes have to be predicted only from input text. The accuracy of prosody prediction has a significant effect on the naturalness of synthesized speech of Chinese. In this paper, we explore using neural networks to predict prosodic boundaries from Chinese text without task specific knowledge or sophisticated feature engineering. We examine sequence character-level features and word-level features, and compare their performance with one-hot and embedding representations. Instead of traditional cascaded prediction, we propose a unified framework which can be considered to be a multi-task learning process. Experimental results show that character-level features can obtain approximate F-scores compared to those with word-level features, and embedding features learned from large unlabeled texts can help to enhance the performance. The unified framework can achieve similar performance to cascaded framework, while using less training time and without the necessary of preparing task-specific features.

Index Terms: Chinese prosodic structure, unified prediction, neural network, embedding features, speech synthesis

1. Introduction

In linguistics, prosody is concerned with properties of syllables and larger units of speech, which contribute to linguistic functions such as phrase-based chunking by intonation, rhythmical organization of an utterance using lexical stress, and so on. Prosody can also reflects various extra- or para-linguistic aspects of various characteristics of speaker or utterance: the emotional state of speaker, the form of utterance, the presence of irony or sarcasm, emphasis, contrast, and focus [1]. These clearly indicate that the accuracy of prosody prediction has a significant effect on the naturalness of synthesized speech.

In Chinese TTS systems, to specify the prosodic structure of a given text, the following hierarchical features have to be predicted automatically [2, 3]: 1) prosodic word boundaries (PW), prosodic phrase boundaries (PP), and intonational phrase boundaries (IP), as is shown in Fig 1. The leaf nodes of the tree structure are lexical words which are deduced from a word segmentation module. A great number of linguistic features and various prosody modeling methods have been investigated in previous research. Some syntactic cues like part-of-speech (POS), syllable identity, syllable stress and their contextual counterparts are commonly used for prosodic structure prediction [4, 5, 6]. Many statistical methods have been investigated to predict prosodic structure, including classification and regression tree (CART) [7], hidden Markov model (HMM) [8], maximum entropy model (MEM) [9] and conditional random fields (CRF) [10]. Due to its ability of relaxing assumption of strong model independence and solving label bias problem, CRF has achieved superior performance in prosodic structure prediction [11, 12, 13]. However, CRF still suffers two major drawbacks. First, its performance can be easily affected by Chinese Word Segmentation (CWS) and Part-of-Speech (POS) tagging. Second, it heavily relies on manually feature engineering [14].

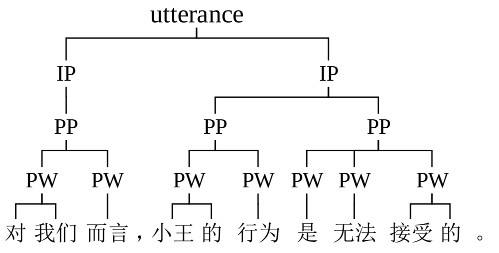


Figure 1: The hierarchical prosodic structure in Chinese.

To overcome the disadvantages of traditional prediction based on CRF, a new architecture based on deep neural networks (DNN) and embedding features has been proposed in previous work [14]. It has proved that stacking feed-forward and bidirectional long short-term memory (BLSTM) recurrent network layers achieves superior performance over the CRF-based method. The embedding features learned from raw text further enhance the performance. Similar conclusion can be drawn from other studies [15]. However, in [14], although character-level features have been investigated while word-level features are always considered indispensable in prosodic structure prediction. Moreover, three separate neural networks were trained independently for PW, PP and IP in a cascaded framework, which is blamed for error accumulation.

Word level-features are usually considered helpful for prosodic structure prediction because word boundaries are always prosodic boundaries. However, manual word segmentation is quite laborious and automatic word segmentation will inevitably cause some errors. The particle size of CWS is difficult to choose. Further, in Chinese dialect synthesis, high-accuracy word segmentation is difficult to realize because it is difficult to prepare a large enough corpus. Even in that case, character-level features can be always correctly extracted. This is because character-level features do not contain boundary information, but it can avoid the negative effect of particle size and inevitable segmentation errors in CWS.

doi: 10.21437/SpeechProsody.2016-14

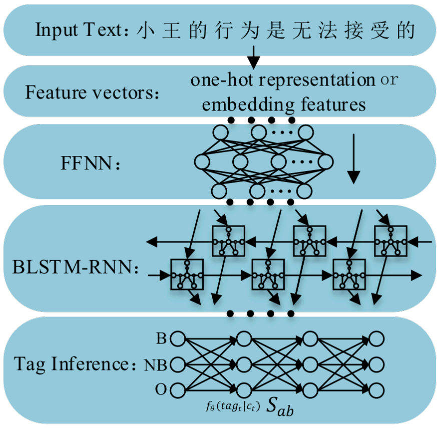

Input Text: 小王的行为是无法接受的
Feature vectors:
one-hot representation or embedding features
FFNN:
BLSTM-RNN:
Tag Inference:
B
NB
O
$f_\theta(tag_t|c_t)$
S_{ab}

Figure 2: The neural network architecture for prosodic boundary prediction. In tag inference, B, NB and O denote boundary, non-boundary and others (e.g., punctuation), respectively.

In this paper, we continue one of the authors' research on prosodic structure prediction with the neural network architecture. This work has two main contributions: (1) N-gram sequence character-level and word-level features are investigated in both one-hot and embedding form to seek the most suitable features for prosodic structure prediction. (2) Instead of traditional cascaded prediction, a unified framework which can be considered as a multi-task learning process is proposed, with expect to eliminate the propagation errors and benefit each target from others' information.

2. Neural network based prediction

To slove the problems in traditional prediction based on CRF, a variant of the neural network architecture [16] for probabilistic language model is proposed in [14]. As shown in Fig. 2, the architecture takes raw text as input and maps each Chinese character into a basic feature vector. The following layers are two types of neural networks, feed-forward neural network (FFNN) and BLSTM recurrent neural network, used to discover multiple levels of feature representations from the basic feature vectors. After network prediction, tag inference is preformed to find an optimum tag transformation path globally.

2.1. Network Structures and Training

A hybrid network structure which includes both feed forward and BLSTM-RNN layers is investigated in work [14]. FFNN, trained with a back-propagation learning algorithm [17], is widely used in many practical applications. In a typical FFNN, every unit in a layer, which is connected to all the units in the previous layer, takes in the output of the previous layer and computes a new set of non-linear activations for next layer. However, the assumption of sample independence results in limited ability of modeling context.

Researchers have proposed RNN to solve the limitation of FFNN. Conventional RNN is only able to make use of previous context information. This is not accurate in modeling prosody that is highly related with both past and future contexts. Instead, bidirectional RNN can access both the preceding and succeeding input contexts with two separate hidden layers, which are then fed to the same output layer. The activation function $\mathcal{H}$ of RNN is usually a sigmoid or hyperbolic tangent function, which often causes the gradient vanishing problem that prevents RNN from modeling the long-span relations in sequence features. An LSTM architecture, which uses purpose-built memory cells to store information, can overcome this problem and model longer contexts. BLSTM-RNN is a combination of LSTM and BRNN.

Deep bidirectional LSTM-RNN can be established by stacking multiple BLSTM-RNN hidden layers on top of each other. The output sequence of one layer is used as input sequence to the next layer. The neural networks can be trained effectively in a layer-wised training manner, which makes it convenient to stack different types of neural network layers on top of each other to form a deep architecture. The deep architecture is able to build up progressively higher level representations of the input data, which is a crucial factor of the recent success of hybrid systems [18].

2.2. Tag inference

To find an optimum tag transformation path globally, tag inference is used to model tag dependency. For input character sequence of a sentence $x_{[1:T]}$ with a tag sequence $tag_{[1:T]}$, a sentence-level score is given by the sum of transition and network scores [19, 20]:

$$l(x_{[1:T]}, tag_{[1:T]}, \theta) = \sum_{t=1}^{T} (S(tag_{t-1}, tag_t) + f_\theta(tag_t|x_t)) \tag{1}$$

where $S(a, b)$ is the transition score from tag a to tag b. a and b belong to a set of tags $G = \{B, NB, O\}$. $f_\theta(tag_t|x_t))$ indicates the score output for tag_t at the t-th character by the network θ. The best tag path $tag^*_{[1:T]}$ can be found by maximizing the sentence score:

$$tag^*_{[1:T]} = \arg\max_{\forall l_{[1:T]}} l(x_{[1:T]}, tag_{[1:T]}, \theta). \tag{2}$$

3. Proposed Approach

3.1. Word-level features vs. character-level features

Word level-features are usually considered helpful for prosodic structure prediction because word boundaries are always prosodic boundaries. However, manual word segmentation is quite laborious and automatic word segmentation will inevitably cause some errors. The particle size of CWS is difficult to choose. In Chinese dialect synthesis, high-accuracy word segmentation is difficult to realize because it is difficult to prepare a large enough corpus. In that case, character-level features can perform better than word-level features. Character-level features do not contain obvious boundary information, but it can avoid the negative effect of particle size and inevitable segmentation errors in CWS. In our work, we investigate both word-level and character-level features. In addition to single Chinese character, a sequence with N characters is also examined, and we call it N-gram sequence. For current character x_t, its N -gram sequence is defined as follows:

$$X = \{x_{t-\frac{N-1}{2}}, \cdots, x_{t-1}, x_t, x_{t+1}, \cdots, x_{t+\frac{N-1}{2}}\} \tag{3}$$

3.2. One-hot features vs. embedding features

Before being fed into network, word-level features or character-level features are transformed into vectors by mapping operation. One-hot and embedding representations are two possible methods. In this work, we would like to compare the results under both kinds of representation. Typically, a character dictionary D of size $|M|$ is extracted from the training set and unknown characters are mapped to a special symbol that is not used elsewhere. The character set in D are represented as $D = (d_1, d_2, \cdots, d_M)$. For n-gram sequence X, we can simply use $H(X) = (h_1, h_2, \cdots, h_M)$ as its one-hot representation. h_i is defined as follows:

$$h_i = \begin{cases} 1 & (d_i \in X), \\ 0 & (d_i \notin X). \end{cases} \tag{4}$$

For a word $W = (c_1, c_2, \cdots, c_N)$ which include N characters, its one-hot form representation $H(W) = (h_1, h_2, \cdots, h_M)$ is defined as (4).

However, one-hot representation is blamed for high dimensions, and it fails to model the semantic similarity between the ideographic characters. In contrast, the distributed representation or embedding feature, the form of a low dimensional continuous-valued vector learned from raw text in a fully unsupervised manner using neural networks, has been experimentally proved to carry important syntactic and semantic information [21]. In [14], in order to prepare embedding features, a skip-gram model *word2vec* which is proposed by Mikolov et al. [22] is chosen. As preliminary experiments did not show much difference of performance among various embedding features, we use *word2vec* in this study.

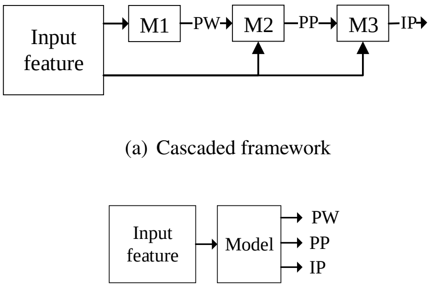
3.3. Unified framework vs. Cascaded framework

In traditional prosody prediction task, prosodic structure is predicted in a cascaded form. PW boundary is firstly predicted according to automatically detected word boundaries. Then, the predicted boundary of PW is used as given information for PP boundary prediction. The result of PP boundary prediction is used for IP boundary prediction. As is shown in Fig. 3(a), such kind of cascading structure can ensure prosodic boundaries consistence, where prosodic hierarchy is guaranteed. That is to say, the predicted boundaries of IP would always be the predicted boundaries of PP, and the predicted boundaries of PP would always be the predicted boundaries of PW. However, the errors that occurred in PW prediction could affect the accuracy of PP prediction, and even IP prediction. Moreover, different input and output vectors should be prepared separately, training three deep network is time-consuming.

In order to avoid or propagation error, we test a unified prediction framework, which is showed in Fig. 3. The network used for unified prediction shares the same input vectors but generate different output forms compared with cascaded structure. In unified framework, PW, PP and IP are transformed into a one vector and predicted at the same time by only one network, which can avoid exploiting specific features carefully optimized for each task and eliminate error propagation. This unified training process can also be considered as multi-task learning. Learning multiple related tasks simultaneously has been empirically as well as theoretically shown to often improve performance significantly compared with learning each task independently [23]. We hope the knowledge of different tasks can benefit each other.

4. Experimental Setup

Our database includes 48,210 Chinese sentences. 90% utterances are used for training, 5% are used for validation, the others are used for testing. Prosodic boundaries of all the sentences are manually labelled by experts. Word segmentation and POS tagging is performed by a software provided by Language Technology Platform (LTP) [24]. The accuracy of word segmentation is 97% and the accuracy of POS tagging is 96%. A dictionary with 4,030 characters is extracted from the training dataset. To prepare embedding features, embedding models are trained with a large set of raw texts which are collected from People's Daily. All texts are preprocessed with text normalization. To perform tag inference, statistical jumping scores between three different boundary tags are estimated from training data.



(a) Cascaded framework

(b) Unified framework

Figure 3: Framework of prosodic structure prediction. M represents for neural network model

We have performed prosodic prediction in a cascaded framework and in our unified framework respectively. For each framework, we prepared four groups of features to test. The four groups of features are listed as follows:

(1) n-gram character-level one-hot features

(2) word-level one-hot features

(3) n-gram character-level embedding features

(4) word-level embedding features

For cascaded framework, three separate neural network models were trained independently for PW, PP and IP. For one-hot features, the network inputs for PW prediction are 4300-dimensional feature vectors, and the inputs for PP or IP prediction have 4031 dimensions. For embedding features, different sizes (256M, 512M, 1024M and 2048M text) of data for unsupervised training and different dimensions (128, 256, 512, 1024) of features are tested. The network outputs have three dimensions and each corresponds to one of three boundary tags: B, BN and O. B for a boundary, NB for non-boundary, and O for other symbols such as punctuation.

For unified prediction, PW, PPH and IPH are predicted by the same and integrated network. Input vectors are the same with the network used for PW prediction in the cascaded framework. Output vectors have nine dimensions which include boundary tags for PW, PP and IP.

Implementation of the network training is done with the help of a machine learning library "CURRENNT" [25]. To find out the best network structure for prediction, we have tested several different distributions of nodes' size and layer components. The momentum of training is set to 0.9. Learning rate for PW prediction is set to 1e-3, for PP and IP prediction it is set to 1e-4 in the cascaded framework. And the learning rate used in unified framework is set as 1e-5. The maximum number of iteration is set to 300 and training process is stopped if no improvement observed within the latest 20 iterations. A softmax output layer is used in the network and statistical scores of boundary

Table 1: F-score (%) of CRF-based prosody prediction

Boundary	PW	PP	IP
F-score	95.72	80.60	76.15

Table 2: F-score (%) of N-gram character-level one-hot features

N-gram	N=1	N=3	N=5	N=7	N=9	N=11
PW	**95.40**	94.90	94.65	94.32	94.15	94.11
PP	**83.71**	83.71	82.79	81.93	81.34	81.21
IP	**81.21**	80.90	80.37	79.95	79.66	79.22

tags can be predicted by the trained network. Then tags inference is performed and an optimum transformation path of all tags in one sequence is searched globally by using Viterbi algorithm.

CRF approach is also used for comparison. It is performed with the help of CRF++ toolkit [26].

5. Evaluations and Analysis

F-score is used as evaluation criteria. In this paper, we only reports the evaluation results with the optimum network structure that was experimentally determined. For embedding features, only results with optimum trained embedding models and feature dimensions are listed.

Table 1 shows the evaluation results of CRF prediction. Table 3 shows the evaluation results of cascaded prediction with four different groups of features. From Table 1 and Table 3 we can see that for all methods, PW owns rather higher F-score than PP and IP. But neural network based approaches can predict prosodic boundaries more accurately than CRF, especially in terms of PP and IP. For n-gram character-level one-hot features, prediction accuracy decreases while the length of sequence increases, and 1-gram achieves highest evaluations. Due to this reason, in other experiments, we only use 1-gram character-level features. Compared with 1-gram character-level one-hot features, word-level features with one-hot representation has higher F-score in terms of PW prediction, but a little worse in terms of PP and IP prediction. According to Table 3, we can find similar phenomenon. Word-level embedding features has higher F-score than character-level embedding features for PW prediction, but lower F-score for PP and IP prediction. Embedding representation outperform one-hot representation in terms of both character-level and word-level features. Among all these experiment mentioned above, word-level embedding features show best performance for PW prediction, and character-level embedding features obtain highest F-score in terms of PP and IP prediction.

We choose character-level and word-level embedding features, which have shown best performance in cascaded experiments, to perform prosodic structure prediction in our unified framework. The results is showed in Table 4. For both character-level and word-level embedding features, unified framework perform better in PW prediction, and F-score of PP and IP prediction is a little lower but quite approximate to cascaded framework.

Table 3: F-score (%) of cascaded prediction. C represents for character-level features, W represents for word-level features

	PW	PP	IP
C one-hot	95.40	83.71	81.21
W one-hot	95.73	83.28	80.99
C embedding	96.04	**84.60**	**81.78**
W embedding	**96.34**	84.31	81.32

Table 4: F-score (%) of unified prediction. C represents for character-level features, W represents for word-level features

	PW	PP	IP
C embedding	96.20	**84.38**	**81.43**
W embedding	**96.54**	84.18	81.04

6. Conclusions

In this paper, we have investigated BLSTM-RNN-based prosodic prediction with different groups of features in both cascaded and unified framework. Experimental results show that embedding features outperform one-hot features in all cases. Word-level features achieve similar evaluations to character-level features. Although unified prediction has achieved similar evaluation to cascaded framework, its performance is not as excellent as we expected. However, unified framework requires less computational work and without the necessity of preparing task specific features. And multi-task learning has already been both experimentally and theoretically proved to be capable of outperforming separate task learning. We will go on our study and try to improve performance.

7. References

[1] Wikipedia, "Prosody (linguistics)," https://en.wikipedia.org/wiki/Prosody_(linguistics).

[2] J. Sun, J. Yang, J. Zhang, and Y. Yan, "Chinese prosody structure prediction based on conditional random fields," in *Proceedings of ICNC*, vol. 3, 2009, pp. 602–606.

[3] F. chiang Chou, C. yu Tseng, and L. shan Lee, "Automatic generation of prosodic structure for high quality mandarin speech synthesis," in *Proceedings of ICSLP*, vol. 3. IEEE, 1996, pp. 1624–1627.

[4] J. H. Jeon and Y. Liu, "Automatic prosodic events detection using syllable-based acoustic and syntactic features," in *Proceedings of ICASSP*, 2009, pp. 4565–4568.

[5] V. Rangarajan, S. Narayanan, and S. Bangalore, "Exploiting acoustic and syntactic features for prosody labeling in a maximum entropy framework," in *Proceedings of NAACL HLT*, 2007, pp. 1–8.

[6] P. Koehn, S. Abney, J. Hirschberg, and M. Collins, "Improving intonational phrasing with syntactic information," in *Proceedings of ICASSP*, 2000, pp. 1289–1290.

[7] M. Chu and Y. Qian, "Locating boundaries for prosodic constituents in unrestricted mandarin texts," *Computational linguistics and Chinese language processing*, pp. 61–82, 2001.

[8] X. Nie and Z.-y. Wang, "Automatic phrase break prediction in chinese sentences," *Journal of Chinese information Processing*, pp. 39–44, 2003.

[9] J.-F. Li, G. Hu, and R.-h. Wang, "Chinese prosody phrase break prediction based on maximum entropy model," in *Proceedings of INTERSPEECH*, 2004, pp. 729–732.

[10] G.-A. Levow, "Automatic prosodic labeling with conditional random fields and rich acoustic features," in *Proceedings of IJCNLP*, 2008, pp. 217–224.

[11] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of 18th ICML*, 2001, pp. 282–289.

[12] M. Nobuaki, K. Shumpei, S. Shinya, and H. Keikichi, "Improved prediction of japanese word accent sandhi using crf," in *Proceedings of INTERSPEECH*, vol. 10, no. 38,900, 2012, pp. 114–783.

[13] Y. Qian, Z. Wu, X. Ma, and F. Soong, "Automatic prosody prediction and detection with conditional random field (crf) models," in *Proceedings of ISCSLP*, 2010, pp. 135–138.

[14] 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 *Proceedings of ASRU*. IEEE, 2015.

[15] A. Rosenberg, R. Fernandez, and B. Ramabhadran, "Modeling phrasing and prominence using deep recurrent learning," in *Proceedings of INTERSPEECH*, 2015.

[16] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *The Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, 2003.

[17] S.-i. Horikawa, T. Furuhashi, and Y. Uchikawa, "On fuzzy modeling using fuzzy neural networks with the back-propagation algorithm," *IEEE transactions on Neural Networks*, vol. 3, no. 5, pp. 801–806, 1992.

[18] A. Graves, N. Jaitly, and A.-R. Mohamed, "Hybrid speech recognition with deep bidirectional lstm," in *Proceedings of ASRU*, 2013, pp. 273–278.

[19] X. Zheng, H. Chen, and T. Xu, "Deep learning for chinese word segmentation and pos tagging." in *Proceedings of EMNLP*, 2013, pp. 647–657.

[20] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *The Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.

[21] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin, "Learning sentiment-specific word embedding for twitter sentiment classification," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, vol. 1, 2014, pp. 1555–1565.

[22] Y. Goldberg and O. Levy, "word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method," *arXiv preprint arXiv:1402.3722*, 2014.

[23] A. Evgeniou and M. Pontil, "Multi-task feature learning," *Advances in neural information processing systems*, vol. 19, p. 41, 2007.

[24] W. Che, Z. Li, and T. Liu, "Ltp: A chinese language technology platform," in *Proceedings of Coling 2010:Demonstrations*. ACL, 2010, pp. 13–16.

[25] F. Weninger, "Introducing currennt: The munich open-source cuda recurrent neural network toolkit," *Journal of Machine Learning Research*, vol. 16, pp. 547–551, 2015.

[26] T. Kudo, "Crf++: Yet another crf toolkit," *Software available at http://crfpp. sourceforge. net*, 2005.