Chinese Syllable-to-Character Conversion with Recurrent Neural Network based Supervised Sequence Labelling

Yi Liu and Jing Hua and Xiangang Li and Tong Fu and Xihong Wu
Peking University, Beijing, China
E-mail: {liuy, huajing, lixg, fut, wxh}@cis.edu.cn

***Abstract*—Chinese Syllable-to-Character (S2C) conversion is the important component for Input Methods, and the key problem in Chinese S2C conversion is the serious phenomenon in Chinese language. In order to disambiguate homophones to improve Chinese S2C conversion, in this paper, Chinese S2C conversion is treated as a sequence labelling task, and the recurrent neural network (RNN) based on supervise sequence labelling is introduced to achieve the direct conversion from syllable sequences to word sequences. Through the direct conversion with the proposed RNN, the cascade error in multi-pass approaches can be eliminated effectively. Experimental results indicate that, in second pass decoding, the re-ranking with RNN language model has better performance than N-gram language model in both perplexity and S2C conversion accuracy. Moreover, the direct S2C conversion with RNN can improve the accuracy from 93.77% (RNN language model) to 94.17%.**

I. Introduction

Chinese Syllable-to-Character (S2C) conversion is the task of mapping the input Chinese syllable sequence to the Chinese character/word sequence, and is one of the important components in both Chinese speech input and text Input Methods. Specifically for Chinese Input Method, because the pronunciations encoded with Chinese syllables is close to spoken speech and is more manageable for users, the Chinese S2C conversion is more widely used in Chinese Input Method than others, such as Wubi font and Zheng Ma Input Method.

The main problem in Chinese S2C conversion task is that there are many homophones in Chinese, which is the inherent phenomenon for Chinese language. There are only 410 unique toneless syllables in the 8105 characters in "General standard Chinese characters" (2013). Obviously, the amount of Chinese characters are more than the amount of Chinese toneless syllables, which results in serious homophones problem for Chinese S2C conversion. Besides, according to the statistics of homophones from the "Modern Chinese Dictionary" (2008), the proportion of toneless homophone characters is 99.28% and the proportion of tone homophone characters is more than 90% [1]. Moreover, the statistic result also shows a decreasing in homophone phenomenon with the word length increasing. For example, the proportion of toneless homophone words with 4 characters drops to 1.26% rapidly. This inspires that the long context information is useful for disambiguating homophones in Chinese S2C conversion task.

Some approaches are proposed to disambiguate homophones in Chinese S2C conversion task with more linguistic and context information. These approaches can be classified into three categories: rule-based methods (e.g. [2], [4], [5]); statistic-based methods (e.g. [6], [7], [11]) and combination of both (e.g. [12], [15]). The expansibility of the rule-based approaches is poor, and with the increasing of rules, the collision and confusion of rules are more serious. Currently most Chinese S2C conversion systems are based on statistical methods and the N-gram language models (LM) are widely used as the basic statistical methods. However, because of the limited context in N-gram LM (usually $3-5$) and the data sparse problem in training process, the ability of sequence modeling in N-gram is limited. Besides, although the rule-based and statistic-based approaches can be combined to integrate more linguistic information, this combination needs more tagged training data which aggravates the difficulty of data acquisition and the complexity of training process.

In fact, the Chinese S2C conversion is an inherently dynamic process, thus the recurrent neural networks (RNNs) can be considered as the alternative models. The cyclic connections in RNNs exploit a self-learnt amount of temporal context, which makes RNNs able to incorporate context information in a flexible way and better suit for sequence modeling tasks. Literatures [13] have indicated that The RNN based LMs achieve better performance and give better context representation than N-gram. However, the RNN LMs are used mostly in second pass or re-ranking process, the cascade error effects the performance of Chinese S2C conversion seriously. The Chinese S2C conversion can be treated as the task of labelling unsegmented sequence data. For labelling unsegmented sequence data with RNNs, the Connectionist Temporal Classification (CTC) algorithm have been successfully used in many tasks, such as the handwriting recognition [9], automatic speech recognition [14], [8] and grapheme-to-phoneme conversion [10]. In this paper, a RNN is trained based on segmented and aligned syllable-character sequence, in which the blank unit proposed in CTC algorithm is used. Then the decoding algorithm is given to achieve the direct conversion from input syllable sequences to output character sequences with the RNNs based supervised sequence labelling. Experiments are conducted on People's Daily of China in 2000, in which, two RNN based approaches are applied: one is the RNN LM, and the other is the RNN based direct conversion. The experimental results reveal that the performance can be improved through

introducing RNNs.

The paper is organized as follows. First, an overview of RNN LM is described in Section 2. Section3, the RNN structure for direct S2C conversion, the training and decoding method. Section 4 presents the experiments. Finally, the paper concludes with a discussion regarding the experiments and detailing directions of future work.

II. Recurrent Neural Network Language Model

In RNN LM, the RNN is used as a generative model over the word sequence, where the output of RNN is next word unit in the sequence [13]. Figure 1 shows the architecture of the class-based RNN LM.

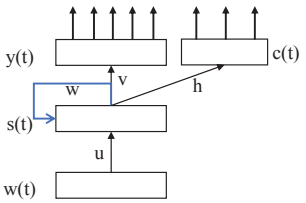

Fig. 1. Architecture of the class-based RNN LM.

The network has an input layer x, hidden layer s and output layer y. Input to the network in time t is vector $x(t)$ which is formed by concatenating vector w representing current word,and output from neurons in context layer s at time $t-1$. $s(t)$ is state of the network (hidden layer), and output is denoted as $y(t)$ which is the probabilities of the predict words for time $t+1$. In addition, the $c(t)$ in the output layer denotes the probabilities of the predict word class for time $t+1$. Input, hidden and output layers are then computed as follows:

$$\begin{aligned} x(t) &= w(t) + s(t-1) \\ s_j(t) &= f(\sum_i (x_i(t)u_{ij}) + \sum_l (s_l(t-1)w_{jl})) \\ y_k(t) &= g(\sum_j (s_j(t)v_{cj})) \\ c_l(t) &= g(\sum_j (s_j(t)h_{lj})) \end{aligned} \tag{1}$$

where $f(z)$ is sigmoid activation function

$$f(z) = \frac{1}{1+e^z} \tag{2}$$

and $g(z)$ is softmax function:

$$g(z_m) = \frac{e^{z_m}}{\sum_k (e^{z_k})} \tag{3}$$

Softmax ensures that this probability distribution is valid, i.e. $y_m m(t) > 0$ for any word m and $\sum_k (y_k(t)) = 1$. At each training step, error vector is computed according to cross entropy criterion and weights are updated with the standard back-propagation algorithm:

$$error(t) = desired(t) - y(t) \tag{4}$$

where desired is a vector using 1-of-N coding representing the word that should have been predicted in a particular context and $y(t)$ is the actual output from the network. Then the RNN LM can be trained by back-propagation through time (BPTT) algorithm in [3].

III. S2C Conversion with RNN Based Supervised Sequence Labelling

The simplest RNN for direct conversion is based on characters, i.e. the input is the current syllable and the output is the predict character. However, this simplest RNN loses the word information and does not perform well.

If the RNN is based on word, ie. at each time, the input is the multi-syllables (e.g. “bei jing”) and the output is the corresponding word (e.g. “ 北京”), there are many alternative segmentations for a syllable sequence and for each segmentation the RNN is used to decode to get the result sequence. In such condition, The computational cost in decoding process for each candidate segmented syllable sequence could be very expensive.

However, in CTC algorithms [14], through introducing the blank units, the multiple candidate segmentation of input sequence in decoding phase is avoided, which makes the decoding quite effective. In this paper, in order to perform direct S2C conversion, the blank unit from CTC algorithms is used as the output label of those syllables not the last one in a Chinese word, and the RNN is trained to distinguish the word from input syllable sequence. This RNN structure for sequence labelling in this section is different from the structure in figure 1, i.e. the input vector at the input layer represents the current syllable, and the output symbol of the output layer is the predict word or not. The hidden layer of this RNN represents all previous syllable history, thus the model can theoretically represent long context patterns. Fortunately, This RNN can also be trained by BPTT algorithm. Next subsections will give the details about the training and decoding algorithm.

A. The Training for Direct Conversion RNN

In order to train the RNN for direct conversion, a blank (NULL) unit is introduced in the output layer: $L' = L \bigcup blank$. Thus, the number of units in output layer is $|L|+1$.

In this work, the units in output layer represent the words in vocabulary or blank. When the RNN outputs blank symbol, it means the current input syllable locates on the beginning or middle of the word. Figure 2 gives the example of training process with blank unit. The input unsegmented syllable sequence is “bei jing da xue” (Peking University). The correct output sequence is “北京/大学”, tagged by red.

Unlike the CTC algorithms, for Chinese S2C conversion, the training process is supervised. Because of the output word sequences are segmented, the output of RNN for current syllable is unique, i.e. word or blank. The training for direct conversion RNN do not need to compute the forward and backward variables as mentioned in [14].

When the blank unit is introduced, the output of the RNN is related to the history output symbols. Therefore, the softmax

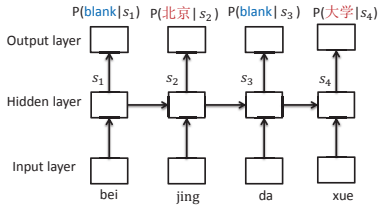

Fig. 2. The example of RNN training process with blank.

operation is not operated over all the units of the output layer when training the RNN. Because of the relationship of the output in the output layer at different time, words in vocabulary can be classified by the pronunciations explicitly. Only the units of the output layer corresponding to the same pronunciations can do the softmax operation together. That is to say, the sofmax operation is done locally on output layer. Then, the RNN can be trained by BPTT algorithm.

B. The Decoding for Direct Conversion RNN

For the input syllable sequence $\mathbf{x}$ of length T, let $\mathbf{y} = N_w(\mathbf{x})$, be the sequence of network outputs, y_k^t denotes the probability of observing label k at time t, π presents the candidate path in L'^T, the set of all candidate paths. Then, the probability of path π is:

$$p(\pi|\mathbf{x}) = \prod_{t=1}^{T} y_{\pi_t}^t, \forall \pi \in L'^T \tag{5}$$

the most probable word sequence $\mathbf{l}^*$ for $\mathbf{x}$ can be calculated:

$$\begin{aligned} \mathbf{l}^* &= F(\pi^*) \\ \pi^* &= arg \max_{\pi \in N^t} p(\pi|x) \end{aligned} \tag{6}$$

where, $F : L'^T \mapsto L^{\leq T}$, the mapping from output symbol sequence of network to word/label sequence. In decoding process, a forward matching or prefix search decoding is applied to find the most probable labelling. Figure 3 gives an example for decoding "bei jing da xue" with RNN trained last subsection. For each input syllable, the units in output layer activated by the current input syllable are listed. The best result sequence of output layer is tagged in red. The computation for probabilities and candidate path is as follows.

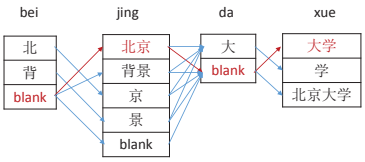

Fig. 3. The example of RNN decoding process with blank.

At time t, the output of RNN can be some word in vocabulary or blank. Then, for the set $Y = \pi \in L'^T : pre = F(\pi_{1:t-1})$, there are two kinds of variables: $\gamma_t(pre, \pi_t \neq blank)$ and $\gamma_t(pre, \pi_t = blank)$, which can be calculated recursively.

Therefore, the probability of l is the sum of the total probability of π with and without the final blank at time T:

$$p(l|x) = \gamma_T(pre, \pi_T \neq blank) + \gamma_T(pre, \pi_t = blank) \tag{7}$$

The details of decoding process is given in Algorithm 1.

Algorithm 1 Framework of decoding with direct conversion RNN.

Initialization:

$$1 \leq t \leq T = \begin{cases} \gamma_t(\phi, \pi_t \neq blank) = 0 \\ \gamma_t(\phi, \pi_t = blank) = \prod_{t'=1}^{t} y_b^{t'} \\ \prod_t = \phi \end{cases}$$

$p(\phi|x) = \gamma_T(\phi, \pi_T = blank)$
$pre^* = \phi$
$flag = false$

Algorithm:

for $t = 1$ to T **do**
  for all all labells $i \in L$ **do**
    $n = |i|$
    $pre^* = argmax_{pre}(\gamma_{t-n}(pre, \pi_{t-n} \neq blank))$
    Where $(pre, \pi_{t-n} \neq blank) \in \Pi_{t-n}$
    $pre = pre^* + i$
    $\gamma_t(pre, \pi_t \neq blank) = \gamma_{t-n}(pre^*, \pi_{t-n} \neq blank) + \sum_{t'=t-n+1}^{t-1}(y_b^{t'}) + y_i^t$
    for all all paths $\pi \in \Pi_t$ **do**
      if $\pi == (pre, \pi_{t-n} \neq blank)$ **then**
        $\gamma_t(\phi, \pi) += \gamma_t(pre, \pi_t \neq blank)$
        $flag = true$
        break
      end if
    end for
    if $flag == false$ **then**
      add $(pre, \pi_{t-n} \neq blank)$ into Π_t
    else
      $flag = false$
    end if
  end for
  $sort(\gamma_t(pre, \pi_t \neq blank), beam)$
end for

IV. Experiments and Results

A. Experimental Conditions

Our experiments are conducted on the corpus of the People's Daily of China in 2000, which is randomly divided into training set, development set and test set, as shown in Table I. A N-gram word segment operation is applied to the corpus, with use of Google N-gram LM trained on the "Chinese Web 5-gram Version 1"(LDC2010T06). The vocabulary has $115,527$ terms with 3 parts: (1) $9,097$ Chinese characters; (2) $106,429$ multi-character words in training set; (3) 1 Out-of-Vocabulary (OOV) tag. There are $116,494$ terms in the dictionary including: 417 single syllables; $88,105$ multi-syllables; 1 for OOV.

For baseline system, a modified Kneser-Ney smoothed 3-gram LM and 5-gram LM are trained and denoted as KN3 and KN5 respectively.

TABLE I
THE PARTITIONS OF PEOPLE'S DAILY OF CHINA IN 2000

Data Set	Sent Account	Word Account	Character Account
Training Set	232, 397	12, 049, 511	20, 933, 240
Development Set	4, 998	263, 479	460, 314
Test Set	12, 494	659, 371	1, 144, 983

For RNN LM, there are $115,527$ units in input layer. For hidden layer, the number of units is from 100 to 800 and when the size is 500, the performance on development set is optimal.

For the direct conversion RNN, the sizes of input layer, hidden layer and output layer are 418, 500 and $116,495$ respectively. In the output layer, the vocabulary is classified into $88,523$ classes according to the homophones.

B. Experimental Results

First of all, this section calculates the perplexity on test set for different LMs. As shown in Table II, RNN LM has the lowest perplexity than the baseline KN3 and KN5 LMs.

TABLE II
THE TEST SET PERPLEXITY

LM	Perplexity
KN3	433.04
KN5	360.40
RNN LM	299.26

Then, the experimental results for S2C conversion on test set are given in Table III. For the second pass decoding, the experiments generate 100-best lists using KN3, and then re-rank this 100-best lists with KN5 and RNN LM respectively. Note that the RNN LM on second pass yields 0.84% absolute improvement in accuracy over KN3 and 0.20% absolute improvement over KN5 re-ranking, due to the RNN LM can get better represent of history context with lower data sparse problem.

Moreover, the last line in Table III shows that the proposed direct conversion RNN can improve the accuracy to 94.17%, an 0.40% absolute improvement over RNN LM. On the one hand, because the direct conversion with the proposed RNN based on sequence labelling can convert the syllable sequence to character sequence directly without second pass or re-rank operation, the cascade error can be eliminated effectively, the direct conversion with the proposed RNN outperforms the re-ranking with RNN LM. On the other hand, there is no need to do multi-pass decoding with use of the proposed direct conversion RNN, which can improve the decoding speed compared to multi-pass decoding with RNN LM.

TABLE III
CHINESE S2C CONVERSION ACCURACY ON TEST SET (%)

First-pass	Second-pass	Accuracy	Absolute Imp.
KN3	-	92.94	-
KN3	KN5	93.57	0.63
KN3	RNN LM	93.77	0.83
Direct Conversion RNN	-	94.17	1.23

V. CONCLUSIONS AND FUTURE WORK

In this paper, a RNN direct Chinese S2C conversion is realized. The architecture of the direct conversion RNN is introduced. Then the training process and decoding algorithm are presented in this paper. The experimental results show that the proposed approach perform better than N-gram and RNN LM in multi-pass decoding strategy. On the one hand, there is no cascade error in the proposed approach. On the other hand, the RNN can represent the long context information without data sparsity.

For the future work, more language information can be introduced in this framework. Besides, the training process and decoding algorithm also can be applied to other kinds of RNN.

ACKNOWLEDGMENT

The work is supported in part by National Basic Research Program (973 Program) of China (No. 2013CB329304), the National Natural Science Foundation of China (No. 90920302, No. 91120001, No.61121002), the "Twelfth Five-Year" National Science & Technology Support Program of China (No. 2012BAI12B01) and the Key Program of National Social Science Foundation of China (No. 12&ZD119).

REFERENCES

[1] Y. Liu, "Research on pronunciation dictionary for Chinese speech recognition", Peking University, 2014.
[2] S. Wan, H. Saiton and K. Mori, "Experiment on pinyin-hanzi conversion Chinese word processor", Computer Processing of Chinese and Oriental Languages, 1984, 1(4): pp.213 – 224.
[3] D.E. Rumelhart, G.E. Hinton, R.J. Williams, "Learning representations by back-propagating errors", Nature, 1986, 323, pp. 533-536.
[4] S. Chen, C. Chang, J. Kuo and M. Hsieh, "The continuous conversion algorithm of Chinese character's phonetic symbols to Chinese character", Proceedings of National Computer Symposium, 1987, pp.437 – 442.
[5] M. Lin and W. Tsai, "Removing the ambiguity of phonetic Chinese input by the relaxation technique", Computer Processing of Chinese and Oriental Languages, 1987, 3(1): pp. 1 – 24.
[6] R. Sproat, "An application of statistical optimization with dynamic programming to phonemic-input-to-character conversion for Chinese" Proceedings of ROCLING III, 1990, pp. 379 – 390.
[7] J. Guo, "Statistical language modelling and some experimental results on Chinese Syllable-to-Words transcription", Journal of Chinese Information Processing, 1993, 7(1): pp. 18-27.
[8] H. Sak, A. Senior, K. Rao, O. Irsoy, A. Graves, F. Beaufays and J. Schalkwyk, "Learning acoustic frame labeling for speech recognition with recurrent neural networks", ICASSP, 2015.
[9] A. Graves, M. Liwichi, S. Fernández et al., "A novel connnectionist system for unconstrained handwriting recognition", IEEE Transaction on Pattern Analysis and Machine Intelligence, 2009, vol. 31, pp. 855 – 868.
[10] K. Rao, F. Peng, H. Sak, F. Beaufays, "Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks", ICASSP, 2015.
[11] J.H. Xiao, B.Q. Liu and X.L. Wang, "A Self-adaptive Lexicon Construction Algorithm for Chinese Language Modeling", Acta Automatica Sinica, 2008, 34(1): pp. 40-47.
[12] Huang, D. Powers, "Adaptive compression-based approach for Chinese pinyininput", Association for Computational Liguistics, 2004.
[13] T. Mikolov, M. Karafiát, L. Burget, J. Černocký and S. Khudanpur, "Recurrent neural network based language model", INTERSPEECH, 2010, pp. 1045-1048.
[14] A. Graves, S. Fernández, F. Gomez and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks", ICML, 2006, pp. 369-376.
[15] Y.F. Peng, "Research on the Variable Dependency Class Language Model", Peking Unversity, 2010.