

# Exploiting Bishun to Predict the Pronunciation of Chinese

Chenggang Mi<sup>1,2</sup>, Yating Yang<sup>1,2,3</sup>, Xi Zhou<sup>1,2</sup>, Lei Wang<sup>1,2</sup>, Xiao Li<sup>1,2</sup>, Tonghai Jiang<sup>1,2</sup>

<sup>1</sup> Xinjiang Technical Institute of Physics & Chemistry, Chinese Academy of Science, China

<sup>2</sup> Xinjiang Key Laboratory of Minority Speech and Language Information Processing, China

<sup>3</sup> Institute of Acoustics, Chinese Academy of Sciences, China

{micg,yangyt,zhouxi,wanglei,xiaoli,jth}@ms.xjb.ac.cn

**Abstract.** Learning to pronounce Chinese characters is usually considered as a very hard part to foreigners to study Chinese. At beginning, Chinese learners must bear in mind thousands of Chinese characters, including their pronunciation, meanings, Bishun (order of strokes) etc., which is very time consuming and boring. In this paper, we proposed a novel method based on translation model to predict the Chinese character pronunciation automatically. We first convert each Chinese character into Bishun, then, we train the pronunciation prediction model (translation model) according to Bishun and their correspondence Pinyin sequences. To make our model practically, we also introduced some error tolerant strategies. Experimental results show that our method can predict the pronunciation of Chinese characters effectively.

**Keywords.** Pronunciation prediction, Bishun, language model, translation model, error tolerant.

## 1 Introduction

Chinese characters are logograms used in the writing of Chinese and some other Asian languages. In Standard Chinese, they are called Hanzi (simplified Chinese: 汉字)<sup>1</sup>. They have been adapted to write a number of other languages including Japanese, Vietnamese, and et.al. Modern Chinese has many homophones; thus the same spoken syllable may be represented by many characters, depending on meaning. Cognates in the several varieties of Chinese are

generally written with the same character. They typically have similar meaning, but often quite different pronunciations [1].

For beginners, the very first step to learn a Chinese character is to learn its pronunciation and Bishun (order of strokes). When he wants to learn the pronunciation of Chinese characters which are very complex (“饗”, “霹”, “犇”) or sharing the same spoken syllable with others (“阿(a)姨”, “东阿(e)”, “西藏(zang)”, “储藏(cang)”), he may feel very puzzled.

If you are a native English speaker, and you know nothing about languages (which written in Latin alphabet) like French, you can also read some French words [2]. Why? That's because English and French share most characters, even some words, and the same Latin character usually has the same/similar pronunciation. For example, given some French words “président”, “restaurant”, “piano”, “gouvernement”, “expert”, “fleur”, we can transfer these words into phonetic symbols by English and French transformation rules (Table 1), respectively.

In this paper, we try to predict the pronunciation of Chinese characters based on a machine translation model [3, 4, 5]. Here, we follow one principle: for characters, same characters may have same/similar pronunciation; for character sequence, same/similar character sequence may have same/similar pronunciation. In Chinese

<sup>1</sup> [https://en.wikipedia.org/wiki/Chinese\\_characters](https://en.wikipedia.org/wiki/Chinese_characters)

**Table 1.** Examples of English and French pronunciation

En_word	En_pron	Fr_word	Fr_pron
President	[prezɪdənt]	président	[prezidã]
Restaurant	[rɛstrɔ̃nt]	restaurant	[rɛstɔ̃rã]
Piano	[pi'ænəʊ̃]	Piano	[pjano]
Government	[gʌvənmənt]	gouvernement	[guvərnəmã]
Expert	[ekspɜ̃:t]	Expert	[ɛksper, -t̪]
Flower	[flaʊ̃ə(r)]	Fleur	[flœ̃r]

learning, we first convert one single Chinese character into a Bihua sequence, and consider the pronunciation prediction as a machine translation problem. We introduced two important features for the pronunciation prediction model: (1) according to local Bishun language modeling, we obtain the local language model score (LLM score); (2) for polyphones in Chinese, we consider global language model score (GLM score) between Chinese characters, also based on Bishun. Moreover, some error tolerant strategies also proposed to make the model more practically.

Section 2 presents some previous work related to our topic. The details of Chinese pronunciation prediction model are described in section 3. We evaluate our model in several experiment sets in section 4. In section 5, we give our conclusions and future directions.

## 2 Related Work

In the field of Chinese pronunciation prediction, [6] proposed a system to foreigners to speak a language they don't know it, this system use a phonetic spelling in the foreigner's own orthography to represent the input text, [7] presented a generative model based on existing dialect pronunciation data plus medieval rime books to discover phonological patterns that exist in multiple dialects. Our work mainly inspired by the framework of statistical machine translation [3, 4]; parameters of the pronunciation prediction model are derived from the analysis of "bilingual" (which means Bishun, Pinyin in our paper) corpus [8, 9].

To the best of our knowledge, this work is the first time that exploiting the orthography of Chinese to predict its pronunciation.

## 3 Methodology

In this section, we first describe the representation of Chinese characters in our model; then, we present two language model features used in our model and details of the translation model based Chinese pronunciation prediction approach; finally, we introduce some error tolerant methods to optimize the pronunciation prediction model.

### 3.1 Representation of Chinese Characters by Bishun

For convenience, we always wrote Chinese words as Pinyin sequence in English research papers. In our method, we introduce another form to represent Chinese characters: Bishun. Bishun, also known as the strokes order (how to write a Chinese character). There are five basic strokes (Bihua, CJKV character) in Chinese, as present in Table 2.

Here, we first convert each Chinese character into Bishun. For convenience, we assign each stroke (Table 2) a number (1-5). Accordingly, Chinese characters can be represented as shown in Table 3.

**Table 2.** Five basic strokes in Chinese characters<sup>2</sup>

n	In Chinese	stroke	In English
1	点/捺	、	Dot/Right-Falling
2	横	—	Horizontal
3	竖		Vertical
4	撇	＼	Left-Falling
5	折	フ	Turning

**Table 3.** Represent Chinese as Bishun

Chn	Bishun (strokes)	Bishun
木	—   ＼ 、	2341
人	＼ 、	41
中	—   —	32323
复	＼ —   —   — —   フ	42323222451
	、	

### 3.2 Translation Model-based Chinese Pronunciation Prediction Model

With method presented above, we can transfer Chinese characters into Bishuns, and consider the Chinese character pronunciation prediction as a machine translation problem.

Statistical machine translation (SMT for short) is a machine translation paradigm where translation results are generated on the basis of statistical translation models whose parameters are derived from the statistical analysis of parallel data. We use the phrase-based translation model as the baseline in our pronunciation prediction model.

In this part, we define the phrase-based translation model formally. The phrase-based translation model is based on the noisy channel model. To train the prediction model, we reformulate the translation probability for translating a Bihua sequence  $f_B$  into pinyin sequence  $e_P$  according to Bayes rule [3, 4] as:

<sup>2</sup> There are also some other strokes in Chinese characters, like “→”(break), and et al., . According to the format of corpus used in our experiments, we only consider five basic strokes.

$$\arg \max_{e_P} p(e_P | f_B) = \arg \max_{e_P} p(f_B | e_P) p(e_P) \quad (1)$$

$p(e_P)$  is a language model and  $p(f_B | e_P)$  is a translation model.

#### Word Alignment

The alignment models used in the pronunciation prediction model are trained based on the IBM models [8]. The IBM alignment models are instances of the EM (Expectation-Maximization) algorithm [14, 15, 16]. The parameters of alignment models are guessed then to apply EM algorithm iteratively so as to approach a local maximum of the likelihood of a particular set of sequence pairs (Bihua sequence and Pinyin sequence).

E-step: the translation probabilities within each sequence are computed;

M-step: they are accumulated to global translation probabilities

#### Model Training

Training of the machine translation is starting from extract phrase table and reordering rule table according to word alignment matrix which obtained from the word alignment stage. In our pronunciation prediction task, the alignment points in a sequence pairs are in order. In detail, we collect each phrase pair from the sequence pair using the word alignment because its characters match up consistently.

We define the consistent with a word alignment as follow: for a phrase pair  $(e, f)$  consistent with an alignment matrix WAM [17] [18], if all characters  $f_1, f_2, \dots, f_{n-1}, f_n$  in  $f$  that have alignment points in WAM have these with characters  $e_1, e_2, \dots, e_{n-1}, e_n$  in  $e$  and vice versa.

#### Decoding

The decoding task in statistical machine translation is to find the best scoring translation according resources obtained from model training. In our pronunciation prediction task, the goal of decoding task is to output the best pronunciation of a given Chinese character.

### 3.3 Language Model Features

Statistical language models [10,11] have been designed to assign probabilities to strings of words (or tokens), which are widely used in speech recognition, machine translation, part-of-speech (POS) tagging, intelligent input method and text to speech system. In this paper, we obtain two important language model features to constraint the output of the pronunciation prediction model's decoding stage. We use n-gram language models [12, 13] in our model, which are trained on unlabeled text.

Let  $w_{i1}^L = (w_{i1}, w_{i2}, w_{i3}, \dots, w_{i(L-1)}, w_{iL})$  denote a string of L characters over a fixed vocabulary.

#### Local Language Model Feature

A Chinese character usually consists of several Bihua. So, when training the pronunciation prediction model, we first consider the very basic feature – the pinyin language model of Chinese character itself, called local language model feature (LLM feature for short). According to the definition of language model, we formulate the LLM feature as follows:

Let  $w_{i1}^L = (w_{i1}, w_{i2}, w_{i3}, \dots, w_{i(L-1)}, w_{iL})$  denotes a string of L characters over a fixed alphabet.  $w_{i1}^L$  indicates the pinyin sequence of the  $i$ th Chinese character. The language model probability of  $w_{i1}^L$  is:

$$P(w_{i1}^L) = \sum_{i=1}^L P(w_{ii'} | w_{i1}^{i'-1}). \quad (2)$$

Here, we use 2-gram language model to denote the language model of a Chinese character's pinyin sequence (LLM).

#### Global Language Model Feature

There are many polyphone characters in Chinese. If we only use LLM features described above, it is difficult to distinguish them. To alleviate this problem, we introduce a global language model feature, which means that when predict the pronunciation of a Chinese character, we not only evaluate the score of it, but also consider its context information (global language model feature, GLM feature for short). In this paper, we define the context information of a Chinese

character as the pronunciation of its previous and next n characters. The GLM feature can be formulated as (3):

$$P'(w_{i1}^L) = \sum_{i''=-3}^{-1} P(w_{i''1}^{L_{i''}}) + P(w_{01}^L) + \sum_{i'''=1}^3 P(w_{i'''1}^{L_{i'''}}). \quad (3)$$

Let  $w_{i1}^L = (w_{i1}, w_{i2}, w_{i3}, \dots, w_{i(L-1)}, w_{iL})$  denote a string of L characters over a fixed alphabet. The  $w_{01}^L$  indicates the pinyin of current Chinese character ( $i = 0$ ), we use its previous 3 characters ( $w_{-31}^L, w_{-21}^G, w_{-11}^K$ ) and next 3 characters ( $w_{11}^M, w_{21}^N, w_{31}^O$ ) as context information.

#### LLM + GLM

The optimized language model can be formulated as follows:

$$P_{opt}(w_{01}^L) = mP + nP'(w_{i1}^L). \quad (4)$$

$P(w_{i1}^L)$  is the local language model feature, and  $P'(w_{i1}^L)$  is the global language model feature. The weights  $m$  and  $n$  are obtained according to optimized language model.

To integrate the LLM and GLM into the machine translation model, we optimize the basic formula of the phrase-based model as follows:

$$\begin{aligned} e_{p-best} &= \arg \max_{e_p} p(f_B | e_p) p(e_p) \\ &= \arg \max_{e_p} p(f_B | e_p) P_{opt}(e_{01}^L) \\ &= \arg \max_{e_p} p(f_B | e_p) (mP(e_{i1}^L) + nP'(e_{i1}^L)) \\ &= \arg \max_{e_p} p(f_B | e_p) \left( m \sum_{i=1}^L P(e_{ii'} | e_{i1}^{i'-1}) \right. \\ &\quad \left. + n \left( \sum_{i''=-3}^{-1} P(w_{i''1}^{L_{i''}}) + P(w_{01}^L) \right. \right. \\ &\quad \left. \left. + \sum_{i'''=1}^3 P(w_{i'''1}^{L_{i'''}}) \right) \right). \end{aligned} \quad (5)$$

The translation probability  $p(f|e)$ , reordering function  $d$  and the language model  $p(e_p)$  are all contribute to the best output:

$$p(f_B | e_p) = \prod_{i=1}^L \phi(f_i | e_i) d(s_i - e_{i-1} - 1). \quad (6)$$

### 3.3 Error Tolerant Strategies

Another most important contribution of this paper is the error tolerant strategies in pronunciation prediction. For a beginner, it is difficult to remember the Bishun of Chinese characters correctly. Some mistakes may make when converts a Chinese character into Bishun. Or, the Chinese learner may not know the correct pronunciation of a character, but he known the pronunciations of some parts. We will introduce the details of the strategy in this section.

#### Conversion Error

When errors occurred during convert, a correction model in our pronunciation prediction framework will be triggered. The conversion error means when a Chinese learner wants to predict the pronunciation of a Chinese character according our model, Bihua lost, Bihua substitution, Bihua insertion may occur. These may reduce the performance of our prediction model. In this paper, we consider the correction of these errors as calculate the similarity of strings.

In the correction system, we use edit distance algorithm to obtain the most similar Chinese character according to the given Bihua sequence. Edit distance<sup>3</sup> is an algorithm of quantifying how dissimilar two strings (in our paper, mean the input Bihua sequence and a standard Chinese character Bihua sequence) are to one another by counting the minimum number of operations required to transform one string into the other. These operations include: insertion, deletion and substitution.

#### LM (Language Model) Based Stroke Order Decision

For Chinese learners, they can remember some basic strokes at first, but it is difficult for them to write all Chinese characters in order (the traditional way). Given an input Bihua sequence, we obtain the stroke order probability based on language model. We follow some general rules<sup>4</sup>:

- (1) Horizontal before vertical;
- (2) Diagonals right-to-left before diagonals left-to-right;

<sup>3</sup> [https://en.wikipedia.org/wiki/Edit\\_distance](https://en.wikipedia.org/wiki/Edit_distance)

- (3) Dots and minor strokes last.

In fact, there are more stroke order rules. In this paper, we just consider these three basics. Here, we reformulate these rules as  $P_{so_1}(w_{hori}, w_{vert})$ ,  $P_{so_2}(w_{r2l}, w_{l2r})$ ,  $P_{so_3}(w_{dts}, </EOS>)$ , respectively, and use the standard language model to get the 2-gram probability based on training data:

$$P_{sod} = mP_{so_1}(w_{hori}, w_{vert}) + nP_{so_2}(w_{r2l}, w_{l2r}) + kP_{so_3}(w_{dts}, </EOS>). \quad (7)$$

The  $P_{sod}$  denotes total score obtain from a Chinese character.  $m$ ,  $n$ ,  $k$  are parameters which derived from training data.

#### Local Pronunciation to Global

In Chinese, there exist many characters that include other Chinese characters, such as “清”(“青”), “歌”(“哥”), “都”(“者”), “信”(“言”). For characters like “清” if the Chinese learner already know the pronunciation of “青”, he could pronounce “清” correctly. But it is not always work when he know the pronunciation of “言” and wants to pronounce “信”. Although not all Chinese characters like “清”(“青”), we can derive some features from these characters to predict the pronunciation of a given Chinese character (“清”) from its component (“青”).

In this paper, we define the local pronunciation to global feature (LPG) as in (8):

$$f(x) = \begin{cases} +1, & \text{pronun}(w_0) = \text{pronun}(w_{0\_1}), \\ -1, & \text{pronun}(w_0) \neq \text{pronun}(w_{0\_1}). \end{cases} \quad (8)$$

Here,  $w_0$ ,  $w_{0\_1}$  denote the current Chinese character and the component of it, respectively. And  $\text{pronun}(*)$  is the pronunciation of a Chinese character. The  $\text{pronun}(w_0) = \text{pronun}(w_{0\_1})$  means the pronunciation of character  $w_0$  is similar with its component  $w_{0\_1}$  and vice versa.

We use the LPG as an important feature in decoding part of the Chinese pronunciation prediction model. With the LPG feature, characters like “清” can be assigned high probability similar as its component.

<sup>4</sup> [https://en.wikipedia.org/wiki/Stroke\\_order](https://en.wikipedia.org/wiki/Stroke_order)

**Table 4.** Size of Chinese dictionary

Corpus	Size(character)
Chinese dictionary	20,902

**Table 5.** Size of training set and test set

Corpus	Size(Sentence)
Training set	10,000
Test Set	1,000

**Table 6.** Experimental results of translation based model (with (W)/without (WO) context information)

Items	Evaluation (%)		
	P	R	F-value
#TMCPP_WO	54.77	60.32	57.41
#TMCPP_W	69.20	72.64	70.88

## 4 Experiments

### 4.1 Data and Setup

Corpus used in this paper can be divided into two kinds, one is a Chinese dictionary<sup>5</sup> crawled online, which include much information about a Chinese character, such as its pronunciation (Pinyin), strokes (Bihua), strokes order (Bishun) and et.al, we convert a Chinese character into Bishun according to rules defined in this dictionary. Another kind of corpus is used to train and test the pronunciation prediction model, these sentences are sampled from People's Daily corpus; we divided the corpus into two data sets: training set and test set. Size of these corpora is listed is presented in Tables 4 and 5.

We use the open source machine translation toolkits Moses<sup>6</sup> [19] as the baseline to train the pronunciation prediction model. GIZA++<sup>7</sup> with the heuristic grow-diag-final-and was used to obtain the word alignments. We use the standard language models provided by the toolkits SRILM<sup>8</sup> [20]. For validate our method, we also compare the

traditional model with several optimized pronunciation prediction models.

In error tolerant experiments, we rebuild the test set according to different task. For example, when validate our method use to overcome the conversion error, when first define a random function to select an index and change, or delete a stroke in the input text.

Although the most commonly used metric to evaluate the performance of machine translation is BLEU (Bilingual Evaluation Understudy), to estimate the pronunciation prediction models effectively, the experimental results are evaluated with P (Precision), R (Recall) and F1. We introduce details of these three metrics as follows:

$$R = \frac{A}{A + C}, \quad P = \frac{A}{A + B}, \quad (9)$$

$$F1 = \frac{2 * R * P}{P + R}.$$

R is the number of correct positive results (A) divided by the number of positive results (A + C) that should have been returned, and P is the number of correct positive results (A) divided by the number of all positive results (A + B). The F score can be interpreted as a weighted average of the R and P.

### 4.2 Results

\*Experiments with the symbol “#” are results of standard inputs.

\*Experiments with the symbol “☆” are results of initial inputs (often with errors), experiments with the symbol “★” are results of corrected inputs with optimized methods.

### 4.3 Analysis and Discussion

Our first evaluation (Table 6) is on the baseline model (Translation model based Chinese Pronunciation Prediction, TMCPP), which include two results TMCPP\_WO (without context information) and TMCPP\_W (with context information). Results shown that the model with context information is outperformed the non-context model significantly. Train the pronunciation

<sup>5</sup> <http://www.chinese-tools.com/tools/sinograms.html>

<sup>6</sup> <http://www.statmt.org/moses/>

<sup>7</sup> <https://github.com/moses-smt/giza-pp>

<sup>8</sup> <http://www.speech.sri.com/projects/srilm/>

**Table 1.** Experimental results of conversion errors correction

Items	Evaluation (%)		
	P	R	F-value
★TMCPP_WO_I	46.18	49.22	47.65
★TMCPP_W_I	48.90	51.43	50.13
#TMCPP_WO	54.77	60.32	57.41
#TMCPP_W	69.20	72.64	70.88
TMCPP_WO_I_EE	52.93	58.80	55.71
★TMCPP_W_I_EE	66.79	70.32	68.51

**Table 2.** Experimental results of stroke order decision

Items	Evaluation (%)		
	P	R	F-value
★TMCPP_WO_I'	42.59	43.12	42.85
★TMCPP_W_I'	43.80	45.21	44.50
#TMCPP_WO	54.77	60.32	57.41
#TMCPP_W	69.20	72.64	70.88
★TMCPP_WO_I'_SO	53.92	58.33	56.04
★TMCPP_W_I'_SO	66.20	69.01	67.58

**Table 3.** Experimental results of TMCPP with LG optimized

Items	Evaluation (%)		
	P	R	F-value
#TMCPP_WO	54.77	60.32	57.41
#TMCPP_W	69.20	72.64	70.88
★TMCPP_WO_LG	60.95	71.86	65.96
★TMCPP_W_LG	72.20	76.59	74.33

prediction model based on context information has a powerful ability to distinguish the pronunciation from several complex situations.

Table 7 shown results in conversion error correction experiment. In this experiment, we first generate some errors occurred in Chinese character – Bishun conversion according a random function. We use the baseline model to predict the pronunciation of the input text (with errors); the performances of TMCPP\_WO\_I and TMCPP\_W\_I

are really low compared with the baselines (TMCPP\_WO and TMCPP\_W). When the input texts were optimized by the edit distance algorithm, the performances are much better, but still lower than the baseline. However, it is shown the effectiveness of our correction method.

Another part of the error tolerant mechanism is disorder of strokes. Like the conversion error correction experiment, we also generate some stroke disorder errors in the input text (Table 8). Compared with the baseline, the results of the disorder input (TMCPP\_WO\_I' and TMCPP\_W\_I') are much lower, which is because our model trained, based on “bilingual” corpus, and the output of our model is strictly constraint by the language model. With the LM (Language Model) Based Stroke Order Decision operation described in section 3.4, some disorder errors occurred in input text can be corrected before decoding. Therefore, experimental results on the corrected input text (TMCPP\_WO\_I'\_SO and TMCPP\_W\_I'\_SO) are outperformed the initial disorder experimental results. In Table 7 and Table 8, we validate the error tolerant ability in our pronunciation prediction framework. Although we just generate some errors automatically, this is a very important step that makes our framework practically.

In Table 9, we have shown the powerful of our model that predict the pronunciation of Chinese character according to its local pronunciation. We extract many richness features to make our pronunciation prediction model better. Experimental results shown that with the LPG features the optimized model is outperformed the baseline, and achieved the best performance.

## 5 Conclusions and Future Directions

In this paper, we proposed a novel approach to predict pronunciation of Chinese characters. Our method based on statistical machine translation framework, and we convert each Chinese character into Bishun firstly. To adapt our task, we introduced two important language model features to improve the performance of our prediction model: (I) predict the pronunciation of a Chinese character according to its local Bishun (LLM); (II) if a Chinese character has more than one pronunciation (polyphones in Chinese), we also extract

disambiguation information according its context (previous n characters / next n characters) (GLM). We presented three error tolerant strategies to improve the flexibility of our model. Experimental results shown that, the pronunciation of Chinese characters can be predicted effectively with our approach.

In our future work, we will try to find an optimized knowledge representation model to further improve the accuracy of Chinese character pronunciation prediction model.

## Acknowledgements

This work is supported by the West Light Foundation of The Chinese Academy of Sciences under Grant No.2015-XBQN-B-10, the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant No.XDA06030400 and the Xinjiang Key Laboratory Fund under Grant No.2015KL031. We sincerely thank the anonymous reviewers for their thorough reviewing and valuable suggestions.

## References

1. **Hsieh, S.-K. (2006).** *Concept and Computation: A preliminary survey of Chinese Characters as a Knowledge Resource in NLP*. Universität Tübingen.
2. **Byrd, R.J. & Tzoukermann, E. (1988).** Adapting an English morphological analyzer for French. *Proceedings of the 26th annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, pp. 1–6. DOI: 10.3111/982023.982024.
3. **Koehn, P., Och, F.J., & Marcu, D. (2003).** Statistical phrase-based translation. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*. Association for Computational Linguistics, pp. 48–54. DOI: 10.3115/1073445.1073462.
4. **Zens, R., Och, F.J., & Ney, H. (2002).** Phrase-based statistical machine translation. *Advances in Artificial Intelligence*. Springer Berlin Heidelberg, pp. 18–32. DOI: 10.1007/3-540-45751-8\_2.
5. **Och, F.J. & Ney, H. (2004).** The alignment template approach to statistical machine translation. *Computational linguistics*, Vol. 30, No. 4, pp. 417–449. DOI: 10.1162/0891201042544884.
6. **Shi, X., Knight, K., & Ji, H. (2014).** *How to Speak a Language without Knowing it*.
7. **Lin, C.-C. & Tsai, R.T.H. (2012).** A Generative Data Augmentation Model for Enhancing Chinese Dialect Pronunciation Prediction. *Audio, Speech, and Language Processing, IEEE Transactions on*, Vol. 20, No. 4, pp. 1109–1117. DOI: 10.1109/tasl.2011.2172424.
8. **Brown, P.F., Pietra, V.J.D., Pietra, S.A.D., & Mercer, R.L. (1993).** The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, Vol. 19, No. 2, pp. 263–311.
9. **Tran, V.H., Pham, A.T., Nguyen, V.V., Nguyen, H.X., & Nguyen, H.Q. (2015).** Parameter Learning for Statistical Machine Translation Using CMA-ES. *Knowledge and Systems Engineering*. Springer International Publishing, pp. 425–432. DOI: 10.1007/978-3-319-11679-2.
10. **Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. (2003).** A neural probabilistic language model. *The Journal of Machine Learning Research*, pp. 1137–1155.
11. **Schwenk, H. (2007).** Continuous space language models. *Computer Speech & Language*, Vol. 21, No. 3, pp. 492–518.
12. **Brown, P.F., deSouza, P.V., Mercer, R.L., Pietra, V.J., & Lai, J.C. (1992).** Class-based n-gram models of natural language. *Computational linguistics*, Vol. 18, No. 4, pp. 467–479.
13. **Buck, C., Heafield, K., & van Ooyen, B. (2014).** N-gram counts and language models from the common crawl. *Proceedings of the Language Resources and Evaluation Conference*.
14. **Singh, A. (2005).** *The EM Algorithm*.
15. **Do, C.B. & Batzoglou, S. (2008).** What is the expectation maximization algorithm? *Nature biotechnology*, Vol. 26, No. 8, pp. 897–900.
16. **Dempster, A.P., Laird, N.M., & Rubin, D.B. (1977).** Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38.
17. **Liu, Y., Xia, T., Xiao, X., & Liu, Q. (2009).** Weighted alignment matrices for statistical machine translation. *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Vol. 2*. Association for Computational Linguistics, pp. 1017–1026.
18. **Niehues, J. & Vogel, S. (2008).** Discriminative word alignment via alignment matrix modeling. *Proceedings of the Third Workshop on Statistical Machine Translation*. Association for Computational Linguistics, pp. 18–25.

19. Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., & Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*. Association for Computational Linguistics, pp. 177-180.

20. Stolcke A. (2002). SRILM-an extensible language modeling toolkit. *INTERSPEECH*.

**Chenggang Mi** received his PhD in Natural Language Processing and Machine Translation from the University of Chinese Academy of Sciences. Currently, he is an assistant professor in Xinjiang Technical Institute of Physics and Chemistry of Chinese Academy of Sciences. His research interests are: machine translation, natural language processing and machine learning.

**Yating Yang** received her PhD from the Graduate University of Chinese Academy of Sciences. Currently, she is an associate professor in Xinjiang Technical Institute of Physics and Chemistry of

Chinese Academy of Sciences. Her research interest is machine translation.

**Xi Zhou** received his PhD from the University of Chinese Academy of Sciences. Currently, he is a professor in Xinjiang Technical Institute of Physics and Chemistry of Chinese Academy of Sciences. His research interest is multilingual processing.

**Lei Wang** received his PhD from the University of Chinese Academy of Sciences. Currently, he is a professor in Xinjiang Technical Institute of Physics and Chemistry of Chinese Academy of Sciences. His research interest is multilingual processing.

**Xiao Li** is currently a professor in Xinjiang Technical Institute of Physics and Chemistry of Chinese Academy of Sciences. His research interest is multilingual processing.

**Tonghai Jiang** is currently a professor in Xinjiang Technical Institute of Physics and Chemistry of Chinese Academy of Sciences. His research interest is multilingual processing.

Article received on 15/01/2016; accepted 25/02/2016.  
Corresponding author is Chenggang Mi.