Can Chinese Phonemes Improve Machine Transliteration?: A Comparative Study of English-to-Chinese Transliteration Models

Jong-Hoon Oh, Kiyotaka Uchimoto, and Kentaro Torisawa
Language Infrastructure Group, MASTAR Project,
National Institute of Information and Communications Technology (NICT)
3-5 Hikaridai Seika-cho, Soraku-gun, Kyoto 619-0289 Japan
{rovellia,uchimoto,torisawa}@nict.go.jp

Abstract

Inspired by the success of English grapheme-to-phoneme research in speech synthesis, many researchers have proposed phoneme-based English-to-Chinese transliteration models. However, such approaches have severely suffered from the errors in Chinese phoneme-to-grapheme conversion. To address this issue, we propose a new English-to-Chinese transliteration model and make systematic comparisons with the conventional models. Our proposed model relies on the joint use of Chinese phonemes and their corresponding English graphemes and phonemes. Experiments showed that Chinese phonemes in our proposed model can contribute to the performance improvement in English-to-Chinese transliteration.

1 Introduction

1.1 Motivation

Transliteration, i.e., *phonetic translation*, is commonly used to translate proper names and technical terms across languages. A variety of English-to-Chinese machine transliteration models has been proposed in the last decade (Meng et al., 2001; Gao et al., 2004; Jiang et al., 2007; Lee and Chang, 2003; Li et al., 2004; Li et al., 2007; Wan and Verspoor, 1998; Virga and Khudanpur, 2003). They can be categorized into those based on Chinese phonemes (Meng et al., 2001; Gao et al., 2004; Jiang et al., 2007; Lee and Chang, 2003; Wan and Verspoor, 1998; Virga and Khudanpur, 2003) and those that don't rely on Chinese phonemes (Li et al., 2004; Li et al., 2007).

Inspired by the success of English grapheme-to-phoneme research in speech synthesis, many researchers have proposed phoneme-based English-to-Chinese transliteration models. In these approaches, Chinese phonemes are generated from English graphemes or phonemes, and then the Chinese phonemes are converted into Chinese graphemes (or characters), where Chinese Pinyin strings[1] are used for representing a syllable-level Chinese phoneme sequence. Despite its high accuracy in generating Chinese phonemes from English, this approach has severely suffered from errors in Chinese phoneme-to-grapheme conversion, mainly caused by Chinese homophone confusion – one Chinese Pinyin string can correspond to several Chinese characters (Li et al., 2004). For example, the Pinyin string "LI" corresponds to such different Chinese characters as 利, 莉, and 里. For this reason, it has been reported that English-to-Chinese transliteration without Chinese phonemes outperforms that with Chinese phonemes (Li et al., 2004).

Then "Can Chinese phonemes improve English-to-Chinese transliteration, if we can reduce the errors in Chinese phoneme-to-grapheme conversion?" Our research starts from this question.

1.2 Our Approach

Previous approaches using Chinese phonemes have relied only on Chinese phonemes in Chinese phoneme-to-grapheme conversion. However, the simple use of Chinese phonemes doesn't always provide a good clue to reduce the ambiguity in Chinese phoneme-to-grapheme conversion. Let us explain with an example, the Chinese transliteration of *Greeley* in Table 1, where Chinese phonemes are represented in terms of Chinese Pinyin strings and English phonemes are represented by *ARPAbet* symbols[2].

In Table 1, Chinese Pinyin string "LI" corresponds to two different Chinese characters, 里 and

[1] Pinyin, the most commonly used Romanization system for Chinese characters, faithfully represents Chinese

Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pages 658–667,
Singapore, 6-7 August 2009.

Table 1: Chinese Pinyin string "LI" and its corresponding Chinese characters in Chinese transliteration of *Greeley*

English grapheme	*g*	*ree*	*ley*
English phoneme	G	R IY	L IY
Chinese Pinyin	GE	LI	LI
Chinese character	格	里	利

利. It seems difficult to find evidence for selecting the correct Chinese character corresponding to each Chinese Pinyin string "LI" by just looking at the sequence of Chinese Pinyin strings "GE LI LI." However, English graphemes (*ree* and *ley*) or phonemes ("R IY" and "L IY") corresponding to Chinese Pinyin string "LI", especially their consonant parts (*r* and *l* in the English graphemes and "R" and "L" in the English phonemes), provide strong evidence to resolve the ambiguity. Thus, we can easily find rules for the conversion from Chinese Pinyin string "LI" to 里 and 利 as follows:

- ⟨ "R IY", LI ⟩ → 里
- ⟨ "L IY", LI ⟩ → 利

Based on the observation, we propose an English-to-Chinese transliteration model based on the joint use of Chinese phonemes and their corresponding English graphemes and phonemes. We define a set of English-to-Chinese transliteration models and categorize them into the following three classes:

- $\mathbf{M_I}$: **M**odels **I**ndependent of Chinese phonemes
- $\mathbf{M_S}$: **M**odels based on **S**imple use of Chinese phonemes
- $\mathbf{M_J}$: **M**odels based on **J**oint use of Chinese phonemes and English graphemes and phonemes that correspond to our proposed model.

Our comparison among the three types of transliteration models can be summarized as follows.

- The $\mathbf{M_I}$ models relying on either English graphemes or phonemes could not outperform those based on both English graphemes and phonemes.
- The $\mathbf{M_S}$ models always showed the worst performance due to the severe error rate in Chinese phoneme-to-grapheme conversion.
- The $\mathbf{M_J}$ models significantly reduced errors in Chinese phoneme-to-grapheme conversion; thus they achieved the best performance.

The rest of this paper is organized as follows. Section 2 introduces the notations used throughout this paper. Section 3 describes the transliteration models we compared. Section 4 describes our tests and results. Section 5 concludes the paper with a summary.

2 Preliminaries

Let E_G be an English word composed of *n* English graphemes, and let E_P be a sequence of English phonemes that represents the pronunciation of E_G. Let C_G be a sequence of Chinese graphemes corresponding to the Chinese transliteration of E_G, and let C_P be a sequence of Chinese phonemes that represents the pronunciation of C_G.

C_P corresponds to a sequence of the Chinese Pinyin strings of C_G. Because a Chinese Pinyin string represents the pronunciation of a syllable consisting of consonants and vowels, we divide a Chinese Pinyin string into consonant and vowel parts like "L+I", "L+I+N", and "SH+A." In this paper, we define a *Chinese phoneme* as the vowel and consonant parts in a Chinese Pinyin string (e.g., "L", "SH", and "I"). A Chinese character usually corresponds to multiple English graphemes, English phonemes, and Chinese phonemes (i.e., 里 corresponds to English graphemes *ree*, English phonemes "R IY", and Chinese phonemes "L I" in Table 1). To represent these many-to-one correspondences, we use the well-known BIO labeling scheme to represent a Chinese character, where B and I represent the beginning and inside/end of the Chinese characters, respectively, and O is not used. Each Chinese phoneme corresponds to a Chinese character with B and I labels. For example, Chinese character "里" in Table 1 can be represented as "里:B" and "里:I", where "里:B" and "里:I" correspond to Chinese phonemes "L" and "I", respectively. In this paper, we define a *Chinese grapheme* as a Chinese character represented with a BIO label, e.g., "里:B" and "里:I."

phonemes and syllables (Yin and Felley, 1990).

[2] `http://www.cs.cmu.edu/~laura/pages/arpabet.ps`

Table 2: eg_i and its corresponding ep_i, cp_i, and cg_i in *Greeley* and its corresponding Chinese transliteration "格里利"

i	1	2	3	4	5	6	7
E_G	*g*	*r*	*e*	*e*	*l*	*e*	*y*
E_P	G	R	IY	ϕ	L	IY	ϕ
C_P	GE	L	I	ϕ	L	I	ϕ
	GE	LI		ϕ	LI		ϕ
C_G	格:B	里:B	里:I	ϕ	利:B	利:I	ϕ
	格	里		ϕ	利		ϕ

Then E_P, C_P, and C_G can be segmented into a series of sub-strings, each of which corresponds to an English grapheme in E_G. We can thus write

- $E_G = eg_1, \cdots, eg_n = eg_1^n$
- $E_P = ep_1, \cdots, ep_n = ep_1^n$
- $C_P = cp_1, \cdots, cp_n = cp_1^n$
- $C_G = cg_1, \cdots, cg_n = cg_1^n$

where eg_i, ep_i, cp_i, and cg_i represent the i^{th} English grapheme, English phonemes, Chinese phonemes, and Chinese graphemes corresponding to eg_i, respectively.

Based on the definition, we model English-to-Chinese transliteration so that each English grapheme is tagged with its corresponding English phonemes, Chinese phonemes, and Chinese graphemes. Table 2 illustrates eg_i, ep_i, cp_i, and cg_i with the same example listed in Table 1 (English word *Greeley* and its corresponding Chinese transliteration "格里利")[3], where ϕ represents an empty string.

3 Transliteration Model

We defined eighteen transliteration models to be compared. These transliteration models are classified into three classes, $\mathbf{M_I}$, $\mathbf{M_S}$, and $\mathbf{M_J}$ as described in Section 1.2; each class has three basic transliteration models and three hybrid ones. In this section, we first describe the basic transliteration models in each class by focusing on the main difference among the three classes and then describe the hybrid transliteration models.

[3]We performed alignment between E_G and E_P and between E_P and C_P in a similar manner presented in Li et al. (2004). Then the two alignment results were merged using E_P as a pivot. Finally, we made a correspondence relation among eg_i, ep_i, cp_i, and cg_i using the merged alignment result and the Pinyin table.

3.1 Basic Transliteration Models

The basic transliteration models in each class are denoted as $M(x, y)$.

- $(x, y) \in X \times Y$
- $x \in X = \{E_G, E_P, E_{GP}\}$
- $y \in Y = \{\phi, C_P, JC_P\}$

x is an English-side parameter representing English grapheme (E_G), English phoneme (E_P), and the joint use of English grapheme and phoneme ($E_{GP} = \langle E_G, E_P \rangle$) that contributes to generating Chinese phonemes or Chinese graphemes in a transliteration model. y is a Chinese-phoneme parameter that represents a way of using Chinese phonemes to generate Chinese graphemes in a transliteration model. Since $M(x, \phi)$ represents a transliteration model that does not rely on Chinese phonemes, it falls into $\mathbf{M_I}$, while $M(x, C_P)$ corresponds to a transliteration model in $\mathbf{M_S}$ that only uses Chinese phonemes in Chinese phoneme-to-grapheme conversion. $M(x, JC_P)$ is a transliteration model in the $\mathbf{M_J}$ class that generates Chinese transliterations based on joint use of x and Chinese phoneme C_P, where $x \in X$. Thus, $M(x, JC_P)$ can be rewritten as $M(x, \langle x, C_P \rangle)$, where the joint representation of x and C_P, $\langle x, C_P \rangle$, is used in Chinese phoneme-to-grapheme conversion. The three basic models in $\mathbf{M_J}$ can be interpreted as follows:

- $M(E_G, JC_P) = M(E_G, \langle E_G, C_P \rangle)$
- $M(E_P, JC_P) = M(E_P, \langle E_P, C_P \rangle)$
- $M(E_{GP}, JC_P) = M(E_{GP}, \langle E_{GP}, C_P \rangle)$

$M(E_G, JC_P)$ directly converts English graphemes into Chinese phonemes without the help of English phonemes and then generates Chinese transliterations based on the joint representation of English graphemes and Chinese phonemes. The main difference between $M(E_P, JC_P)$ and $M(E_{GP}, JC_P)$ lies in the use of English graphemes to generate Chinese phonemes and graphemes. English graphemes are only used in English grapheme-to-phoneme conversion, and English phonemes play a crucial role for generating Chinese transliteration in $M(E_P, JC_P)$. Chinese phoneme-to-grapheme conversion that relies on the joint use of English graphemes, English phonemes, and Chinese

$$P_{M(E_G,JC_P)}(C_G|E_G) = \sum_{\forall C_P} P(C_P|E_G) \times P(C_G|E_G, C_P) \quad (1)$$

$$P_{M(E_P,JC_P)}(C_G|E_G) = \sum_{\forall C_P} \sum_{\forall E_P} P(E_P|E_G) \times P(C_P|E_P) \times P(C_G|E_P, C_P) \quad (2)$$

$$P_{M(E_{GP},JC_P)}(C_G|E_G) = \sum_{\forall C_P} \sum_{\forall E_P} P(E_P|E_G) \times P(C_P|E_G, E_P) \times P(C_G|E_G, E_P, C_P) \quad (3)$$

$$P_{M(E_G,C_P)}(C_G|E_G) = \sum_{\forall C_P} P(C_P|E_G) \times P(C_G|C_P) \quad (4)$$

$$P_{M(E_P,C_P)}(C_G|E_G) = \sum_{\forall C_P} \sum_{\forall E_P} P(E_P|E_G) \times P(C_P|E_P) \times P(C_G|C_P) \quad (5)$$

$$P_{M(E_{GP},C_P)}(C_G|E_G) = \sum_{\forall C_P} \sum_{\forall E_P} P(E_P|E_G) \times P(C_P|E_G, E_P) \times P(C_G|C_P) \quad (6)$$

phonemes is the key feature of $M(E_{GP}, JC_P)$. Because $M(x, JC_P)$ can be interpreted as $M(x, \langle x, C_P\rangle)$, English-side parameter x determines the English graphemes and phonemes, or both jointly used with Chinese phonemes in Chinese phoneme-to-grapheme conversion. Then we can represent the three basic transliteration models as in Eqs. (1)–(3), where $P(C_G|E_G, C_P)$, $P(C_G|E_P, C_P)$, and $P(C_G|E_G, E_P, C_P)$ are the key points in our proposed models, $\mathbf{M_J}$.

The three basic transliteration models in $\mathbf{M_S}$ – $M(E_G, C_P)$, $M(E_P, C_P)$, and $M(E_{GP}, C_P)$ – are formulated as Eqs. (4)–(6). Chinese phoneme-based transliteration models in the literature fall into either $M(E_G, C_P)$ or $M(E_P, C_P)$ (Meng et al., 2001; Gao et al., 2004; Jiang et al., 2007; Lee and Chang, 2003; Wan and Verspoor, 1998; Virga and Khudanpur, 2003). The three basic transliteration models in $\mathbf{M_S}$ are identical as those in $\mathbf{M_J}$, except for the Chinese phoneme-to-grapheme conversion method. They only depend on Chinese phonemes in Chinese phoneme-to-grapheme conversion represented as $P(C_G|C_P)$ in Eqs. (4)–(6).

$$P_{M(E_G,\phi)}(C_G|E_G) = P(C_G|E_G) \quad (7)$$

$$P_{M(E_P,\phi)}(C_G|E_G) = \sum_{\forall E_P} P(E_P|E_G) \times P(C_G|E_P) \quad (8)$$

$$P_{M(E_{GP},\phi)}(C_G|E_G) = \sum_{\forall E_P} P(E_P|E_G) \times P(C_G|E_G, E_P) \quad (9)$$

The three basic transliteration models in $\mathbf{M_I}$ are represented in Eqs. (7)–(9). Because the $\mathbf{M_I}$ models are independent of Chinese phonemes, they are the same as the transliteration models in the literature used for machine transliteration from English to other languages without relying on target-language phonemes (Karimi et al., 2007; Malik, 2006; Oh et al., 2006; Sherif and Kondrak, 2007; Yoon et al., 2007). Note that $M(E_G, \phi)$ is the same transliteration model as the one proposed by Li et al. (2004).

3.2 Hybrid Transliteration Models

The hybrid transliteration models in each class are defined by discrete mixture between the probability distribution of the two basic transliteration models, as in Eq. (10) (Al-Onaizan and Knight, 2002; Oh et al., 2006), where $0 < \alpha < 1$. We denote a hybrid transliteration model between two basic transliteration models $M(x_1, y)$ and $M(x_2, y)$ as $M(x_1 + x_2, y, \alpha)$, where $y \in Y = \{\phi, C_P, JC_P\}$, $x_1 \neq x_2$, and $x_1, x_2 \in X = \{E_G, E_P, E_{GP}\}$. In this paper, we define three types of hybrid transliteration models in each class: $M(E_G + E_P, y, \alpha)$, $M(E_G + E_{GP}, y, \alpha)$, and $M(E_P + E_{GP}, y, \alpha)$.

$$P_{M(x_1+x_2,y,\alpha)}(C_G|E_G) = \alpha \times P_{M(x_1,y)}(C_G|E_G) + (1-\alpha) \times P_{M(x_2,y)}(C_G|E_G) \quad (10)$$

3.3 Probability Estimation

Because Eqs. (1)–(9) can be estimated in a similar way, we limit our focus to Eq. (3) in this section. Assuming that $P(E_P|E_G)$, $P(C_P|E_G, E_P)$, and $P(C_G|E_G, E_P, C_P)$ in Eq. (3) depend on the size of the context window, k ($k = 3$ in this paper),

Table 3: Feature functions for $P(cg_i|cg_{i-k}^{i-1}, \langle eg, ep, cp\rangle_{i-k}^{i+k})$ with an example in Table 2, where $i = 2$

f_1	$gram_3(eg_i)$	eg_i^{i+2} = "ree"	cg_i = "里:B"
f_2	$pair_{11}(cp_{i-1}, cg_{i-1})$	cp_{i-1} = "G", cg_{i-1} = "格:B"	cg_i = "里:B"
f_3	$pair_{12}(cg_{i-1}, cp_{i-1})$	cp_{i-1}^i = "GE L", cg_{i-1}= "格:B"	cg_i = "里:B"
f_4	$pair_{22}(cp_{i-1}, cg_{i-2})$	eg_{i-1}^i = "gr", ep_{i-1}^i = "G R"	cg_i = "里:B"
f_5	$triple_1(eg_i, cp_i, cg_{i-1})$	eg_i = "r", cp_{i-1} = "GE", cg_{i-1}= "格:B"	cg_i = "里:B"
f_6	$triple_2(eg_{i-1}, cg_{i-1}, cp_{i-1})$	eg_{i-1} = "g", cp_{i-1}^i= "GE L", cg_{i-1}= "格:B"	cg_i = "里:B"

they can be simplified into a series of products in Eqs. (11)–(13).

The maximum entropy model is used to estimate the probabilities in Eqs. (11)–(13) (Berger et al., 1996). Generally, a conditional maximum entropy model is an exponential model that gives the conditional probability, as described in Eq. (14), where λ_i is the parameter to be estimated and $f_i(a, b)$ is a feature function corresponding to λ_i (Berger et al., 1996; Ratnaparkhi, 1997):

$$P(E_P|E_G) \approx \prod_i P(ep_i|ep_{i-k}^{i-1}, eg_{i-k}^{i+k}) \quad (11)$$

$$P(C_P|E_G, E_P) \quad (12)$$
$$\approx \prod_i P(cp_i|cp_{i-k}^{i-1}, \langle eg, ep\rangle_{i-k}^{i+k})$$

$$P(C_G|E_G, E_P, C_P) \quad (13)$$
$$\approx \prod_i P(cg_i|cg_{i-k}^{i-1}, \langle eg, ep, cp\rangle_{i-k}^{i+k})$$

$$P(b|a) = \frac{exp(\sum_i \lambda_i f_i(a, b))}{\sum_{b'} exp(\sum_i \lambda_i f_i(a, b'))} \quad (14)$$

$f_i(a, b)$ is a binary function returning TRUE or FALSE based on context a and output b. If $f_i(a, b)$=1, its corresponding model parameter λ_i contributes toward conditional probability $P(b|a)$ (Berger et al., 1996; Ratnaparkhi, 1997). The feature functions used here are defined in terms of context predicates — a function returning TRUE or FALSE that depends on the presence of the information in the current context (Ratnaparkhi, 1997). Context predicates and their descriptions used are given in Table 4.

N-GRAM includes $gram_1(u_j)$, $gram_2(u_j)$, and $gram_3(u_j)$ corresponding to a unigram, a bigram, and a trigram, respectively. PAIR includes a pair of unigrams ($pair_{11}$), unigram and bigram ($pair_{12}$), and bigrams ($pair_{22}$). TRIPLE includes a triple of three unigrams ($triple_1$) and a triple of two unigrams and one bigram ($triple_2$). Note that if different context predicates represent the same context, we accept one of them and ignore the others

Table 4: Context predicates and their descriptions

Category	Context predicates	Description
N-GRAM	$gram_1(u_j)$	u_j
	$gram_2(u_j)$	u_j^{j+1}
	$gram_3(u_j)$	u_j^{j+2}
PAIR	$pair_{11}(u_j, v_k)$	u_j, v_k
	$pair_{12}(u_j, v_k)$	u_j, v_k^{k+1}
	$pair_{22}(u_j, v_k)$	u_j^{j+1}, v_k^{k+1}
TRIPLE	$triple_1(u_j, v_k, w_l)$	u_j, v_k, w_l
	$triple_2(u_j, v_k, w_l)$	u_j, v_k, w_l^{l+1}

(e.g., $pair_{12}(u_j, u_{j+1}) = trigram(u_j) = u_j^{j+2}$). Table 3 represents the examples of feature functions for $P(cg_i|cg_{i-k}^{i-1}, \langle eg, ep, cp\rangle_{i-k}^{i+k})$.

We used the "Maximum Entropy Modeling Toolkit"[4] to estimate the probabilities and the LBFGS algorithm to find λ_i in Eq. (14). For each transliteration model, we produced n-best transliterations using a stack decoder (Schwartz and Chow, 1990).

3.4 Summary

In this paper, we defined eighteen transliteration models to be compared. There are six transliteration models, three basic and three hybrid ones, in each class, $\mathbf{M_I}$, $\mathbf{M_S}$, and $\mathbf{M_J}$. We compared the transliteration models from the viewpoint of Chinese phonemes or the class of transliteration models in our experiments.

4 Testing and Results

We used the same test set used in Li et al. (2004) for our testing[5]. It contains 37,694 pairs of English words and their official Chinese transliterations

[4]Available at `http://homepages.inf.ed.ac.uk/s0450736/maxent_toolkit.html`

[5]This test set was also used in "NEWS09 machine transliteration shared task" for English-to-Chinese transliteration (Li et al., 2009)

extracted from the "Chinese Transliteration of Foreign Personal Names" (Xinhua News Agency, 1992), which includes names in English, French, German, and many other foreign languages (Li et al., 2004). We used the same test data as in Li et al. (2004). But we randomly selected 90% of the training data used in Li et al. (2004) as our training data and the remainder as the development data, as shown in Table 5.

Table 5: Number of English-Chinese transliteration pairs in each data set

	Ours	Li et al. (2004)
Training data	31,299	34,777
Development data	3,478	N/A
Blind test data	2,896	2,896

We used the training data for training the transliteration models. For each model, we tuned the parameters including the number of iterations for training the maximum entropy model and a Gaussian prior for smoothing the maximum entropy model using the development data. Further, the development data was used to select parameter α of the hybrid transliteration models. We varied parameter α from 0 to 1 in 0.1 intervals (i.e., α=0, 0.1, 0.2, $\cdots$,1) and tested the performance of the hybrid models with the development data. Then we chose α that showed the best performance in each hybrid model. The blind test data was used for evaluating the performance of each transliteration model. *The CMU Pronouncing Dictionary*[6], which contains about 120,000 English words and their pronunciations, was used for estimating $P(E_P|E_G)$.

We conducted two experiments. First, we compared the overall performance of the transliteration models. Second, we investigated the effect of training data size on the performance of each transliteration model.

The evaluation was done for word accuracy in top-1 (ACC), Chinese pronunciation accuracy (CPA) and a mean reciprocal rank (MRR) metric (Kantor and Voorhees, 2000; Li et al., 2009; Chang et al., 2009). ACC measures how many correct transliterations appeared in the top-1 result of each system. CPA measures the Chinese pronunciation accuracy in the top-1 of the n-best Chinese pronunciation. We used CPA for comparing the performance between systems based on Chinese phonemes. MRR, mean reciprocal ranks of n-best results of each system over the test entries, is an evaluation measure for n-best transliterations. If a transliteration generated by a system matches a reference transliteration[7] at the r^{th} position of the n-best results, its reciprocal rank equals $1/r$; otherwise its reciprocal rank equals 0, where $1 \leq r \leq n$. We produced 10-best Chinese transliterations for each English word in our experiments.

4.1 Comparison of the Overall Performance

Table 6 represents the overall performance of one system in a previous work (Li et al., 2004) and eighteen systems based on the transliteration models defined in this paper. ACC, MRR, and CPA represent the evaluation results for each model trained by our training data. To test transliteration models without the errors introduced by incorrect Chinese phonemes, we carried out the experiments with the correct Chinese pronunciation (or the correct Chinese phoneme sequence) in Chinese phoneme-to-grapheme conversion. In the experiment, we put the correct Chinese pronunciation into the top-1 of the n-best Chinese pronunciation with the highest probability, say $P(C_P|E_G)$=1; thus CPA was assumed to be 100%. The ACC of the transliteration models under this condition is denoted as ACC' in Table 6. TRAIN represents the evaluation results of the transliteration models trained by our training data. To compare Li et al. (2004) and transliteration models defined in this paper under the same condition, we also carried out experiments with the same training data in Li et al. (2004). Since the training data used in Li et al. (2004) is identical as the union of our training and development data, we denoted it as TRAIN+DEV in Table 6. In both TRAIN and TRAIN+DEV, we used the same parameter setting that was obtained by using the development data.

LI04 represents a system in Li et al. (2004), and its ACC' in TRAIN+DEV is taken from the literature. The systems based on the transliteration models defined in our paper are represented from the second row in Table 6. The phoneme-based transliteration models in the literature correspond to either $M(E_G, C_P)$ (Wan and Verspoor, 1998; Lee and Chang, 2003; Jiang et al., 2007) or $M(E_P, C_P)$ (Meng et al., 2001; Gao et al., 2004;

[6]Available at `http://www.speech.cs.cmu.edu/cgi-bin/cmudict`

[7]In our test set, an English word corresponds to one reference Chinese transliteration.

Table 6: Comparison of the overall performance

Class	Model	TRAIN				TRAIN+DEV			
		ACC	MRR	CPA	ACC'	ACC	MRR	CPA	ACC'
LI04		N/A	N/A	N/A	N/A	70.1	N/A	N/A	N/A
$\mathbf{M_J}$	$M(E_G, JC_P)$	71.9	80.4	72.3	88.2	72.3	80.7	73.1	88.9
	$M(E_P, JC_P)$	61.1	70.3	62.4	82.8	61.1	70.6	63.1	83.8
	$M(E_{GP}, JC_P)$	72.3	80.9	73.2	89.6	73.5	81.5	73.9	90.4
	$M(E_G{+}E_P, JC_P, 0.7)$	72.8	80.7	73.8	89.7	73.2	81.0	74.7	90.5
	$M(E_G{+}E_{GP}, JC_P, 0.6)$	**73.5**	**81.7**	**74.2**	**90.6**	**73.7**	**81.8**	**74.8**	**91.2**
	$M(E_P{+}E_{GP}, JC_P, 0.1)$	71.6	80.3	73.3	89.8	72.5	80.8	73.8	90.1
$\mathbf{M_I}$	$M(E_G, \phi)$	70.0	78.5	N/A	N/A	70.6	79.0	N/A	N/A
	$M(E_P, \phi)$	58.5	69.3	N/A	N/A	59.4	70.1	N/A	N/A
	$M(E_{GP}, \phi)$	71.2	79.9	N/A	N/A	72.3	80.7	N/A	N/A
	$M(E_G{+}E_P, \phi, 0.7)$	70.7	79.1	N/A	N/A	72.0	80.0	N/A	N/A
	$M(E_G{+}E_{GP}, \phi, 0.4)$	72.0	80.3	N/A	N/A	72.8	80.9	N/A	N/A
	$M(E_P{+}E_{GP}, \phi, 0.1)$	71.0	79.6	N/A	N/A	72.0	80.4	N/A	N/A
$\mathbf{M_S}$	$M(E_G, C_P)$	58.9	70.2	72.3	78.4	59.1	70.4	73.1	78.4
	$M(E_P, C_P)$	50.2	62.3	62.4	78.4	50.4	62.6	63.1	78.5
	$M(E_{GP}, C_P)$	59.1	70.4	73.2	78.4	59.3	70.5	73.9	78.5
	$M(E_G{+}E_P, C_P, 0.8)$	59.7	71.3	73.8	79.0	60.3	71.7	74.7	79.0
	$M(E_G{+}E_{GP}, C_P, 0.6)$	59.8	71.7	74.2	78.9	60.6	72.1	74.8	78.9
	$M(E_P{+}E_{GP}, C_P, 0.1)$	58.8	70.4	73.3	78.9	59.4	70.7	73.8	78.8

Virga and Khudanpur, 2003).

A comparison between the basic and hybrid transliteration models showed that the hybrid ones usually performed better (the exception was $M(E_P{+}E_{GP}, y, \alpha)$ but the performance still comparable to the basic ones in each class). Especially, the hybrid ones based on the best two basic transliteration models, $M(E_G{+}E_{GP}, y, \alpha)$, showed the best performance.

A comparison among the $\mathbf{M_I}$, $\mathbf{M_S}$, and $\mathbf{M_J}$ models showed that Chinese phonemes did contribute to the performance improvement of English-to-Chinese transliteration when Chinese phonemes were used together with their corresponding English graphemes and phonemes in Chinese phoneme-to-grapheme conversion. A one-tail paired t-test between the $\mathbf{M_I}$ and $\mathbf{M_J}$ models showed that the results of the $\mathbf{M_J}$ models were always significantly better than those of the $\mathbf{M_I}$ models if the $\mathbf{M_I}$ and $\mathbf{M_J}$ models shared the same English-side parameter, $x \in \{E_G, E_P, E_{GP}\}$ (level of significance = 0.001). In the results obtained by the $\mathbf{M_S}$ and $\mathbf{M_J}$ models, the figures in CPA are the same when the $\mathbf{M_S}$ and our $\mathbf{M_J}$ models share the same English-side parameter. Moreover, the difference between the figures in ACC and CPA can be interpreted as the error rate of Chinese phoneme-to-grapheme conversion. Our proposed $\mathbf{M_J}$ models generated Chinese transliterations with a very low error rate in Chinese phoneme-to-grapheme conversion, while the $\mathbf{M_S}$ models suffered from a significant error rate in Chinese phoneme-to-grapheme conversion. ACC' showed that the $\mathbf{M_J}$ models still outperformed the $\mathbf{M_S}$ models even without errors in generating Chinese pronunciation from the English words. These results indicate that the joint use of Chinese phonemes and their corresponding English graphemes and phonemes significantly improved the performance in Chinese phoneme-to-grapheme conversion and English-to-Chinese transliteration.

Table 7 shows the Chinese transliterations generated by $M(E_G, \phi)$, $M(E_{GP}, \phi)$, $M(E_G, JC_P)$, and $M(E_{GP}, JC_P)$ where English or Chinese phonemes contributed to the correct transliteration. In this table, the first column show the English words and their English phonemes, and the second and third columns represent the Chinese transliterations and their phonemes. Note that the Chinese phonemes in the second and third columns of the $\mathbf{M_I}$ models are not used in transliteration. They are shown in the table to indicate the difference in the Chinese phonemes of Chinese

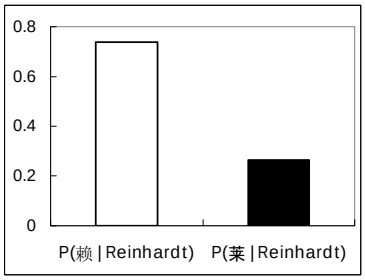
Table 7: Top-1 results of $M(E_G, \phi)$, $M(E_{GP}, \phi)$, $M(E_G, JC_P)$, and $M(E_{GP}, JC_P)$, where * represents incorrect transliterations

$\mathbf{M_I}$ models	$M(E_G, \phi)$	$M(E_{GP}, \phi)$
Emily (EH M IH L IY)	埃米利* (AI MI LI)	埃米利* (AI MI LI)
Ivy (AY V IY)	伊维* (YI WEI)	艾维 (AI WEI)
Reinhardt (R AI N HH AA R T)	赖因哈特* (LAI YIN HA TE)	赖因哈特* (LAI YIN HA TE)
$\mathbf{M_J}$ models	$M(E_G, JC_P)$	$M(E_{GP}, JC_P)$
Emily (EH M IH L IY)	埃米莉 AI MI LI	埃米莉 AI MI LI
Ivy (AY V IY)	伊维* YI WEI	艾维 AI WEI
Reinhardt (R AI N HH AA R T)	莱因哈特 LAI YIN HA TE	莱因哈特 LAI YIN HA TE

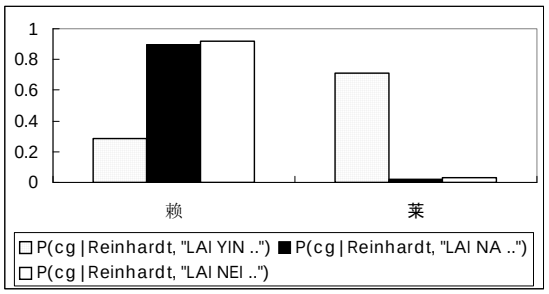
transliterations between the $\mathbf{M_I}$ and $\mathbf{M_J}$ models.

For *Emily* and *Reinhardt*, the $\mathbf{M_J}$ models generated correct Chinese transliterations, but the $\mathbf{M_I}$ models did not. Figure 1 shows the probability distribution when a transliteration model generates the first Chinese character in the Chinese transliteration of *Reinhardt* with and without Chinese phonemes. Two Chinese characters, 赖 and 莱, were strong candidates and 莱 is the correct one in this case. Without Chinese phonemes, $M(E_G, \phi)$, which is based on P(cg|$Reinhardt$) in Figure 1(a) preferring 赖 to 莱, generated the incorrect transliteration as shown in Table 7. However, Figure 1(b) shows that 莱 can be selected if the correct Chinese phoneme sequence "LAI YIN ..." is given. Three Chinese phoneme sequences starting with "LAI YIN ...", "LAI NA ...", and "LAI NEI ..." were generated from *Reinhardt*, where "LAI YIN ..." was the best Chinese phoneme sequence based on the probability distribution in Figure 1(c). As a result, $M(E_G, JC_P)$, which jointly used Chinese phonemes with English graphemes, generated the correct Chinese transliteration of *Reinhardt* based on two probability distribution in Figures 1(b) and 1(c). In the case of *Ivy*, English phonemes contributed to generating the correct transliteration in the $M(E_{GP}, \phi)$ and $M(E_{GP}, JC_P)$ models.

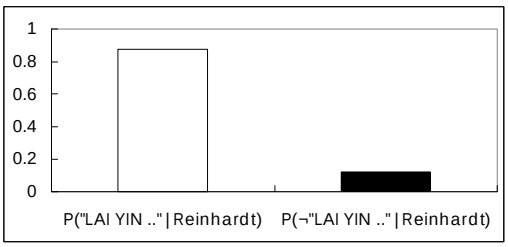
Chinese transliterations sometimes reflect the English word's pronunciation as well as the Chinese character's meaning (Li et al., 2007). Li et al. (2007) defined such a Chinese transliteration as a phonetic-semantic transliteration (semantic transliteration) to distinguish it from a usual phonetic transliteration. One fact that affects semantic transliteration is gender association (Li et al., 2007). For example, 莉 (meaning *jasmine*) is frequently used in Chinese transliterations of female names but seldom in common person names. Because *Emily* is often used in female names, the results obtained by the $M(E_G, JC_P)$ and $M(E_{GP}, JC_P)$ models are acceptable. This indicates that Chinese phonemes coupled with English graphemes or those coupled with English graphemes and phonemes could provide evidence required for semantic transliteration as well as phonetic transliteration. As a result, $M(E_{GP}, \phi)$, $M(E_G, JC_P)$,



(a) Probability distribution when Chinese phonemes are not given



(b) Probability distribution when Chinese phonemes are given

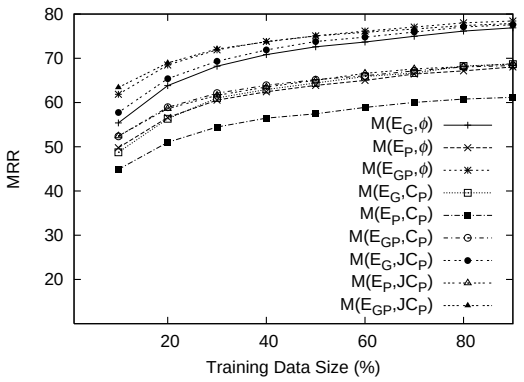


(c) Probability distribution for Chinese phoneme sequence "LAI YIN ..." and others

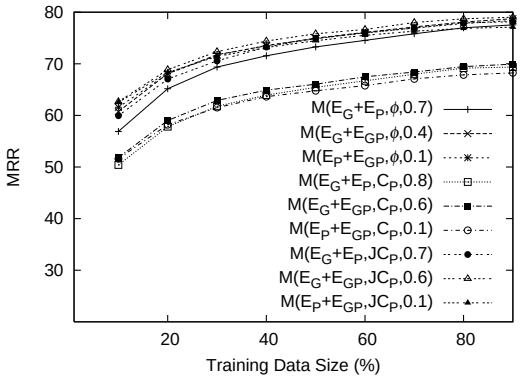
Figure 1: Probability distribution for the first Chinese character in the Chinese transliteration of *Reinhardt*: $M(E_G, \phi)$ vs. $M(E_G, JC_P)$

and $M(E_{GP}, JC_P)$, which used phonemes coupled with English graphemes, achieved higher performance than $M(E_G, \phi)$, which relied only on English graphemes.

4.2 Effect of Training Data Size



(a) Basic transliteration models



(b) Hybrid transliteration models

Figure 2: Performance of each system with different training data size

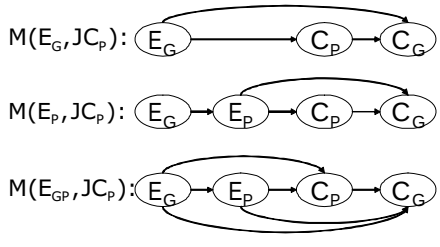
We investigated the effect of training data size on the performance of each transliteration model. We randomly selected training data with ratios from 10 to 90% and compared the performance of each system trained by different sizes of training data. The results for the basic transliteration models in Figure 2(a) can be categorized into three groups. $M(E_{GP}, \phi)$ and $M(E_{GP}, JC_P)$ fall into the best group, where they showed the best performance regardless of training data size. $M(E_G, \phi)$ and $M(E_G, JC_P)$ belong to the middle group, where they showed lower performance than the best group if the training data size is small, but their performance is comparable to the best group if the size of the training data is large enough. The others always showed lower performance than both the best and middle groups. Figure 2(b) shows that hybrid transliteration models, on average, were less sensitive to the training data size than the basic ones, because the two different basic transliteration models used in the hybrid ones boosted transliteration performance by complementing each other's weak points.

5 Conclusion

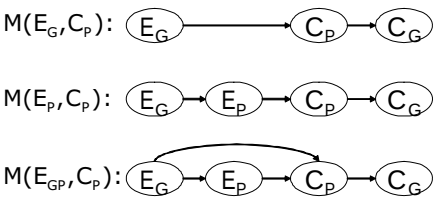
We proposed a new English-to-Chinese transliteration model based on Chinese phonemes and their corresponding English graphemes and phonemes. We defined eighteen English-to-Chinese transliteration models including our proposed model and classified them into three classes based on the role of Chinese phonemes in the transliteration models. Experiments showed that Chinese phonemes in our proposed model can contribute to the performance improvement in English-to-Chinese transliteration.

Now we can answer *Yes* to this paper's key question, "Can Chinese phonemes improve machine transliteration?" Actually, this is the second time the same question has been answered. The previous answer, which was unfortunately reported as *No* by Li et al. (2004), has been accepted as true for the last five years; the research issue has been considered closed. In this paper, we found a new answer that contradicts the previous answer. We hope that our answer promotes research on phoneme-based English-to-Chinese transliteration.

Appendix: Illustration of Basic Transliteration Models in $\mathbf{M_J}$ and $\mathbf{M_S}$



(a) $\mathbf{M_J}$ models



(b) $\mathbf{M_S}$ models

References

Y. Al-Onaizan and Kevin Knight. 2002. Translating named entities using monolingual and bilingual resources. In *Proc. of ACL '02*, pages 400–408.

A. L. Berger, S. D. Pietra, and V. J. D. Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

M. Chang, D. Goldwasser, D. Roth, and Y. Tu. 2009. Unsupervised constraint driven learning for transliteration discovery. In *Proceedings of NAACL HLT' 09*.

Wei Gao, Kam-Fai Wong, and Wai Lam. 2004. Phoneme-based transliteration of foreign names for OOV problem. In *Proc. of IJCNLP 2004*, pages 110–119.

Long Jiang, Ming Zhou, Lee-Feng Chien, and Cheng Niu. 2007. Named entity translation with web mining and transliteration. In *Proc. of IJCAI '07*, pages 1629–1634.

Paul B. Kantor and Ellen M. Voorhees. 2000. The trec-5 confusion track: Comparing retrieval methods for scanned text. *Information Retrieval*, 2:165–176.

Sarvnaz Karimi, Falk Scholer, and Andrew Turpin. 2007. Collapsed consonant and vowel models: New approaches for English-Persian transliteration and back-transliteration. In *Proceedings of ACL '07*, pages 648–655.

Chun-Jen Lee and Jason S. Chang. 2003. Acquisition of English-Chinese transliterated word pairs from parallel-aligned texts using a statistical machine transliteration model. In *Proc. of HLT-NAACL 2003 Workshop on Building and Using Parallel Texts*, pages 96–103.

Haizhou Li, Min Zhang, and Su Jian. 2004. A joint source-channel model for machine transliteration. In *Proceedings of the 42th Annual Meeting of the Association of Computational Linguistics*, pages 160–167.

Haizhou Li, Khe Chai Sim, Jin-Shea Kuo, and Minghui Dong. 2007. Semantic transliteration of personal names. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*.

Haizhou Li, A Kumaran, Min Zhang, and Vladimir Pervouchine. 2009. Whitepaper of NEWS 2009 machine transliteration shared task. In *Proc. of ACL-IJCNLP 2009 Named Entities Workshop*.

M.G. Abbas Malik. 2006. Punjabi machine transliteration. In *Proceedings of the COLING/ACL 2006*, pages 1137–1144.

H.M. Meng, Wai-Kit Lo, Berlin Chen, and K. Tang. 2001. Generating phonetic cognates to handle named entities in English-Chinese cross-language spoken document retrieval. In *Proc. of Automatic Speech Recognition and Understanding, 2001. ASRU '01*, pages 311–314.

Jong-Hoon Oh, Key-Sun Choi, and Hitoshi Isahara. 2006. A comparison of different machine transliteration models. *Journal of Artificial Intelligence Research (JAIR)*, 27:119–151.

Adwait Ratnaparkhi. 1997. A linear observed time statistical parser based on maximal entropy models. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 1–10.

Richard Schwartz and Yen-Lu Chow. 1990. The N-Best algorithm: an efficient procedure for finding top N sentence hypotheses. In *Proc. of ICASSP '90*, pages 81–84.

Tarek Sherif and Grzegorz Kondrak. 2007. Substring-based transliteration. In *Proceedings of ACL '07*, pages 944–951.

Paola Virga and Sanjeev Khudanpur. 2003. Transliteration of proper names in cross-lingual information retrieval. In *Proc. of ACL 2003 Workshop on Multilingual and Mixed-language Named Entity Recognition*, pages 57–64.

Stephen Wan and Cornelia Maria Verspoor. 1998. Automatic English-Chinese name transliteration for development of multilingual resources. In *Proc. of COLING '98*, pages 1352–1356.

Xinhua News Agency. 1992. *Chinese transliteration of foreign personal names*. The Commercial Press.

Binyong Yin and Mary Felley. 1990. *Chinese Romanization: Pronunciation and Orthography*. Sinolingua.

Su-Youn Yoon, Kyoung-Young Kim, and Richard Sproat. 2007. Multilingual transliteration using feature based phonetic method. In *Proceedings of ACL'07*, pages 112–119.