Computational Linguistics and Chinese Language Processing
Vol. 19, No. 3, September 2014, pp. 25-38

Transliteration Extraction from Classical Chinese Buddhist Literature Using Conditional Random Fields with Language Models

Yu-Chun Wang[*] , Karol Chia-Tien Chang[+] ,

Richard Tzong-Han Tsai[#] , and Jieh Hsiang[*]

Abstract

Extracting plausible transliterations from historical literature is a key issue in historical linguistics and other research fields. In Chinese historical literature, the characters used to transliterate the same loanword may vary because of different translation eras or different Chinese language preferences among translators. To assist historical linguists and digital humanities researchers, this paper proposes a transliteration extraction method based on the conditional random field method with features based on the language models and the characteristics of the Chinese characters used in transliterations. To evaluate our method, we compiled an evaluation set from two Buddhist texts, the Samyuktagama and the Lotus Sutra. We also constructed a baseline approach with a suffix array based extraction method and phonetic similarity measurement. Our method significantly outperforms the baseline approach, and the method achieves recall of 0.9561 and precision of 0.9444. The results show our method is very effective for extracting transliterations in classical Chinese texts.

Keywords: Ttransliteration Extraction, Classical Chinese, Buddhist Literation, Langauge Model, Conditional Random Fields, CRF.

[*] Department of Computer Science and Information Engineering, National Taiwan University, Taiwan
E-mail: d97023@csie.ntu.edu.tw; jhsiang@ntu.edu.tw

[+] Department of Computer Science and Engineering, Yuan Ze University, Taiwan
E-mail: s1003325@mail.yzu.edu.tw

[#] Department of Computer Science and Information Engineering, National Central University, Taiwan
E-mail: thtsai@csie.ncu.edu.tw
The author for correspondence is Richard Tzong-Han Tsai.

1. Introduction

Cognates and loanwords play important roles in the research of language origins and cultural interchange. Therefore, extracting plausible cognates or loanwords from historical literature is a key issue in historical linguistics. The adoption of loanwords from other languages is usually through transliteration. In Chinese historical literature, the characters used to transliterate the same loanword may vary because of different translation eras or different Chinese language/dialect preferences among translators. For example, in classical Chinese Buddhist scriptures, the translation process of Buddhist scriptures from Sanskrit to classical Chinese occurred mainly from the 1st century to 10th century. In these works, the same Sanskrit words may be transliterated into different Chinese loanword forms. For instance, the surname of the Buddha, Gautama, is transliterated into several different forms, such as "瞿曇" (qü-tan) or "喬答摩" (qiao-da-mo), and the name "Culapanthaka" has several different Chinese transliterations, such as "朱利槃特" (zhu-li-pan-te) and "周利槃陀伽" (zhou-li-pan-tuo-qie). In order to assist researchers in historical linguistics and other digital humanities research fields, an approach to extract transliterations in classical Chinese texts is necessary.

Many transliteration extraction methods require a bilingual parallel corpus or text documents containing two languages. For example, Sherif & Kondrak (2007) proposed a method for learning the string distance measurement function from a sentence-aligned English-Arabic parallel corpus to extract transliteration pairs. Kuo *et al*. (2007) proposed a transliteration pair extraction method using a phonetic similarity model. Their approach is based on the general rule that, when a new English term is transliterated into Chinese (in modern Chinese texts, *e.g*. newswire), the English source term usually appears alongside the transliteration. To exploit this pattern, they identify all of the English terms in a Chinese text and measure the phonetic similarity between those English terms and their surrounding Chinese terms, treating the pairs with the highest similarity as the true transliteration pairs. Despite its high accuracy, this approach cannot be applied to transliteration extraction in classical Chinese literature since the prerequisite (of the source terms alongside the transliteration) does not apply.

Some researchers have tried to extract transliterations from a single language corpus. Oh & Choi (2003) proposed a Korean transliteration identification method using a Hidden Markov Model (HMM) (Rabiner, 1989). They transformed the transliteration identification problem into a sequential tagging problem in which each Korean syllable block in a Korean sentence is tagged as either belonging to a transliteration or not. They compiled a human-tagged Korean corpus to train a hidden Markov model with predefined phonetic features to extract transliteration terms from sentences by sequential tagging. Goldberg & Elhadad (2008) proposed an unsupervised Hebrew transliteration extraction method. They

adopted an English-Hebrew phoneme mapping table to convert the English terms in a named entity lexicon into all of the possible Hebrew transliteration forms. The Hebrew transliterations then were used to train a Hebrew transliteration identification model. Nevertheless, Korean and Hebrew use an alphabetical writing system, while Chinese is ideographic. These identification methods heavily depend on the phonetic characteristics of the writing system. Since Chinese characters do not necessarily reflect actual pronunciation, these methods are difficult to apply to the transliteration extraction problem in classical Chinese.

This paper proposes an approach to extract transliterations automatically in classical Chinese texts, especially Buddhist scriptures, with supervised learning models based on the probability of the characters used in transliterations and the language model features of Chinese characters.

2. Method

To extract the transliterations from the classical Chinese Buddhist scriptures, we adopted a supervised learning method, the conditional random fields (CRF) model. The features we used in the CRF model are described in the following subsections.

2.1 Probability of each Chinese Character in Transliterations

According to our observation, in the classical Chinese Buddhist texts, the Chinese characters used in transliteration show some characteristics. Translators were inclined to choose characters without obstructing the comprehension of the sentences. Although the number of Chinese characters is large, the number of possible syllables in Chinese is limited. Therefore, one Chinese character may share the same pronunciation with several other characters, and a translator may choose rarely used characters for transliteration.

Thus, the probability of a Chinese character being used in transliteration is an important feature to identify transliteration in the classical Buddhist texts. In order to measure the probability of every character used in transliterations, we collected the frequency of all the Chinese characters in the Chinese Buddhist Canon. Then, we applied the suffix array method (Manzini & Ferragina, 2004) to extract the terms with their counts from all the texts of the Chinese Buddhist Canon. The extracted terms then were filtered through a list of selected transliteration terms from the Buddhist Translation Lexicon and Ding Fubao's Dictionary of Buddhist Studies. The extracted terms in the list were retained, and the frequency of each Chinese character was calculated. Thus, the probability of a given Chinese character c in transliteration can be defined as:

$$Prob(c) = \log \frac{freq_{trans}(c)}{freq_{all}(c)} \tag{1}$$

where $freq_{trans}(c)$ is c's frequency used in transliterations, and $freq_{all}(c)$ is c's frequency appearing in the entire Chinese Buddhist Canon. The logarithm in the formula is designed for CRF discrete feature values.

2.2 Character-based Language Model of the Transliteration

Transliterations may appear many times in one Buddhist sutra. The preceding character and the following character of the transliteration may be different. For example, for the phrase "於憍薩羅國" (yu-jiao-sa-luo-guo, "in Kosala state"), if we want to identify the actual transliteration, "憍薩羅" (jiao-sa-luo, Kosala), from the extra characters "於" (yu, in) and "國" (guo, state), we must first use an effective feature to identify the boundaries of the transliteration.

In order to do that, we propose a language-model-based feature. A language model assigns a probability to a sequence of m words $P(w_1,w_2,...,w_m)$ by means of a probability distribution. The probability of a sequence of m words can be transformed into a conditional probability:

$$\begin{aligned} &P(w_1, w_2, \cdots, w_m) = P(w_1)P(w_2 \mid w_1)P(w_3 \mid w_1, w_2) \cdots P(w_m \mid w_1, w_2, \cdots w_{m-1}) \\ &= \prod_{i=1}^{m} P(w_i \mid w_1, w_2, \cdots, w_{i-1}) \end{aligned} \tag{2}$$

In practice, we can assume that the probability of a word only depends on its previous word (bi-gram assumption). Therefore, the probability of a sequence can be approximated as:

$$P(w_1, w_2, \cdots, w_m) = \prod_{i=1}^{m} P(w_i \mid w_1, w_2, \cdots, w_{i-1}) \approx \prod_{i=1}^{m} P(w_i \mid w_{i-1}) \tag{3}$$

We collected person and location names from the Buddhist Authority Database and the known Buddhist transliteration terms from The Buddhist Translation Lexicon (翻譯名義集)[1] to create a dataset with 4,301 transliterations for our bi-gram language model. We used these transliterations to train the bi-gram language model. Such a language model may suffer from the sparse data problem. Nevertheless, since we adopted the language models as a feature for a supervised learning model, the sparse data problem is not serious in our approach.

After building the bi-gram language model, we applied it as a feature for the supervised model. Following the previous example, "於憍薩羅國" (yu-jiao-sa-luo-guo, "in Kosala state"), for each character in the sentence, we first computed the probability of the current

[1] http://www.cbeta.org/result/T54/T54n2131.htm

character and its previous character. For the first character “於”, since there is no previous word, the probability is P(於). For the second character “憍”, the probability of the two characters is P(於憍) = P(於)P(憍|於). We then computed the probability of the second and third characters: P(憍薩) = P(憍)P(薩|憍), and so on. If the probability changes sharply from that of the previous bi-gram, the previous bi-gram may be the boundary of the transliteration. Since the character “於” rarely appears in transliterations, P(於憍) is much lower than P(憍薩). We may conclude that the left boundary is between the first two characters “於憍”.

2.3 Pronunciation-based Language Model of the Transliteration

In addition to the character-based language model mentioned in the previous section, we also constructed a language model based on the pronunciations instead of characters. Since many Chinese characters may have the same pronunciation, different Chinese characters might be chosen to translate the same Sanskrit term. For example, the Sanskrit term, Arhat, has different Chinese transliteration forms, such as “阿羅訶” (a-luo-he) and “阿羅呵” (a-luo-he). The last Chinese characters (“訶” and “呵”) are different, but the Chinese pronunciations are the same. Therefore, a language model based on the pronunciation instead of Chinese character may overcome this kind of character variation problem. In order to construct a pronunciation based language model, the Chinese characters have to be converted into phoneme forms.

The pronunciation of Chinese characters, however, varies diachronically and synchronically. The same Chinese characters may have been pronounced differently in different regions and eras of ancient China. Therefore, we cannot base our language model on modern Chinese pronunciation. Furthermore, Chinese characters are ideographic and the pronunciations may not be reflected by the ideogram. Thus, it is difficult to find out how a character was pronounced in the past. In the seventh century (in the Sui dynasty), a new kind of pronunciation dictionary, *Qieyun*, was published by Lu Fayan, based on five earlier rhyming dictionaries that no longer exist. As a guide to the recitation of literary texts and an aid in the composition of verse, *Qieyun* quickly became popular and became the national standard of pronunciation during the ensuing Tang dynasty (618 - 907 C.E.). Unfortunately, the actual content of the *Qieyun* did not last into the modern era. In 1008, during the Northern Song Dynasty (960 - 1127 C.E.), a group of scholars commissioned by the emperor produced an expanded revision of *Qieyun* called *Guangyun*. Until the mid-20th century, the oldest complete rhyming book known was *Guangyun*, although existing copies are marred by numerous transcription errors. Thus, all studies of the rhyming book tradition were actually based on *Guangyun*. Since the period of the Buddhist literature translation from Sanskrit to classical Chinese is mainly from Tang dynasty to Song dynasty, which all belong to middle

Chinese era, we use *Guangyun* as an approximation to the pronunciation of Chinese characters in middle Chinese.

Since there were no phonological symbols or alphabetical writing systems in middle Chinese, rhyming books like Guangyun record contemporary character pronunciations with fanqie "反切" analyses. Fanqie represents a character's pronunciation with another two characters, combining the former's "initial" and the latter's "rhyme" and tone. An English equivalent would be to combine the initial of 'peek' /p^{h}i:k/ and the rhyme of 'cat' /kæt/ to get 'pat' /p^{h}æt/. Take the character "東" [tuŋ] for example. The fanqie of the character is "德紅" ([tok] and [huŋ]), so that we can get its pronunciation if we know the actual pronunciation of these two fanqie characters. Although fanqie is an effective method to represent the pronunciation of a Chinese character, it is still hard to analyze because the usage of two characters for the initial and rhyme is arbitrary. Fortunately, a revision of *Guangyun* included additional annotations by some scholars. They analyzed all the characters used in fanqie and categorized the homophones into groups and chose an identical Chinese character standing for the group. After the analysis, there are 36 initials and 106 rhymes in *Guangyun*. In addition to the initial and rhyme, there are other features added into the revision of Guangyun, such as fanqie, initial, rhyme, openness (round or unround), level (different medial vowels), and tone.

To employ the data from *Guangyun* in our analyses, we must first convert the Chinese characters it uses to represent pronunciation into International Phonetic Alphabet (IPA) notation. There are many researchers who have tried to reconstruct the actual pronunciations of the characters in middle Chinese. We use the reconstruction of middle Chinese pronunciation proposed by Wang Li for this task. Take the character "洪" for example. Its initial "匣" is converted to IPA [ɣ] and its rhyme "東" is converted to [uŋ], giving us a final reconstructed IPA phonemic form of [ɣuŋ].

All of the Chinese characters used to construct the language model are converted into middle Chinese IPA representations by *Guangyun*. Nevertheless, one Chinese character might have several pronunciations. The homographs create difficulty in converting the Chinese characters into their actual pronunciations. Nevertheless, there are few tools and resources for classical Chinese and middle Chinese to deal with this problem. Therefore, we use a heuristic method to determine the most used pronunciation for each Chinese character. We found that the *Kangxi Dictionary* (康熙字典) often gives the most used pronunciation first for each Chinese character. Therefore, if one Chinese character has several different fanqie pronunciations, we check the description of the character in the Kangxi Dictionary and find the first matched fanqie as the final pronunciation of the character. Take the character "解" for example. In *Guangyun*, the character "解" has two fanqie pronunciations: "佳買" and "胡買". The description of the character "解" in Kangxi Dictionary is "【唐韻】

【正韻】佳買切【集韻】【韻會】舉嶰切，𠂤皆上聲。【說文】判也。从刀判牛角。【莊子・養生主】庖丁解牛。【左傳・宣四年】宰夫解黿。【前漢・陳湯傳】支解人民。【註】謂解戡其四支也。…”. We can find the first fanqie is “佳買”. Therefore, we can determine the most used pronunciation of the character “解” is “佳買”, then convert it into IPA representation form [kai].

The feature value of the pronunciation-based language model is similar to the character-based language model described in Section 2.2. Following the previous example, “於憍薩羅國” (in Kosala state), we first convert the characters in the sentence into IPA representation, such as [ʔo kiu sat la kuok]. We then compute the probability of the current character and its previous character. For the first character “於” [ʔo], since there is no previous word, the probability is P(ʔo). For the second character “憍” [kiu], the probability of the two characters is P(ʔo kiu) = P(ʔo)P(kiu|ʔo). We then compute the probability of the second and third characters: P(kiu sat) = P(kiu)P(sat|kiu), and so on.

2.4 Functional Words

We take classical Chinese functional words into consideration. These characters have special grammatical functions in classical Chinese and are seldom used to transliterate foreign names. This is a binary feature that records the character as a functional word or not. The functional words are listed as follows: 之 (zhi), 乎 (hu), 且 (qie), 矣 (yi), 邪 (ye), 於 (yu), 哉 (zai), 相 (xiang), 遂 (sui), 嗟 (jie), 與 (yu), and 噫 (yi).

2.5 Appellation and Quantifier Words

After observing the transliterations appearing in classical Chinese literature, we note that there are some specific patterns in the characters following the transliteration terms. Most of the characters following the transliteration are appellation or quantifier words, such as 山 (san, mountain), 海 (hai, sea), 國 (guo, state), 洲 (zhou, continent). Examples are 耆闍崛山 (qi-du-jui-san, Vulture mountain), 拘薩羅國 (jü-sa-luo-guo, Kosala state), and 瞻部洲 (zhan-bu-zhou, Jambu continent). Therefore, we collect the Chinese characters that usually are used as appellation or quantifiers following transliterations and design this feature. This is a binary feature that records whether a character is used as an appellation or quantifier word or not.

2.6 CRF Model Training

We adopted the supervised learning models, conditional random field (CRF) (Lafferty *et al*., 2011), to extract the transliterations in classical Buddhist texts. For the CRF model, we formulated the transliteration extraction problem as a sequential tagging problem.

2.6.1 Conditional Random Fields

Conditional random fields (CRFs) are undirected graphical models trained to maximize a conditional probability (Lafferty *et al*., 2011). A linear-chain CRF with parameters $\Lambda = \lambda_1, \lambda_2, \ldots$ defines a conditional probability for a state sequence $\mathbf{y} = \mathbf{y}_1 \ldots \mathbf{y}_T$, given that an input sequence $\mathbf{x} = \mathbf{x}_1 \ldots \mathbf{x}_T$ is

$$P_\Lambda(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z_\mathbf{x}} \exp\left(\sum_{t=1}^{T} \sum_{k} \lambda_k f_k(\mathbf{y}_{t-1}, \mathbf{y}_t, \mathbf{x}, t) \right) \tag{4}$$

where Z_x is the normalization factor that makes the probability of all state sequences sum to one, $f_k(\mathbf{y}_{t-1}, \mathbf{y}_t, \mathbf{x}, t)$ is often a binary-valued feature function, and λ_k is its weight. The feature functions can measure any aspect of a state transition, $\mathbf{y}_{t-1} \rightarrow \mathbf{y}_t$, and the entire observation sequence, $\mathbf{x}$, centered at the current time step, t. For example, one feature function might have the value 1 when $\mathbf{y}_{t-1}$ is the state B, $\mathbf{y}_t$ is the state I, and $\mathbf{x}_t$ is the character "國" (guo). Large positive values for λ_k indicate a preference for such an event; large negative values make the event unlikely.

The most probable label sequence for $\mathbf{x}$,

$$\mathbf{y}^* = \arg\max_{y} P_\Lambda(\mathbf{y} \mid \mathbf{x}) \tag{5}$$

can be efficiently determined using the Viterbi algorithm.

2.6.2 Sequential Tagging and Feature Template

The classical Buddhist texts were separated into sentences by the Chinese punctuation. Then, each character in the sentences was taken as a data row for CRF model. We adopted the tagging approach motivated by the Chinese segmentation (Tsai *et al*., 2006) which treats Chinese segmentation as a tagging problem. The characters in a sentence are tagged in **B** class if it is the first character of a transliteration word or in **I** class if it is in a transliteration word but not the first character. The characters that do not belong to a transliteration word are tagged in **O** class. We adopted the CRF++ open source toolkit[2]. We trained our CRF models with the unigram and bigram features over the input Chinese character sequences. The features are shown as follows.

Unigram: $s_{-2}, s_{-1}, s_0, s_1, s_2$

Bigram: $s_{-1}s_0, s_0s_1$

[2] http://crfpp.googlecode.com

where the current substring is s_0 and s_i is other characters relative to the position of the current character.

3. Evaluation

3.1 Data Set

We chose Samyuktagama (雜阿含經), a Buddhist scripture from the Chinese Buddhist Canon maintained by Chinese Buddhist Electronic Text Association (CBETA), as our data set for evaluation. The Samyuktagama is one of the most important scriptures in Early Buddhism and contains a lot of transliterations because it records in detail the teachings and the lives of the Buddha and many of his disciples.

The Samyuktagama is an early Buddhist scripture collected shortly after the Buddha's death. The term agama in Buddhism refers to a collection of discourses, and the name Samyuktagama means "connected discourses". It is among the most important sutras in Early Buddhism. The authorship of the Samyuktagama is traditionally attributed to Mahakssyapa, Buddha's disciple, and five hundred Arhats three months after the Buddha's death. An Indian monk, Gunabhadra, translated this sutra into classical Chinese in the Liu Song dynasty around 443 C.E. The classical Chinese Samyuktagama has 50 volumes containing about 660,000 characters. As the amount of text in the Samyuktagama is immense, we took the first 20 volumes as the training set, and the last 10 volumes as the test set.

We also wanted to see whether the supervised learning model trained by one Buddhist scripture can be applied to another Buddhist scripture translated in a different era. Therefore, we chose another scripture, the Lotus Sutra (妙法蓮華經), to create another test set. The Lotus sutra is a famous Mahayana Buddhist scripture probably written down between 100 BC and 100 C.E. The earliest known Sanskrit title for the sutra is the Saddharma Pundarika Sutra, which translates to "the Good Dharma Lotus Flower Sutra". In English, the shortened form Lotus Sutra is common. The Lotus Sutra has been regarded highly in a number of Asian countries where Mahayana Buddhism traditionally has been practiced, such as China, Japan, and Korea. The Lotus Sutra has several classical Chinese translation versions. The most widely used version was translated by Kumarajiva ("鳩摩羅什" in Chinese) in 406 C.E. It has eight volumes and 28 chapters containing more than 25,000 characters. We selected the first 5 chapters as a different test set to evaluate our method.

3.2 Baseline Method

There are a few research projects focused on transliteration extraction from classical Chinese literature. Nevertheless, in order to compare and show the effectiveness of our method, we constructed a baseline system with widely used information extraction methods. Since many

previous research projects on transliteration extraction are based on phonetic similarity or phoneme mapping approaches, we also used these methods to construct the baseline system. First, the baseline system used the suffix array method to extract all the possible terms for the classical Chinese Buddhist scriptures. Then, the extracted terms were converted into Pinyin sequences by a modern Chinese pronunciation dictionary. We also adopted the collected transliteration list used in Section 2.1 and converted the transliterations into Pinyin sequences. Next, for each extracted term, the baseline system measured the Levenshtein distance between the Pinyin sequences of the extracted terms and all the transliterations as the phonetic similarity. If the extracted term had a Levenshtein distance less than the threshold (distance ≤ 3 in our baseline) from one of the transliterations we collected, the extracted term would be regarded as a transliteration; otherwise, the term would be dropped.

3.3 Evaluation Metrics

We used two evaluation metrics, recall and precision, to estimate the performance of our system. Recall and precision are widely used measurements in many research fields, such as information retrieval and information extraction (Manning *et al*., 2008). In digital humanities, a key issue is the coverage of the extraction method. To maximize usefulness to researchers, a method should be able to extract as many potential transliterations from literature as possible. Therefore, in our evaluation, we used recall, defined as follows:

$$Recall = \frac{|\text{Correctly extracted transliterations}|}{|\text{Transliterations in the data set}|} \tag{6}$$

In addition, the correctness of the extracted transliterations is also important. To avoid wasting time on useless information, a method should be able to extract correct transliterations from literature as much as possible. Thus, we also used precision, defined as follows:

$$Precision = \frac{|\text{Correctly extracted transliterations}|}{|\text{All extracted transliterations}|} \tag{7}$$

With precision and recall, the F-score measurement also was adopted as a weighted average of the precision and recall. The F1-score is defined as follows:

$$F_1\text{-score} = \frac{2 \times precision \times recall}{precision + recall} \tag{8}$$

3.4 Evaluation Results

Table 1 shows the results of our method with two different language models and the baseline system on different test sets. The gold standards of these two test sets were compiled by human experts who examined all of the sentences in the test sets and recognized each

transliteration for evaluation. The results show that our method with the character-based language model could extract 95.61% of the transliterations in the Sumyuktagama and 94.74% in the Lotus Sutra. On the precision measurement, our method also achieved pretty good results, which show that most of the terms our method extract are actual transliterations. The pronunciation-based language model does not perform well as the character-based one in both recall and precision metrics.

Table 1. Evaluation results of transliteration extraction.

	Data Set	Precision	Recall	F_1-score
Our Approach (character-based LM)	Samyuktagama	0.8810	0.9561	0.9170
	Lotus Sutra	0.9444	0.9474	0.9459
Our Approach (pronunciation-based LM)	Samyuktagama	0.8477	0.7530	0.7975
	Lotus Sutra	0.2081	0.6447	0.3146
Our Approach (character & pronunciation based LM)	Samyuktagama	0.8224	0.7349	0.7762
	Lotus Sutra	0.4581	0.7763	0.5762
Baseline	Samyuktagama	0.0399	0.7771	0.0759
	Lotus Sutra	0.0146	0.5789	0.2848

Our method outperforms the baseline system. The baseline system cannot extract most transliterations due to the limit of the suffix array method since the suffix array method only extracts the terms that appear twice or more often in the context. Furthermore, phonetic similarity is not effective to filter the transliterations, which causes the low precision performance of the baseline method. These results demonstrate that our method can save humanities researchers a lot of labor-intensive work in examining the transliteration.

4. Discussion

4.1 Effectiveness of Transliteration Extraction

Our method can extract many transliterations from the Samyuktagama, such as “迦毘羅衛” (jia-pi-luo-wei, *Kapilavastu*, the name of an ancient kingdom where the Buddha was born and raised), “尼拘律” (ni-jü-lü, *Nyagro*, the forest name in the Kapilavastu kingdom), and “摩伽陀” (muo-qie-tuo, *Magadha*, the name of an ancient Indian kingdom). These transliterations do not appear in the training set, but our method can still identify them. In addition, our method also discovered many transliterations in the Lotus Sutra that do not appear in the Samyuktagama, such as “娑伽羅” (suo-qie-luo, *Sagara*, the name of the king of the sea world in ancient Indian mythology), “鳩槃茶 / 鳩槃荼”

(jiu-pan-cha/jiu-pan-tu, *Kumbhanda*, one of a group of dwarfish, misshapen spirits among the lesser deities of Buddhist mythology), and “阿鞞跋致” (a-pi-ba-zhi, *Avaivart*, “not turn back” in Sanskrit). Since the characteristics of the Lotus Sutra are different from the Samyuktagama in many aspects, it shows that the supervised learning model trained by one Buddhist scripture may apply to other Buddhist scriptures translated in different eras and translators.

We have also discovered that transliterations may vary even in the same scripture. In the Samyuktagama, the Sanskrit term“Chandala” (someone who deals with the disposal of corpses and is a Hindu lower caste, formerly considered untouchable) has two different transliterations: “旃陀羅” (zhan-tuo-luo) and “栴陀羅” (zhan-tuo-luo).

The Sanskrit term “*Magadha*” (the name of an ancient Indian kingdom) has three different transliterations: “摩竭陀” (muo-jie-tuo), “摩竭提” (muo-jie-ti), and “摩伽陀” (muo-qie-tuo). The variations of the transliterations of the same word give clues of who the translators were and the progress of the translations. These variations may help the study of historical Chinese phonology and philology.

4.2 Comparison between Character-based and Pronunciation-based Language Models

From the evaluation results, we find that the pronunciation-based language model approach does not perform as well as the character-based one. Especially for the Lotus Sutra data set, the precision of the pronunciation-based approach drops sharply. Many non-transliteration candidates are extracted by the approach with the pronunciation-based language model, such as “逮得”, “何因”, “悅可”, “後必憂”, and “但離”. Since the pronunciation-based language model only considers the pronunciations instead of the actual characters and semantics, some terms that are not transliterations but have similar pronunciation patterns to those used in transliterations are extracted as false positives. The results also show that the supervised learning model with the pronunciation-based language model trained by the Samyuktagama does not predict well on other Buddhist literature, such as the Lotus Sutra. Since these two Buddhist works have many differences in content, the model that is only based on pronunciation cannot deal with the differences to get better results.

4.3 Error Cases

Although our method can extract and identify most transliteration pairs, some transliteration pairs cannot be identified. The error cases can be divided into several categories. The first one is that a few terms cannot be extracted, such as “闍維” (she-wei, *Jhapita*, cremation, a monk’s funeral pyre). This transliteration is seldom used and only appears three times in the final part of the Samyuktagama. The widely used transliteration of the term “*Jhapita*” is “荼

毘” (tu-pi). This may cause difficulty for the supervised learning model to identify these terms.

The other case is incorrect boundaries of the transliterations. Sometimes, our method may extract shorter terms, such as “韋提” (wei-ti, correct transliteration is “韋提希”, wei-ti-xi, *Vaidehi*, a female person name), “波羅” (po-luo, correct transliteration is “波羅柰”, po-luo-nai, *Varanasi*, a location name in Northern India), “瞿利摩羅” (qü-li-muo-luo, correct transliteration is “央瞿利摩羅”, yang-qü-li-muo-luo, *Angulimala*, one of the Buddha’s disciples). This problem is due to the probability generated by the language model. For example, the probability of the first two characters of the transliteration “央瞿利摩羅”, *P*(央瞿), is very low. This causes the CRF model to predict that the first character “央” (yang) does not belong to the transliteration. If more transliterations can be collected to build a better language model, this problem can be overcome.

In some cases, our method extracts longer terms, such as “阿那律陀夜” (a-na-lü-tuo-ye) while the correct transliteration is “阿那律陀”, (a-na-lü-tuo, *Aniruddha*, one of the Buddha’s closest disciples); and “兒富那婆藪” (er-fu-na-po-sou), while the correct transliteration is “富那婆藪” (fu-na-po-sou, *Punabbasu*, a kind of ghost in Buddhist mythology). In these cases, the preceding or following characters are often used in transliterations. There are cases where a transliteration is immediately followed by another transliteration. For example, our method extracts the term “闡陀舍利” (chan-tuo-she-li), which actually comprises two transliteration terms “闡陀” (chan-tuo, *Chanda*, one of the Buddhist’s disciples) and “舍利” (she-li, *Sarira*, Buddhist relics). It is difficult to separate them without any additional semantic clues. Although our method sometimes might extract incomplete transliterations with incorrect boundary, checking the boundary of a transliteration is not difficult for a human expert. Therefore, the extracted incorrect transliterations also have the benefit of helping humanities researchers quickly find and check plausible transliterations.

5. Conclusion

The transliteration extraction of foreign loanwords is an important task in research fields, such as historical linguistics and digital humanities. We propose an approach that can extract transliteration automatically from classical Chinese Buddhist scriptures. Our approach comprises the conditional random fields method with designed features that are suitable to identify transliteration characters based on language models and textual characteristics. The first feature is the probability of each Chinese character used in transliterations. The second feature is probability of the sequential bigram characters or phonemic representations measured by the language model method. In addition, functional words, appellation, and

quantifier words are regarded as binary features. The transliteration extraction problem is formulated as a sequential tagging problem, and the CRF method is used to train a model to extract the transliterations from the input classical Chinese sentences. To evaluate our method, we constructed an evaluation set from the two Buddhist texts, the Samyuktagama and the Lotus Sutra, which were translated into Chinese in different eras. We also constructed a baseline system with a suffix array based extraction method and phonetic similarity measurement for comparison. The recall of our method achieved 0.9561 and the precision was 0.9444. The results show our method outperforms the baseline system and is effective for extracting transliterations from classical Chinese texts. Our method can find the transliterations among the immense classical literature to help many research fields, such as historical linguistics and philology.

Reference

Goldberg, Y., & Elhadad, M. (2008). Identification of transliterated foreign words in hebrew script. *Computational Linguistics and Intelligent Text Processing*.

Kuo, J.-S., Li, H., & Yang, Y.-K. (2007). A phonetic similarity model for automatic extraction of transliteration pairs. *ACM Trans. Asian Language Information Processing*, *6*(2).

Lafferty, J., McCallum, A., & Pereira, F. (2011). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 29th Internation Conference on Machine Learning (ICML)*, 282-289.

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*, Cambridge University Press Cambridge.

Manzini, G., & Ferragina, P. (2004). Engineering a lightweight suffix array construction algorithm. *Algorithmica*, *40*(1), 33-50.

Oh, J., & Choi, K. (2003). A statistical model for automatic extraction of Korean transliterated foreign words. *International Journal of Computer Processing of Oriental Languages*, *16*(1), 41-62.

Rabiner, L. (1989). Tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the Institute of Electrical and Electronics Engineers (IEEE)*, 77.

Sherif, T., & Kondrak, G. (2007). Bootstrapping a stochastic transducer for Arabic-English transliteration extraction. In *Proceedings of Annual Meeting Asociation for Computational Linguistics*.

Tsai, R. T.-H., Hung, H.-C., Sung, C.-L., Dai, H.-J., & Hsu, W.-L. (2006). On closed task of chinese word segmentation:An improved CRF model coupled with character clustering and automatically generated template matching. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, 134-137.