GeoName: a system for back-transliterating pinyin place names

Kui Lam Kwok
Computer Science Dept., CUNY
Queens College, 65-30 Kissena Blvd.
Flushing, NY, 11367
kwok@ir.cs.qc.edu

Qiang Deng
Computer Science Dept., CUNY
Queens College, 65-30 Kissena Blvd.
Flushing, NY, 11367
peterqc@yahoo.com

Abstract

To be unambiguous about a Chinese geographic name represented in English text as Pinyin, one needs to recover the name in Chinese characters. We present our approach to this back-transliteration problem based on processes such as bilingual geographic name lookup, name suggestion using place name character and pair frequencies, and confirmation via a collection of monolingual names or the WWW. Evaluation shows that about 48% to 72% of the correct names can be recovered as the top candidate, and 82% to 86% within top ten, depending on the processes employed.

1 Introduction

Names referring to entities can be ambiguous because different entities may have been given the same name. When one encounters foreign place names within English texts, further complication arises because the English alphabet may not represent the native writing uniquely or adequately, and transliteration has to be employed. This is true for Chinese place names. In this information age, documents on Chinese events such as news stories, commentaries, reviews, analysis, can originate from various sources and languages other than Chinese. Authors may reference Chinese place names but not necessary accompany it with the actual Chinese characters. It is therefore useful to build an automatic algorithm to decode such a place name in English and map it to the original Chinese character representation.

Chinese language is written as a contiguous string of ideographs (characters) without white space. Geographic names of most cities, provinces, mountains, etc are two to four characters long. Border regions have longer place names. Unlike person names, there is not a preferred closed set for family name characters. Any of the over 6K GB-encoded character is theoretically admissible as part of a place name. When one refers to them in English text, one needs to represent them using English alphabets – a process of romanization. Two main systems exist for this process: Pinyin, official in Mainland China, and Wade-Giles convention, popular in Taiwan (see e.g. http://www.romanization.com). Their objective is to spell out the pronunciation of the Chinese characters with alphabets. Unfortunately, although written Chinese is by and large uniform (except for a few hundred characters that have simplified vs. traditional forms), spoken Chinese can vary from region to region with different dialects. The Pinyin system was introduced by the PRC government in the 1950's. It attempts to standardize the representation according to the official Beijing Potunghua dialect (Northern China Mandarin) for the whole country. The Wade-Giles system is an older convention designed by authors of the same names in the late 19th and early 20th century, and is popular in Taiwan and some parts of South-East Asia.

There are also other haphazard romanization conventions in different regions where Chinese is used. For example, Hong Kong has its own British colonial history and Southern (GuangDong) dialect, and entity names are often spelt differently. The representation 'Hong Kong' itself is neither Pinyin nor Wade-Giles. It should have been 'Xiang Gang' in the former, and 'Hsiang Kang' in the latter. This is also true for 'Singapore'. In this investigation, we will mainly focus on the Pinyin convention. This is used by most of the Chinese (PRC) and there has been discussion in Taiwan to adopt it even though there are still political obstacles around this issue. There is evidence that this system is gaining popularity in the U.S. as the default choice (Library of Congress 2000).

This paper investigates methods of recovering a Chinese place name in GB-encoding (the character codes used for simplified Chinese characters) when its English Pinyin is given. We have previously built a system PYName (Kwok and Deng 2002) to back-transliterate Chinese person names. This paper extends it to provide similar functionality for place names. It is a tool to help reduce ambiguity in cross language geographic entity reference, and would be useful for cross language information retrieval. The organization of this paper is as follows: Section 2 discusses some properties of Pinyin place names. Section 3 discusses the use of frequencies to help back-transliteration. Section 4 describes GeoName, our system to map English Pinyin place names to Chinese characters. Section 5 contains some evaluation of Geoname, and Section 6 contains our conclusion and future work.

2 Pinyin Place Names

The mapping from Chinese character to Pinyin is more or less unique because the majority of Chinese characters have only one sound (with some exceptions). Given a Pinyin, however, there can be many homophonic candidate characters depending on which sound it is. When one encounters such a Pinyin entity ambiguity can arise. Even if the context specifies the place precisely, there is still uncertainty as to its original character representation. This is true for all entity types rendered into Pinyin unless they are well known. As an example, the capital of China, Beijing, originates from the characters:

Bei; jing.

However, when back-transliterating from the English, the following are some of the possible mappings:

Bei { …}

Jing { …}

Candidates like: … are all possible place names in addition to the intended one. In fact, these two are highly fertile Pinyin: 'Bei' maps to 23 and 'jing' maps to 20, leading to a total of 460 possible pairs. Many of the pairs of course may not be used as place names.

It is possible to diminish the above ambiguity by capturing also the tone of a Pinyin character as is done in most Chinese input systems that accept Pinyin as input. The simplest convention has five tones. One tone can be assigned to each character represented as Pinyin, and this can separate the mapped characters into tonal sets. However, most printed or electronic texts such as newspapers or newswires do not have tones assigned. Our system assumes input texts have no tonal indication, and so can be adapted to online text processing.

Chinese place names are mostly two to three characters long. Four-character names exist and longer ones are possible. Unlike person names where the family name character is selected from a fairly closed set, character use is practically unrestricted for places. This means that when mapping a Pinyin representation into its original Chinese format, one can result in x^y candidates, where y is the average number of possible single character mappings for each of x syllables. To further complicate the issue, place names in Pinyin can be separated with white spaces or not. For example, the representation for , a place near Beijing, can be written as: 'Qin Huang Dao', 'QinHuangDao' or 'Qinhuangdao'. The first item shows the original character one by one separated by a white space. The second item is a composite Pinyin denoting that the three individual Pinyin should be treated as a single entity. Each individual Pinyin character however is initialized with a capital letter. The third item is like the second composite but without capital letter except for the first character. (For example, on 3/25/03, the New York Times reported a coalmine explosion at 'Mengnanzhuang' employing this style.) All three styles can be found in texts. The first two indicate unique segmentation of the Pinyin characters. The third style however presents the additional problem of segmentation: how to recover the characters correctly. The string 'Qinhuangdao' may be broken up as 'Qin huang dao', 'Qin huang da o', 'Qin hu ang dao', etc. because it so happens that the listed components -- call them syllables -- are all legitimate Pinyin. Thus, the 'Qinhuangdao' composite can be either a three-, four- or five-character entity. One can imagine the exponential increase in candidates if each Pinyin syllable maps back to ~10 possibilities, for example. There is a fourth style that employs an apostrophe to indicate syllable separation in case of extreme ambiguity such as: Xian (province) and Xi'an (the city). This is very useful, like style one or two. Unfortunately, none of these is mandatory.

3 Mapping Pinyin to Chinese Character

Back-transliteration is a difficult problem as exemplified in (Knight and Graehl 1997, Chen, et.al. 1998). We limit ourselves to text input of a place name. Let $E = e_1\ e_2\ e_3\ ..\ e_N$ be a given English place name with Pinyin syllables e_k, $1<=k<=N$. It may have originated from a Chinese character sequence $C = c_1\ c_2\ c_3..\ c_N$ with probability: $P(C|E) = P(E|C)*P(C)/P(E)$. $P(E)$ can be ignored, and $P(E|C)$ is reduced to a product of $P(e_k|c_k)$ if independence of e_k with e_j, and e_k with c_j $(j•k)$ are assumed. Since most Chinese characters have unique Pinyin, we also set $P(e_k|c_k)$ to a constant, leaving the unknown $P(C)$. If one has sufficient bilingual translation of place names, the neglected probability $P(e_k|c_k)$ can be estimated.

Hence $P(C|E)$ is roughly reduced to $P(C)$ up to a constant. The most probable Chinese character sequence corresponding to the input Pinyin E is therefore equal to the one argmax $P(C)$, or $P(C)$ can be used to rank candidates C. To estimate $P(C)$, we initially used a bigram model: $P(C)\sim P(c_1)*P(c_2|c_1)*P(c_3|c_2)..*P(c_N|c_{N-1})$ which turns out to be less effective than the following heuristic approach. Instead of probability, we work with occurrence frequencies of the string itself, bigrams, and single characters. The function for ranking is

$$g(C) = a_1*\log\ [f(C)+a_1] + \Sigma\ a_2*\log\ [f(c_ic_j)+a_2] + \Sigma a_3*\log\ [f(c_i)+a_3] \quad (1)$$

where f(.) is frequency, and the sums run over all consecutive bigrams and singles composing the string C, and a_i, i=1,..3 are constants, which are larger for longer strings. A factor is not counted if its f(.) is zero. When string C has been seen before, its effect is larger if the length of C is longer. If C does not exist, its component bigram and single character frequencies determine the ranking value g(C). It is generally true that for a character string matching some dictionary entries or previous use, the longer the length, the more legitimate it is.

The issues raised in Section 2 are addressed as follows. Many Pinyin of the third style do lead to unique segmentation. For those that do not, all possible segmentations are captured, but they are sorted longest spelling sequence (and minimum syllables) first: e.g. in the previous example, ‘Qin huang dao’ is preferred over ‘Qin hu ang dao’. The candidates (c_1 c_2 c_3.. c_N) for Eqn.1 are limited to all possible combinations of characters that exist in the training data and can be mapped from the segmented Pinyin. Because of limitation of hardware, our prototype currently limits the number of Pinyin syllables to four in order to cut down on the number of candidates for certain input.

4 GeoName

GeoName is designed to accept a Pinyin place name and suggest Chinese GB-encoded candidates for it. Back-transliteration is an ambiguous and inaccurate process. Also, non-standard romanization exists historically for many common places names. The system does not yet have the capability to extract such names from running text, but requires that each name be entered on a separate line. Each Pinyin name is subjected to segmentation and character mapping, and a set of candidate GB-encoded Chinese names is produced as discussed in Section 2 and 3. GeoName employs a three-step process to effect back-transliteration: 1) table lookup on a bilingual place name list; 2) suggest names based on frequency usage of place characters and pairs; 3) confirmation via web retrieval or a monolingual geographic list. The following sub-sections present details of our approach.

4.1 Bilingual Place Name List

Geographic entities tend not to change much over time, and the number of places is relatively fixed, unlike person name for example. Thus, it is a good strategy to produce a lookup table to map place names between Chinese and English. It will give accurate translation; it can handle 1:m mappings well when a Chinese name may be represented differently due to different systems of romanization, and is very efficient in real time computation. The disadvantages are that it is difficult to locate such a bi-list, it will not be complete, relatively fixed, and it cannot suggest possible new names that are not on the list. We think such a list is an important component of any system that tries to do this kind of mappings, as there would always be many well-known places that have non-standard or peculiar romanization. From ftp://ftpserver.ciesin.columbia.edu/pub/data/China /CITAS/gb_code/ we located such a bi-list that contains about 4K unique Chinese place names. This we call List-A. Using the English Pinyin as key, a direct hit on this list will provide most probably the correct translation for the input. The first bit (A-bit) of a 3-bit tag would be set to 1, thus 100. The tag is attached to each candidate.

4.2 Place Name Suggestion

The total number of GB-encoded characters is about 6,000, but around 2,500 are the most often used. Since we limit our domain to geographical names here, we can collect such names in monolingual Chinese text and estimate the probabilities for single and paired Chinese characters use in this context. We employed similar methods in our PYName system for person names and it worked reasonably well. However, unlike person names where many people may share the same name characters, geographic names tend to be relatively more unique, i.e. not too many places have similar characters in our training data. Thus, the effectiveness of using frequency to suggest GB-encoded place names based on a given Pinyin name in English is more limited. This is compounded by the difficulty of finding a sufficiently large name list. The main advantage of the probabilistic mapping exercise is to be able to suggest new names as candidates by composing with characters, and rank them according to how characters appear in the monolingual name list as discussed in Section 3.

The ranking formula in Eqn.(1) has to be estimated from some training data. We failed to find sufficient downloadable Chinese place names and employed BBN's IdentiFinder (Miller, et.al. 1999) that brackets location entities in running text. The collections used are from the TREC and NTCIR experiments. Location names were identified and extracted. The result is about 80K “approximate place names” called List-B. The software is not perfect and many entries are not place names, or contain several names together. But the data can still serve its purpose.

4.3 Name Confirmation

To improve the accuracy of candidate ranking obtained in Section 4.2, we further use a process of confirmation. The hypothesis is that if a GB-encoded place name candidate has been seen before, it has a high probability of being correct. Each candidate name is compared to the monolingual Chinese name list consisting of (List-A U List-B). If it exists, the second bit (B-bit) of the 3-bit tag is set giving 010.

However, as suggested before, name lists are seldom complete. To mitigate this problem, we also utilize the Word Wide Web for confirmation. The basic idea is to treat WWW as another name collection, but a dynamic one. The English Pinyin name is treated as a query and sent to a search engine (such as Google). By using the advanced search option to return GB-encoded documents, each candidate of the Pinyin is searched in the documents to confirm whether it has been used as a sub-string. If true, the third C-bit of the tag is set giving 001. Another benefit of using the WWW is to resolve some dialect-based problems. As an example, both 'Hong Kong' and 'Xiang Gang' as Pinyin place names have been found on web documents with the Chinese name

confirmed. However, we do have to pay a price on performance, since web searches are relatively slow. Another draw back is that, web confirmation is effective only on popular, well-known names. Otherwise, domain specific name lists can be used if available.

Thus, all candidates are tagged and rank value assigned. Our current strategy is to rank candidates by the 3-bit tag first, followed by minimum syllable number, and then by g(C) of Eqn.1. If a candidate is confirmed somewhere, especially on our bi-list, it will be a good translation. Otherwise, shorter names are preferred.

4.4 System Description

Fig.1 below is a flowchart of GeoName showing how the different functions are tied together. Steps 2, 5 and 6 for bi-list lookup and confirmation can be enabled or disabled. Although our main focus is on Pinyin input, GeoName does have limited support in Step 3 for other

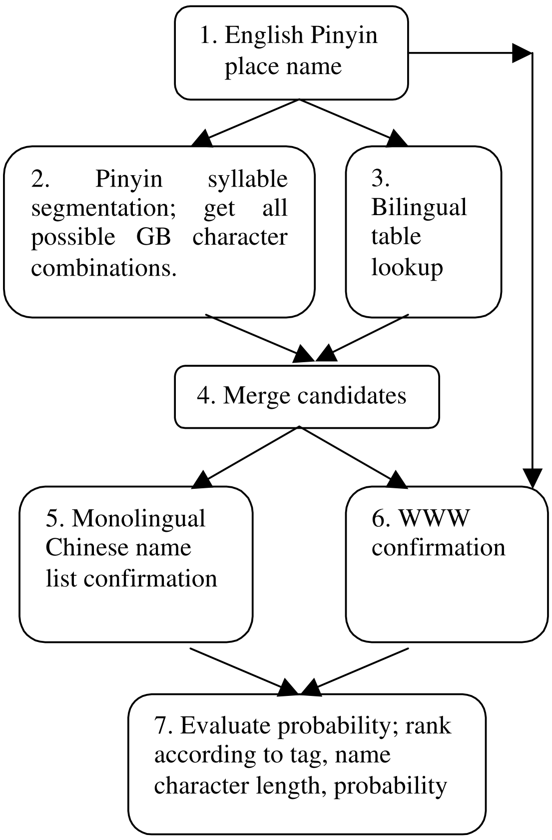

Fig.1. GeoName System Flowchart

romanization systems such as Wade-Giles and Hong Kong Pinyin. The system allows selection if the input romanization convention is known. A table converts Wade-Giles spelling into PRC Pinyin. For Hong Kong style spelling, another table converts it directly into GB character. Example back-transliterations are shown on the GUI screen of GeoName in Fig.2. The 1^{st} and 3^{rd} names are correct at rank 1, the 2^{nd} at rank 2.

5 Evaluation of GeoName

To evaluate the performance of GeoName, we need to test a set of Chinese place names in English Pinyin and compare the output from GeoName with the known Chinese characters for each name. In essence, we need another bi-list for testing, independent of the List-A that we used for training. Bilingual lists are difficult to obtain. Eventually a bilingual map (Map of Peoples' Republic of China 2001) with both Chinese and English names printed was located. The test set consists of 162 non-capital city names randomly selected from the map, six from each of the twenty-seven provinces excluding Taiwan (where some names are in Wade-Giles convention). The rank position of the correct Chinese name for each Pinyin returned from GeoName was noted within top ten; else it is considered a failure. We tested four settings of the tag values, viz.: 000 (only frequency prediction), 001 (frequency and web confirmation), 010 (frequency and monolingual list confirmation), and 111 (full function). A tabulation of the number of correct names found vs. rank position is shown in Table 1.

Rank 1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**> 10**
tag = 000										
78	22	13	3	4	4	6	1	1	1	29
tag = 001										
95	20	7	7	2	1	2	1	0	2	25
tag = 010										
88	24	11	1	4	2	3	0	0	1	28
tag = 111										
116	10	2	3	3	1	2	2	0	1	22

Table1: Number of Correct Candidates in Top Ten

The result with tag=000 (using frequency only for candidate suggestion) shows that 78 candidates out of 162 (48%) are correct at rank 1, and 133 (82%) correct within top 10. Both runs with tag=001 (add WWW confirmation) or tag=010 (add monolingual List-A U List-B confirmation) improves over tag=000 results, especially at rank 1, bringing this percentage to 59% and 54% respectively. Web confirmation is expensive in processing time, and may be variable depending on the state of the Web. Monolingual list confirmation is useful, especially when one has a list that is more region-specific to the desired input names. The best result is returned when all the processes are employed including checking on the bilingual List-A. Apparently many of our input names appear on this list, and it leads to simple table-lookup for the back-transliteration. This is probably not surprising because the bilingual map is not large (2'x3'), and it would only show the more well- known cities. Thus for the tag=111 run, it is seen that the correct candidates at rank 1 increase to 116 (71.6%), and if up to rank 10 candidates are included, 140 (86.4%) of the correct names are identified.

Conclusion

We have described GeoName, a system to back-transliterate English Pinyin geographic names to Chinese characters based on bilingual list lookup, monolingual place name character frequency, and Web confirmation. Evaluation using Pinyin city names shows that nearly 72% of the names suggested are correct at rank 1, and over 86% of correct names are included in the top ten candidates.

The evaluation is small involving only 162 city names. One needs larger scale studies with more obscure names or names actually in use. The resources we employed are rather limited. We intend to improve our training data, as well as our formula for name suggestion. Bilingual resources are difficult to locate. We are exploring how to use the Web as a gigantic bilingual name list in order to improve our system further.

Acknowledgments

We like to thank Beth Sundheim for suggesting the problem and pointing out some geographic resources to us, and BBN for the use of their IdentiFinder software. This work was partially sponsored by the Space and Naval Warfare Systems Center San Diego, under Grant No. N66001-00-1-8912.

References

Chen, H.H., Huang, S-J., Ding, Y-W. and Tsai, S-C. (1998) *Proper name translation in cross-language information retrieval.* Proceedings of COLING-ACL98. pp.232-236.

Knight, K. and Graehl, J. (1997) *Machine transliteration.* Proceedings of 35th Annual Meeting of Association for Computational Linguistics, pp. 128-135.

Kwok, K.L. and Deng, Q. (2002) *Corpus-based Pinyin Name Resolution*. Proceedings of the First SIGHAN Workshop on Chinese Language Processing (COLING 2002). pp. 41-47.

Library of Congress Pinyin Conversion Project (2000). http://www.loc.gov/catdir/pinyin/outline.html.

Map of the Peoples' Republic of China. (2001) ISBN7-80544-601-6/K.573. Chengdu Cartographic Publishing House. (http://www.ccph-map.com)

Miller, D., Schwartz, R., Weischedel, R. and Stone, R. (1999) *Named Entity Extraction from Broadcast News*. Proceedings of DARPA Broadcst News Workshop. pp. 37-40.

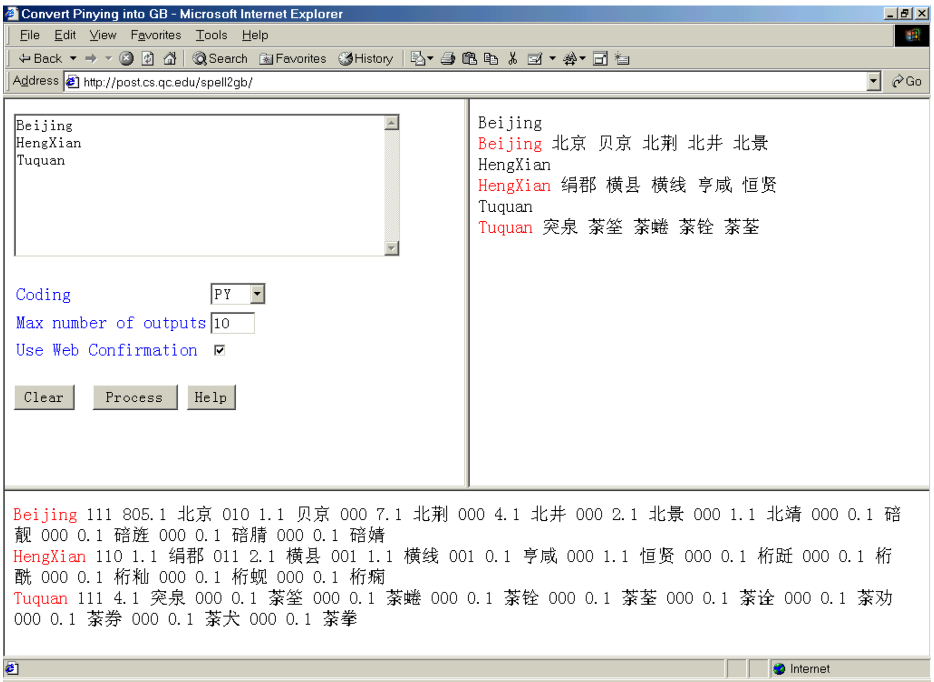

Fig.2: GUI of GeoName