Transliteration of Proper Names in Cross-Language Applications*

Paola Virga
Johns Hopkins University
3400 North Charles Street
Baltimore, MD 21218, USA
paola@jhu.edu

Sanjeev Khudanpur
Johns Hopkins University
3400 North Charles Street
Baltimore, MD 21218, USA
khudanpur@jhu.edu

ABSTRACT

Translation of proper names is generally recognized as a significant problem in many multi-lingual text and speech processing applications. Even when large bilingual lexicons used for machine translation (MT) and cross-lingual information retrieval (CLIR) provide significant coverage of the words encountered in the text, a significant portion of the tokens not covered by such lexicons are proper names (cf e.g. [3]). For CLIR applications in particular, proper names and technical terms are particularly important, as they carry some of the more distinctive information in a query. In IR systems where users provide very short queries (e.g. 2-3 words), their importance grows even further.

Proper names are amenable to a speech-inspired translation approach. When writing a foreign name in ones native language, one tries to preserve the way it sounds. i.e. one uses an orthographic representation which, when "read aloud" by a native speaker of the language sounds as it would when spoken by a speaker of the foreign language — a process referred to as *transliteration*. If mechanisms were available (a) to render, say, an English name in its phonemic form, and (b) to convert this phonemic string into the orthography of, say, Mandarin Chinese, then one would have a mechanism for transliterating English names using Chinese characters. The first part has been addressed extensively in the automatic text-to-speech synthesis literature. This paper describes a statistical approach for the second part.

Several techniques have been proposed in the recent past for name transliteration. Finite state transducers that implement transformation rules for *back*-transliteration from Japanese to English are described in [2], and extended to Arabic in [5]. In both cases, the goal is to recognize words in Japanese or Arabic text which happen to be transliterations of English names. The strongly phonetic orthography of Korean is exploited in [1] to obtain good transliteration using relatively simple HMM-based models. A set of handcrafted rules for *locally* editing the phonemic spelling of an English name to conform to Mandarin syllabification is provided to a transformation-based learning algorithm in [4], which then learns how to convert an English phoneme sequence to a Mandarin syllable sequence. We describe here a fully data driven counterpart to the technique of [4] for English-to-Mandarin name transliteration.

In addition to *intrinsic* evaluation, we test our transliteration system *extrinsically* for cross-lingual spoken document retrieval by using English text queries to retrieve Mandarin audio from the Topic Detection and Tracking (TDT) corpus.

*This research was partially supported by DARPA via Grant No N66001-00-2-8910 and ONR via Grant No N00014-01-1-0685.

Copyright is held by the author/owner.
SIGIR'03, July 28–August 1, 2003, Toronto, Canada.
ACM 1-58113-646-3/03/0007.

Categories and Subject Descriptors

H.3.4 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Query formulation*

General Terms

Algorithms, Design and Performance

1. TRANSLITERATION SYSTEM

We break down the transliteration process into four steps as depicted in Figure 1: (i) converting an English name into a phonemic representation using the Festival speech synthesis system, (ii) translating the English phoneme sequence into a sequence of initials and finals – standard sub-syllabic units for expressing pronunciations of Chinese characters, (iii) transformating the initial/final sequence into a sequence of pin-yin symbols, and (iv) translating the pin-yin sequence to a character sequence. Steps (i) and (iii) are *deterministic* transformations, and steps (ii) and (iv) are accomplished using *statistical means*.

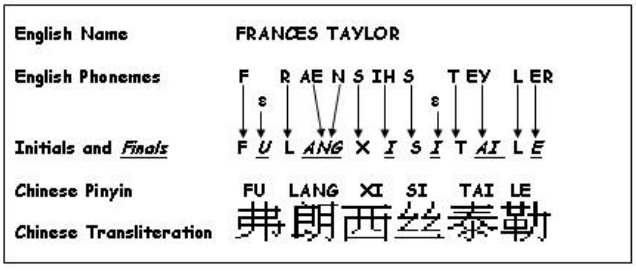

Figure 1: Four Steps in English-Chinese Transliteration.

Since the IBM source-channel model for statistical machine translation will play a vital role in our system, we describe it very briefly here for completeness. If one is translating English sentences to Chinese using this model, a I-word English sentence $\mathbf{e} = e_1 \ldots e_I$ is thought of as the output of a noisy "channel" whose input is its correct J-word Chinese translation $\mathbf{c} = c_1 \ldots c_J$. Having observed the channel output $\mathbf{e}$, one seeks *a posteriori* the most likely Chinese sentence

$$\hat{\mathbf{c}} = \arg\max_{\mathbf{c}} P(\mathbf{c}|\mathbf{e}) = \arg\max_{\mathbf{c}} P(\mathbf{e}|\mathbf{c})P(\mathbf{c})$$

The *translation model* $P(\mathbf{e}|\mathbf{c})$ is estimated from a paired corpus of English sentences and their Chinese translations, and the *language model* $P(\mathbf{c})$ is trained from Chinese text.

Since we seek Chinese names which are transliteration of a given English name, the notion of words in a sentence in the IBM model

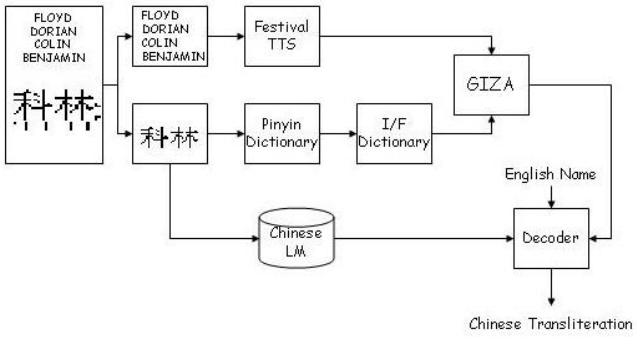

Figure 2: English-to-Chinese Name Transliteration System.

above is replaced with phonemes in a word. In our case, $\mathbf{e}$ is an English name, with $e_1\ e_2\ \ldots\ e_I$ beign its phonemic pronunciation, and $c_1\ c_2\ \ldots\ c_J$ is the string of sub-syllabic units (initials and finals) corresponding to the Chinese transliteration $\mathbf{c}$ of $\mathbf{e}$. A block diagram showing how such a system is trained and utilized is depicted in Figure 2.

Transliteration Model Training : We have available from the authors of [3] a small corpus of about 3875 English names and their Chinese transliteration. A pin-yin rendering of the Chinese transliteration is also provided. We use the Festival system to obtain a phonemic pronunciation of each English name. Finally, the pronunciation of a pin-yin symbol using an inventory of initials and finals is obtained from an elementary text on Mandarin phonology. The net result is a corpus of 3875 pairs of "sentences" of the kind depicted in the second and third lines of Fig 1. A second corpus of 3875 "sentence" pairs is derived corresponding to the fourth and fifth lines of Fig 1, this time to train a statistical model to translate pin-yin sequences to Chinese characters.

Training for Intrinsic Evaluation (MT-small): For purposes of comparing with the transliteration accuracy reported in [4], we divide our corpus into 2584 name-pairs for training the system and 1291 name-pairs for testing its intrinsic transliteration performance.

Training for Extrinsic Evaluation (MT-big): For subsequent CLIR experiments, we create a larger training set of 3625 name-pairs, leaving out only a small set of 250 names-pairs for testing.

The actual training of all translation models proceeds according to a standard recipe recommended in the GIZA++ statistical MT toolkit, namely 5 iterations of Model 1, followed by 5 of Model 2, 10 HMM-iterations and 10 iterations of Model 4.

Intrinsic Evaluation of Transliteration: We evaluate the efficacy of our transliteration at two levels. For comparison with the set-up in [4], we measure the accuracy of the pin-yin output produced by our system after Step (iii) mentioned above. The results are shown in Table 1, where pin-yin error rate (PER) is the edit distance between the "correct" pin-yin representation of the correct transliteration and the system output. Note that the performance of our fully statistical method is quite competitive with previously known results. We further note that increasing the training data results in further reduction of the syllable error rate. We remark that

System	Training	Test	PER	CER
H. Meng [4]	2233	1541	52.5	NA
MT-small	2233	1541	50.8	57.4
MT-big	3650	250	49.1	57.4

Table 1: Pinyin and Character Error Rates

this performance, while comparable to other systems, is not very satisfactory and merits further investigation.

We also evaluate the efficacy of our second translation system which maps the pin-yin sequence produced by the previous stage to a sequence of Chinese characters, and obtain a character error rate (CER) of 12.6%.

2. SPOKEN DOCUMENT RETRIEVAL

We build upon the experimental infrastructure developed at the 2000 Johns Hopkins Summer Workshop [3] where considerable work was done towards indexing and retrieving Mandarin audio to match English text queries. Specifically, we find that in a large number of queries used in those experiments, English proper names are not available in the translation lexicon, and are subsequently ignored during retrieval. We use the technique described above to transliterate all such names into Chinese characters and observe the effect on retrieval performance.

The TDT-2 corpus, which we use for our experiments, contains 2265 audio clips of Mandarin news, along with several thousand contemporaneously published Chinese text articles, and English text and audio broadcasts. The articles tend to be several hundred to a few thousand words long, while the audio clips tend to be two minutes or less on average. Exhaustive relevance judgments are provided for several topics in TDT-2. i.e. for each of at least 17 topics, every English and Chinese article and news clip has been examined by a human assessor and determined to be either on- or off-topic. We use a randomly selected *English article* on each of the 17 topics as a query, and wish to retrieve all the *Mandarin audio clips* on the same topic without retrieving any that are off-topic. We use the query term-selection and translation technique described in [3] to convert the English document to Chinese, the only augmentation being the transliterated names — there are roughly 3000 tokens in the queries which were not translatable in [3], and almost all of them are proper names. We report IR performance with and without the name-transliteration.

Retrieval Performance: We indexed the ASR transcriptions of the TDT-2 Mandarin audio using the HAIRCUT system. Using a Mandarin text document as a query, we obtained a monolingual baseline mean average precision (mAP) of 0.701 for word-based indexing and 0.762 for character-bigram indexing. The difference is statistically significant at a p-value of 0.014. We then used English documents as queries, performed query translation, and retrieved using character-bigram indexing. The mAP improved from 0.501 to 0.515 by adding name-transliteration ($p = 0.084$).

Ongoing Work: We have recently received a large list of nearly 2M Chinese-English named-entity pairs from the LDC. As a pilot experiment, we simply "translated" those names in our English queries which happened to be available in this LDC list. The mAP improvement from 0.501 to 0.506 was insignificant ($p = 0.421$). We are investigating the cause for this disappointing improvement, and are also training a better transliteration model from this corpus.

3. REFERENCES

[1] S. Y. Jung and al. An english to korean transliteration model of extended markov window. In *COLING*, 2000.

[2] K. Knight and J. Graehl. Machine transliteration. In *ACL*, 1997.

[3] H. Meng and al. Mandarin-english information (mei): Investigating translingual speech retrieval. Technical report, http://www.clsp.jhu.edu/ws2000/groups/mei.

[4] H. Meng and al. Generating phonetic cognates to handle named entities in english-chinese cross-language spoken document retrieval. In *ASRU*, 2001.

[5] B. G. Stalls and K. Knight. Translating names and technical terms in arabic text. In *COLING/ACL Workshop on Computational Approaches to Semitic Languages*, 1998.