Unsupervised Named Entity Transliteration Using Temporal and Phonetic Correlation

Tao Tao, Su-Youn Yoon, Andrew Fister, Richard Sproat and ChengXiang Zhai
University of Illinois at Urbana-Champaign
{syoon9,afister2,rws}@uiuc.edu, {taotao,czhai}@cs.uiuc.edu

Abstract

In this paper we investigate *unsupervised* name transliteration using *comparable corpora*, corpora where texts in the two languages deal in some of the same topics — and therefore share references to named entities — but are not translations of each other. We present two distinct methods for transliteration, one approach using an unsupervised phonetic transliteration method, and the other using the temporal distribution of candidate pairs. Each of these approaches works quite well, but by combining the approaches one can achieve even better results. We believe that the novelty of our approach lies in the phonetic-based scoring method, which is based on a combination of carefully crafted phonetic features, and empirical results from the pronunciation errors of second-language learners of English. Unlike previous approaches to transliteration, this method can in principle work with any pair of languages in the absence of a training dictionary, provided one has an estimate of the pronunciation of words in text.

1 Introduction

As a part of a on-going project on multilingual named entity identification, we investigate unsupervised methods for transliteration across languages that use different scripts. Starting from paired comparable texts that are about the same topic, but are not in general translations of each other, we aim to find the transliteration correspondences of the paired languages. For example, if there were an English and Arabic newspaper on the same day, each of the newspapers would likely contain articles about the same important international events. From these comparable articles across the two languages, the same named entities such as persons and locations would likely be found. For at least some of the English named entities, we would therefore expect to find Arabic equivalents, many of which would in fact be transliterations.

The characteristics of transliteration differ according to the languages involved. In particular, the exact transliteration of say, an English name is highly dependent on the language since this will be influenced by the difference in the phonological systems of the language pairs. In order to show the reliability of a multi-lingual transliteration model, it should be tested with a variety of different languages. We have tested our transliteration methods with three unrelated target languages — Arabic, Chinese and Hindi, and a common source language — English. Transliteration from English to Arabic and Chinese is complicated (Al-Onaizan and Knight, 2002). For example, while Arabic orthography has a conventional way of writing long vowels using selected consonant symbols — basically <w>, <y> and <?>, in ordinary text short vowels are rarely written. When transliterating English names there is the option of representing the vowels as either short (i.e. unwritten) or long (i.e. written with one of the above three mentioned consonant symbols). For example *London* is transliterated as لندن *lndn*, with no vowels; *Washington* often as وَاشنطون *wSnjTwn*, with <w> representing the final <o>. Transliterations in Chinese are very different from the original English pronunciation due to the limited syllable structure and phoneme inventory of Chinese. For example, Chinese does not allow consonant clusters or coda consonants except [n, N], and this results in deletion, substitution of consonants or insertion of vowels. Thus while a syllable initial /d/ may surface as in *Bagh**d**ad* 巴格达 *ba-ge-da*, note that the syllable final /d/ is not represented.

Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006), pages 250–257,
Sydney, July 2006.

Hindi transliteration is not well-studied, but it is in principle easier than Arabic and Chinese since Hindi phonotactics is much more similar to that of English.

2 Previous Work

Named entity transliteration is the problem of producing, for a name in a source language, a set of one or more transliteration candidates in a target language. Previous work — e.g. (Knight and Graehl, 1998; Meng et al., 2001; Al-Onaizan and Knight, 2002; Gao et al., 2004) — has mostly assumed that one has a training lexicon of transliteration pairs, from which one can learn a model, often a source-channel or MaxEnt-based model.

Comparable corpora have been studied extensively in the literature — e.g.,(Fung, 1995; Rapp, 1995; Tanaka and Iwasaki, 1996; Franz et al., 1998; Ballesteros and Croft, 1998; Masuichi et al., 2000; Sadat et al., 2004), but transliteration in the context of comparable corpora has not been well addressed. The general idea of exploiting time correlations to acquire word translations from comparable corpora has been explored in several previous studies — e.g., (Fung, 1995; Rapp, 1995; Tanaka and Iwasaki, 1996). Recently, a Pearson correlation method was proposed to mine word pairs from comparable corpora (Tao and Zhai, 2005); this idea is similar to the method used in (Kay and Roscheisen, 1993) for sentence alignment. In our work, we adopt the method proposed in (Tao and Zhai, 2005) and apply it to the problem of transliteration; note that (Tao and Zhai, 2005) compares several different metrics for time correlation, as we also note below — and see (Sproat et al., 2006).

3 Transliteration with Comparable Corpora

We start from comparable corpora, consisting of newspaper articles in English and the target languages for the same time period. In this paper, the target languages are Arabic, Chinese and Hindi. We then extract named-entities in the English text using the named-entity recognizer described in (Li et al., 2004), which is based on the SNoW machine learning toolkit (Carlson et al., 1999). To perform transliteration, we use the following general approach: **1** Extract named entities from the English corpus for each day; **2** Extract candidates from the same day's newspapers in the target language; **3** For each English named entity, score and rank the target-language candidates as potential transliterations. We apply two unsupervised methods — time correlation and pronunciation-based methods — independently, and in combination.

3.1 Candidate scoring based on pronunciation

Our phonetic transliteration score uses a standard string-alignment and alignment-scoring technique based on (Kruskal, 1999) in that the distance is determined by a combination of substitution, insertion and deletion costs. These costs are computed from a language-universal cost matrix based on phonological features and the degree of phonetic similarity. (Our technique is thus similar to other work on phonetic similarity such as (Frisch, 1996) though details differ.) We construct a single cost matrix, and apply it to English and all target languages. This technique requires the knowledge of the phonetics and the sound change patterns of the language, but it does not require a transliteration-pair training dictionary. In this paper we assume the WorldBet transliteration system (Hieronymus, 1995), an ASCII-only version of the IPA.

The cost matrix is constructed in the following way. All phonemes are decomposed into standard phonological features. However, phonological features alone are not enough to model the possible substution/insertion/deletion patterns of languages. For example, /h/ is more frequently deleted than other consonants, whereas no single phonological feature allows us to distinguish /h/ from other consonants. Similarly, stop and fricative consonants such as /p, t, k, b, d, g, s, z/ are frequently deleted when they appear in the coda position. This tendency is very salient when the target languages do not allow coda consonants or consonant clusters. So, Chinese only allows [n, N] in coda position, and stop consonants in coda position are frequently lost; *Stanford* is transliterated as *sitanfu*, with the final /d/ lost. Since phonological features do not consider the position in the syllable, this pattern cannot be captured by conventional phonological features alone. To capture this, an additional feature "deletion of stop/fricative consonant in the coda position" is added. We base these observations, and the concomitant *pseudofeatures* on pronunciation error data of learners of English as a second language, as reported in (Swan and Smith, 2002). Er-

rors in second language pronunciation are determined by the difference in the phonological system of learner's first and second language. The same substitution/deletion/insertion patterns in the second language learner's errors appear also in the transliteration of foreign names. For example, if the learner's first language does not have a particular phoneme found in English, it is substituted by the most similar phoneme in their first language. Since Chinese does not have /v/, it is frequently substituted by /w/ or /f/. This substitution occurs frequently in the transliteration of foreign names in Chinese. Swan & Smith's study covers 25 languages, and includes Asian languages such as Thai, Korean, Chinese and Japanese, European languages such as German, Italian, French, and Polish and Middle Eastern languages such as Arabic and Farsi. Frequent substitution/insertion/deletion patterns of phonemes are collected from these data. Some examples are presented in Table 1.

Twenty phonological features and 14 pseudofeatures are used for the construction of the cost matrix. All features are classified into 5 classes. There are 4 classes of consonantal features — place, manner, laryngeality and major (consonant, sonorant, syllabicity), and a separate class of vocalic features. The purpose of these classes is to define groups of features which share the same substitution/insertion/deletion costs. Formally, given a class $\mathcal{C}$, and a cost $C_{\mathcal{C}}$, for each feature $f \in \mathcal{C}$, $C_{\mathcal{C}}$ defines the cost of substituting a different value for f than the one present in the source phoneme. Among manner features, the feature *continuous* is classified separately, since the substitution between stop and fricative consonants is very frequent; but between, say, nasals and fricatives such substitution is much less common. The cost for frequent sound change patterns should be low. Based on our intuitions, our pseudofeatures are classified into one or another of the above-mentioned five classes. The substitution/deletion/insertion cost for a pair of phonemes is the sum of the individual costs of the features which are different between the two phonemes. For example, /n/ and /p/ are different in sonorant, labial and coronal features. Therefore, the substitution cost of /n/ for /p/ is the sum of the sonorant, labial and coronal cost (20+10+10 = 40). Features and associated costs are shown in Table 2. Sample substitution, insertion, and deletion costs for /g/ are presented in Table 3.

The resulting cost matrix based on these principles is then used to calculate the edit distance between two phonetic strings. Pronunciations for English words are obtained using the Festival text-to-speech system (Taylor et al., 1998), and the target language words are automatically converted into their phonemic level transcriptions by various language-dependent means. In the case of Mandarin Chinese this is based on the standard pinyin transliteration system. For Arabic this is based on the orthography, which works reasonably well given that (apart from the fact that short vowels are no represented) the script is fairly phonemic. Similarly, the pronunciation of Hindi can be reasonably well-approximated based on the standard Devanagari orthographic representation. The edit cost for the pair of strings is normalized by the number of phonemes. The resulting score ranges from zero upwards; the score is used to rank candidate transliterations, with the candidate having the lowest cost being considered the most likely transliteration. Some examples of English words and the top three ranking candidates among all of the potential target-language candidates are given in Table 4.[1] Starred entries are correct.

3.2 Candidate scoring based on time correlation

Names of the same entity that occur in different languages often have correlated frequency patterns due to common triggers such as a major event. For example, the 2004 tsunami disaster was covered in news articles in many different languages. We would thus expect to see a peak of frequency of names such as *Sri Lanka*, *India*, and *Indonesia* in news articles published in multiple languages in the same time period. In general, we may expect topically related names in different languages to tend to co-occur together over time. Thus if we have comparable news articles over a sufficiently long time period, it is possible to exploit such correlations to learn the associations of names in different languages.

The idea of exploiting time correlation has been well studied. We adopt the method proposed in (Tao and Zhai, 2005) to represent the source name and each name candidate with a frequency vector and score each candidate by the similarity of the

[1] We describe candidate selection for each of the target languages later.

Input	Output	Position
D	D, d, z	everywhere
T	T, t, s	everywhere
N	N, n, g	everywhere
p/t/k	**deletion**	coda

Table 1: Substitution/insertion/deletion patterns for phonemes based on English second-language learner's data reported in (Swan and Smith, 2002). Each row shows an input phoneme class, possible output phonemes (including null), and the positions where the substitution (or deletion) is likely to occur.

Class	Feature	Cost
Major features and Consonant Del	consonant	20
	sonorant	
	consonant deletion	
Place features and Vowel Del	coronal	10
	vowel del/ins	
	stop/fricative consonant del at coda position	
	h del/ins	
Manner features	nasal	5
	dorsal feature for palatal consonants	
Vowel features and Exceptions	vowel height	3
	vowel place	
	exceptional	
Manner/ Laryngeal features	continuous	1.5
	voicing	

Table 2: Examples of features and associated costs. Pseudofeatures are shown in boldface. **Exceptional** denotes a situation such as the semivowel [j] substituting for the affricate [dZ]. Substitutions between these two sounds actually occur frequently in second-language error data.

two frequency vectors. This is very similar to the case in information retrieval where a query and a document are often represented by a term vector and documents are ranked by the similarity between their vectors and the query vector (Salton and McGill, 1983). But the vectors are very different and should be constructed in quite different ways. Following (Tao and Zhai, 2005), we also normalize the raw frequency vector so that it becomes a frequency distribution over all the time points. In order to compute the similarity between two distribution vectors $\vec{x} = (x_1, ..., x_T)$ and $\vec{y} = (y_1, ..., y_T)$, the Pearson correlation coefficient was used in (Tao and Zhai, 2005). We also consider two other commonly used measures – **cosine** (Salton and McGill, 1983), and **Jensen-Shannon divergence** (Lin, 1991), though our results show that Pearson correlation coefficient performs better than these two other methods. Since the time correlation method and the phonetic correspondence method exploit *distinct* resources, it makes sense to combine them. We explore two approaches to combining these two methods, namely *score combination* and *rank combination*. These will be defined below in Section 4.2.

4 Experiments

We evaluate our algorithms on three comparable corpora: English/Arabic, English/Chinese, and English/Hindi. Data statistics are shown in Table 5.

From each data set in Table 5, we picked out all news articles from seven randomly selected days. We identified about 6800 English names using the entity recognizer from (Carlson et al., 1999), and chose the most frequent 200 names as our English named entity candidates. Note that we chose the most frequent names because the reliability of the statistical correlation depends on the size of sample data. When a name is rare in a collection,

Source	Target	*Cost*	Target	*Cost*
g	g	0	r	40.5
	kh	2.5	e	44.5
	cCh	5.5	**del**	24
	tsh	17.5	**ins**	20
	N	26.5		

Table 3: Substitution/deletion/insertion costs for /g/.

English		Candidate	
		Script	Worldbet
Philippines	1	فلبين	f l b y n
	*2	فلبينية	f l b y n y t
	3	فلبينَى	f l b y n a
Megawati	*1	مَحَافظ	m h a f th
	2	ميجَاوَاتَى	m i j a w a t a
	3	مَاكوزَا	m a k w z a

English		Candidate		
		Script	Romanization	Worldbet
Belgium	*1	बेल्जियम	beljiyam	b e l j i y a m
	2	बेरहम	beraham	b e 9 a h a m
	3	फोरम	phoram	ph o 9 a m
Paraguay	1	परिचय	paricay	p a 9 i c a y
	*2	पैराग्वे	pairaagve	p a i 9 a g v e
	3	भिड़ेगी	bhir.egii	bh i rr e g i

English		Candidate		
		Script	Pinyin	Worldbet
Angola	*1	安哥拉	an-ge-la	a n k & l a
	1	安格拉	an-ge-la	a n k & l a
	2	阿格拉	a-ge-la	a k & l a
Megawati	*1	梅加瓦蒂	me-jia-wa-ti	m & i cC j a w a t i
	2	米九几	mi-jie-ji	m i cC j & u cC i
	3	马哈蒂尔	ma-ha-ti-er	m a x a t i & r

Table 4: Examples of the three top candidates in the transliteration of English/Arabic, English/Hindi and English/Chinese. The second column is the rank.

one can either only use the phonetic model, which does not depend on the sample size; or else one must expand the data set and hope for more occurrence. To generate the Hindi and Arabic candidates, all words from the same seven days were extracted. The words were stemmed all possible ways using simple hand-developed affix lists: for example, given a Hindi word $c_1c_2c_3$, if both c_3 and c_2c_3 are in our suffix and ending list, then this single word generates three possible candidates: c_1, c_1c_2, and $c_1c_2c_3$. In contrast, Chinese candidates were extracted using a list of 495 characters that are frequently used for foreign names (Sproat et al., 1996). A sequence of three or more such characters from the list is taken as a possible name. The number of candidates for each target language is presented in the last column of Table 5.

We measured the accuracy of transliteration by Mean Reciprocal Rank (MRR), a measure commonly used in information retrieval when there is precisely one correct answer (Kantor and Voorhees, 2000).

We attempted to create a complete set of answers for 200 English names in our test set, but a small number of English names do not seem to have any standard transliteration in the target language according to the resources that we looked at, and these names we removed from the evaluation set. Thus, we ended up having a list of less than 200 English names, shown in the second column of Table 6 (**All**). Furthermore some correct transliterations are not found in our candidate list for the second language, for two reasons: (1) The answer does not occur at all in the target news articles; (Table 6 # Missing 1) (2) The answer is there, but our candidate generation method has missed it. (Table 6 # Missing 2) Thus this results in an even smaller number of candidates to evaluate (**Core**); this smaller number is given in the fifth column of Table 6. We compute MRRs on the two sets

Languages	News Agency	Period	# days	# Words	# Cand.
Eng/Arab	Xinhua/Xinhua	08/06/2001–11/07/2001	150	12M/1.8M	12466
Eng/Chin	Xinhua/Xinhua	08/06/2001– 11/07/2001	150	12M/21M	6291
Eng/Hind	Xinhua/Naidunia	08/01/1997–08/03/1998	380	24M/5.5M	10169

Table 5: Language-pair datasets.

Language	# All	# missing 1	# missing 2	# Core
Arabic	192	113	9	70
Chinese	186	83	1	82
Hindi	147	82	0	62

Table 6: Number of evaluated English NEs.

of candidates — those represented by the count in column 2, and the smaller set represented by the count in column 5; we term the former MRR "AllMRR" and the latter "CoreMRR".[2] It is worth noting that the major reason for not finding a candidate transliteration of an English name in the target language is almost always because it is really not there, rather than because our candidate generation method has missed it. Presumably this reflects the fact that the corpora are merely comparable, rather than parallel. But the important point is that the true performance of the system would be closer to what we report below for CoreMRR, if we were working with truly parallel data where virtually all source language names would have target-language equivalents.

4.1 Performance of phonetic method and time correlation method

The performance of the phonetic method and the time correlation method are reported in Table 7, top and middle panels, respectively. In addition to the MRR scores, we also report another metric — CorrRate, namely the proportion of times the first candidate is the correct one.

Each of the two methods has advantages and disadvantages. The time correlation method relies more on the quality of the comparable corpora. It is perhaps not surprising that the time correlation method performs the best on English/Chinese, since these data come from the same source (Xinhua). Because the English and Hindi corpora are from different new agencies (Xinhua and Naidunia), the method performs relatively poorly. On the other hand, the phonetic method is less affected by corpus quality, but is sensitive to differences between languages. As discussed in the introduction, Hindi is relatively easy, and so we see the best MRR scores there. The performance is worse on Chinese and Arabic. It makes sense then to consider combining the two methods.

[2]We are aware that the resulting test set is very small, but we believe that it is large enough to demonstrate that the method is effective.

4.2 Method combination

In this section, we evaluate the performance of such a combination. We first use the phonetic method to filter out unlikely candidates, and then apply both the phonetic method and the time correlation method to rank the candidates.

We explore two combination methods: *score combination* and *rank combination*. In score combination, since the scores of two methods are not on the same scale, we first normalize them into the range [0,1] where the 1 is the best transliteration score and 0 the worst. Given a phonetic score p and a time correlation score t on the same transliteration pairs, the final combination score f would be: $f = \alpha \times p + (1 - \alpha) \times t$, where $\alpha \in [0, 1]$ is a linear combination parameter. For the rank combination, we take the *unnormalized* rankings of each candidate pair by the two methods and combine as follows: $r_{combined} = \alpha \times r_p + (1 - \alpha) \times r_t$, where r_p and r_t are the phonetic and temporal rankings, respectively.

The bottom panel of Table 7 shows the CoreMRR scores for these combination methods. In the second and third column, we repeat the phonetic and time correlation scores for ease of comparison. The fourth column and the sixth column represent the combination results with $\alpha = 0.5$ for both combination methods. The fifth column and the last column are the best MRR scores that we can achieve through tuning α's. Score combination, in particular, significantly outperforms the individual phonetic and time correlation methods alone.

Figure 1 plots the performance for all three languages with a variety of α's for the score combination method. Note that a higher α puts more weight on the phonetic model. As we have noted above, favoring the phonetic model is an advantage in our English/Hindi evaluation where the

Language	AllMRR	ALLCorrRate	CoreMRR	CoreCorrRate
Arabic	0.226	0.120	0.599	0.320
Chinese	0.281	0.203	0.637	0.462
Hindi	0.309	0.259	0.727	0.610

Language	AllMRR	AllCorrRate	CoreMRR	CoreCorrRate
Arabic	0.246	0.164	0.676	0.450
Chinese	0.363	0.292	0.824	0.662
Hindi	0.212	0.158	0.499	0.372

Language	Phonetic	Time Correlation	ScoreComb $\alpha = 0.5$	ScoreComb best α	RankComb $\alpha = 0.5$	RankComb best α
Arabic	0.599	0.676	0.733	0.788	0.733	0.754
Chinese	0.637	0.824	0.864	0.875	0.811	0.843
Hindi	0.727	0.499	0.749	0.761	0.689	0.765

Table 7: MRRs and CorrRate for the pronunciation method (top) and time correlation method (middle). The bottom table shows the scores for the combination (CoreMRR).

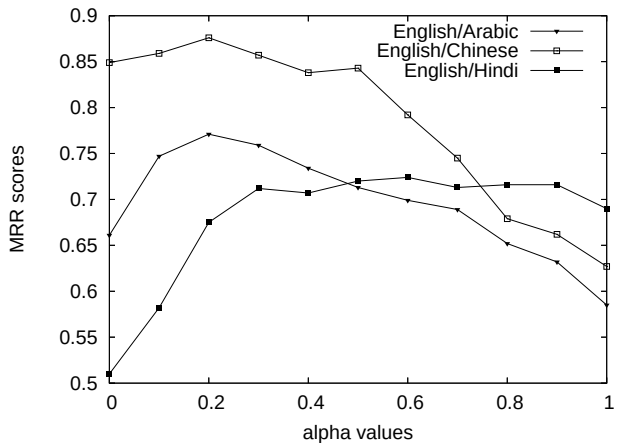

Figure 1: CoreMRR scores with different α values using score combination. A higher α puts more weight on the phonetic model.

phonetic correspondence between the two languages is fairly close, but the data sources are quite different; whereas for Arabic and Chinese we observe the opposite tendency. This suggests that one can balance the α scores according to whether one trusts one's data source versus whether one trusts in the similarity of the two languages' phonotactics.[3]

[3]A reviewer notes that we have not compared our method to state-of-the-art supervised transliteration models. This is true, but in the absence of a common evaluation set for transliteration, such a comparison would be meaningless. Certainly there are no standard databases, so far as we know, for the three language pairs we have been considering.

5 Conclusions and Future Work

In this paper we have discussed the problem of name transliteration as one component of a system for finding matching names in comparable corpora. We have proposed two unsupervised methods for transliteration, one that is based on carefully designed measures of phonetic correspondence and the other that is based on the temporal distribution of words. We have shown that both methods yield good results, and that even better results can be achieved by combining the methods.

One particular area that we will continue to work on is phonetic distance. We believe our hand-assigned costs are a reasonable starting point if one knows nothing about the particular pair of languages in question. However one could also train such costs, either from an existing list of known transliterations, or as part of an iterative bootstrapping method as, for example, in Yarowsky and Wicentowski's (2000) work on morphological induction.

The work we report is ongoing and is part of a larger project on multilingual named entity recognition and transliteration. One of the goals of this project is to develop tools and resources for under-resourced languages. Insofar as the techniques we have proposed have been shown to work on three language pairs involving one source language (English) and three unrelated and quite different target languages, one can reasonably claim that the techniques are language-independent. Furthermore, as

the case of Hindi shows, even with data from completely different news agencies we are able to extract useful correspondences.

6 Acknowledgments

This work was funded by Dept. of the Interior contract NBCHC040176 (REFLEX). We thank three EMNLP reviewers for useful feedback.

References

Y. Al-Onaizan and K. Knight. 2002. Machine transliteration of names in Arabic text. In *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, Philadelphia, PA.

Lisa Ballesteros and W. Bruce Croft. 1998. Resolving ambiguity for cross-language retrieval. In *Research and Development in Information Retrieval*, pages 64–71.

A. Carlson, C. Cumby, J. Rosen, and D. Roth. 1999. The SNoW learning architecture. Technical Report UIUCDCS-R-99-2101, UIUC CS Dept.

Martin Franz, J. Scott McCarley, and Salim Roukos. 1998. Ad hoc and multilingual information retrieval at IBM. In *Text REtrieval Conference*, pages 104–115.

S. Frisch. 1996. *Similarity and Frequency in Phonology*. Ph.D. thesis, Northwestern University, Evanston, IL.

Pascale Fung. 1995. A pattern matching method for finding noun and proper noun translations from noisy parallel corpora. In *Proceedings of ACL 1995*, pages 236–243.

W. Gao, K.-F. Wong, and W. Lam. 2004. Phoneme-based transliteration of foreign names for OOV problem. In *IJCNLP*, pages 374–381, Sanya, Hainan.

James Hieronymus. 1995. Ascii phonetic symbols for the world's languages: Worldbet. http://www.ling.ohio-state.edu/ edwards/worldbet.pdf.

P. Kantor and E. Voorhees. 2000. The TREC-5 confusion track: Comparing retrieval methods for scanned text. *Information Retrieval*, 2:165–176.

M. Kay and M. Roscheisen. 1993. Text translation alignment. *Computational Linguistics*, 19(1):75–102.

K. Knight and J. Graehl. 1998. Machine transliteration. *Computational Linguistics*, 24(4).

J. Kruskal. 1999. An overview of sequence comparison. In D. Sankoff and J. Kruskal, editors, *Time Warps, String Edits, and Macromolecules*, chapter 1, pages 1–44. CSLI, 2nd edition.

X. Li, P. Morie, and D. Roth. 2004. Robust reading: Identification and tracing of ambiguous names. In *NAACL-2004*.

J. Lin. 1991. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151.

H. Masuichi, R. Flournoy, S. Kaufmann, and S. Peters. 2000. A bootstrapping method for extracting bilingual text pairs.

H.M. Meng, W.K Lo, B. Chen, and K. Tang. 2001. Generating phonetic cognates to handle named entities in English-Chinese cross-languge spoken document retrieval. In *Proceedings of the Automatic Speech Recognition and Understanding Workshop*.

R. Rapp. 1995. Identifying word translations in non-parallel texts. In *Proceedings of ACL 1995*, pages 320–322.

F. Sadat, M. Yoshikawa, and S. Uemura. 2004. Bilingual terminology acquisition from comparable corpora and phrasal translation to cross-language information retrieval. http://acl.ldc.upenn.edu/P/P03/P03-2025.pdf.

G. Salton and M. McGill. 1983. *Introduction to Modern Information Retrieval*. McGraw-Hill.

R. Sproat, C. Shih, W. Gale, and N. Chang. 1996. A stochastic finite-state word-segmentation algorithm for Chinese. *Computational Linguistics*, 22(3).

Richard Sproat, Tao Tao, and ChengXiang Zhai. 2006. Named entity transliteration with comparable corpora. In *Proceedings of COLING-ACL 2006*, Sydney, July.

Michael Swan and Bernard Smith. 2002. *Learner English*. Cambridge University Press, Cambridge.

K. Tanaka and H. Iwasaki. 1996. Extraction of lexical translation from non-aligned corpora. In *Proceedings of COLING 1996*.

Tao Tao and ChengXiang Zhai. 2005. Mining comparable bilingual text corpora for cross-language information integration. In *Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 691–696.

P. Taylor, A. Black, and R. Caley. 1998. The architecture of the Festival speech synthesis system. In *Proceedings of the Third ESCA Workshop on Speech Synthesis*, pages 147–151, Jenolan Caves, Australia.

D. Yarowsky and R. Wicentowski. 2000. Minimally supervised morphological analysis by multimodal alignment. In K. Vijay-Shanker and Chang-Ning Huang, editors, *Proceedings of the 38th Meeting of the Association for Computational Linguistics*, pages 207–216, Hong Kong, October.