A Modified Joint Source-Channel Model for Transliteration

Asif Ekbal
Comp. Sc. & Engg. Deptt.
Jadavpur University
India
ekbal_asif12@
yahoo.co.in

Sudip Kumar Naskar
Comp. Sc. & Engg. Deptt.
Jadavpur University
India
sudip_naskar@
hotmail.com

Sivaji Bandyopadhyay
Comp. Sc. & Engg. Deptt.
Jadavpur University
India
sivaji_cse_ju@
yahoo.com

Abstract

Most machine transliteration systems transliterate out of vocabulary (OOV) words through intermediate phonemic mapping. A framework has been presented that allows direct orthographical mapping between two languages that are of different origins employing different alphabet sets. A modified joint source–channel model along with a number of alternatives have been proposed. Aligned transliteration units along with their context are automatically derived from a bilingual training corpus to generate the collocational statistics. The transliteration units in Bengali words take the pattern C^+M where C represents a vowel or a consonant or a conjunct and M represents the vowel modifier or matra. The English transliteration units are of the form C*V* where C represents a consonant and V represents a vowel. A Bengali-English machine transliteration system has been developed based on the proposed models. The system has been trained to transliterate person names from Bengali to English. It uses the linguistic knowledge of possible conjuncts and diphthongs in Bengali and their equivalents in English. The system has been evaluated and it has been observed that the modified joint source-channel model performs best with a Word Agreement Ratio of 69.3% and a Transliteration Unit Agreement Ratio of 89.8%.

1 Introduction

In Natural Language Processing (NLP) application areas such as information retrieval, question answering systems and machine translation, there is an increasing need to translate OOV words from one language to another. They are translated through transliteration, the method of translating into another language by expressing the original foreign words using characters of the target language preserving the pronunciation in their original languages. Thus, the central problem in transliteration is predicting the pronunciation of the original word. Transliteration between two languages, that use the same set of alphabets, is trivial: the word is left as it is. However, for languages that use different alphabet sets, the names must be transliterated or rendered in the target language alphabets.

Technical terms and named entities make up the bulk of these OOV words. Named entities hold a very important place in NLP applications. Proper identification, classification and translation of named entities are very crucial in many NLP applications and pose a very big challenge to NLP researchers. Named entities are usually not found in bilingual dictionaries and they are very productive in nature. Translation of named entities is a tricky task: it involves both translation and transliteration. Transliteration is commonly used for named entities, even when the words could be translated. Different types of named entities are translated differently. Numerical and temporal expressions typically use a limited set of vocabulary words (e.g., names of months, days of the week etc.) and can be translated fairly easily using simple translation patterns. The named entity machine transliteration algorithms presented in this work

Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, pages 191–198,
Sydney, July 2006.

focus on person names, locations and organizations. A machine transliteration system that is trained on person names is very important in a multilingual country like India where large name collections like census data, electoral roll and railway reservation information must be available to multilingual citizens of the country in their vernacular. In the present work, the various proposed models have been evaluated on a training corpus of person names.

A hybrid neural network and knowledge-based system to generate multiple English spellings for Arabic personal names is described in (Arbabi et al., 1994). (Knight and Graehl, 1998) developed a phoneme-based statistical model using finite state transducer that implements transformation rules to do back-transliteration. (Stalls and Knight, 1998) adapted this approach for back transliteration from Arabic to English for English names. A spelling-based model is described in (Al-Onaizan and Knight, 2002a; Al-Onaizan and Knight, 2002c) that directly maps English letter sequences into Arabic letter sequences with associated probability that are trained on a small English/Arabic name list without the need for English pronunciations. The phonetics-based and spelling-based models have been linearly combined into a single transliteration model in (Al-Onaizan and Knight, 2002b) for transliteration of Arabic named entities into English.

Several phoneme-based techniques have been proposed in the recent past for machine transliteration using transformation-based learning algorithm (Meng et al., 2001; Jung et al., 2000; Vigra and Khudanpur, 2003). (Abduljaleel and Larkey, 2003) have presented a simple statistical technique to train an English-Arabic transliteration model from pairs of names. The two-stage training procedure first learns which n-gram segments should be added to unigram inventory for the source language, and then a second stage learns the translation model over this inventory. This technique requires no heuristic or linguistic knowledge of either language.

(Goto et al., 2003) described an English-Japanese transliteration method in which an English word is divided into conversion units that are partial English character strings in an English word and each English conversion unit is converted into a partial Japanese Katakana character string. It calculates the likelihood of a particular choice of letters of chunking into English conversion units for an English word by linking them to Katakana characters using syllables. Thus the English conversion units consider phonetic aspects. It considers the English and Japanese contextual information simultaneously to calculate the plausibility of conversion from each English conversion unit to various Japanese conversion units using a single probability model based on the maximum entropy method.

(Haizhou et al., 2004) presented a framework that allows direct orthographical mapping between English and Chinese through a joint source-channel model, called n-gram transliteration model. The orthographic alignment process is automated using the maximum likelihood approach, through the Expectation Maximization algorithm to derive aligned transliteration units from a bilingual dictionary. The joint source-channel model tries to capture how source and target names can be generated simultaneously, i.e., the context information in both the source and the target sides are taken into account.

A tuple n-gram transliteration model (Marino et al., 2005; Crego et al., 2005) has been log-linearly combined with feature functions to develop a statistical machine translation system for Spanish-to-English and English-to-Spanish translation tasks. The model approximates the joint probability between source and target languages by using trigrams.

The present work differs from (Goto et al., 2003; Haizhou et al., 2004) in the sense that identification of the transliteration units in the source language is done using regular expressions and no probabilistic model is used. The proposed modified joint source-channel model is similar to the model proposed by (Goto et. al., 2003) but it differs in the way the transliteration units and the contextual information are defined in the present work. No linguistic knowledge is used in (Goto et al., 2003; Haizhou et al., 2004) whereas the present work uses linguistic knowledge in the form of possible conjuncts and diphthongs in Bengali.

The paper is organized as follows. The machine transliteration problem has been formulated under both noisy-channel model and joint source-channel model in Section 2. A number of transliteration models based on collocation statistics including the modified joint source-channel model and their evaluation scheme have been proposed in Section 3. The Bengali-English machine transliteration scenario has been presented in Section 4. The proposed

models have been evaluated and the result of evaluation is reported in Section 5. The conclusion is drawn in Section 6.

2 Machine Transliteration and Joint Source-Channel Model

A transliteration system takes as input a character string in the source language and generates a character string in the target language as output. The process can be conceptualized as two levels of decoding: segmentation of the source string into transliteration units; and relating the source language transliteration units with units in the target language, by resolving different combinations of alignments and unit mappings. The problem of machine transliteration has been studied extensively in the paradigm of the noisy channel model.

For a given Bengali name B as the observed channel output, we have to find out the most likely English transliteration E that maximizes $P(E \mid B)$. Applying Bayes' rule, it means to find E to maximize

$$P(B,E) = P(B \mid E) * P(E) \quad (1)$$

with equivalent effect. This is equivalent to modelling two probability distributions: P(B|E), the probability of transliterating E to B through a noisy channel, which is also called transformation rules, and P(E), the probability distribution of source, which reflects what is considered good English transliteration in general. Likewiswe, in English to Bengali (E2B) transliteration, we could find B that maximizes

$$P(B,E) = P(E \mid B) * P(B) \quad (2)$$

for a given English name. In equations (1) and (2), P(B) and P(E) are usually estimated using n-gram language models. Inspired by research results of grapheme-to-phoneme research in speech synthesis literature, many have suggested phoneme-based approaches to resolving $P(B \mid E)$ and $P(E \mid B)$, which approximates the probability distribution by introducing a phonemic representation. In this way, names in the source language, say B, are converted into an intermediate phonemic representation P, and then the phonemic representation is further converted into the target language, say English E. In Bengali to English (B2E) transliteration, the phoneme-based approach can be formulated as $P(E \mid B) = P(E \mid P) * P(P \mid B)$ and conversely we have $P(B \mid E) = P(B \mid P) * P(P \mid E)$ for E2B back-transliteration.

However, phoneme-based approaches are limited by a major constraint that could compromise transliteration precision. The phoneme-based approach requires derivation of proper phonemic representation for names of different origins. One may need to prepare multiple language-dependent grapheme-to-phoneme(G2P) and phoneme-to-grapheme(P2G) conversion systems accordingly, and that is not easy to achieve.

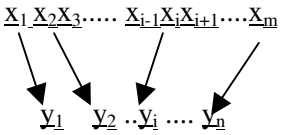
In view of close coupling of the source and target transliteration units, a joint source-channel model, or n-gram transliteration model (TM) has been proposed in (Haizhou et al., 2004). For K alligned transliteration units, we have

$$\begin{aligned} P(B,E) &= P(b_1, b_2 \ldots b_k, e_1, e_2 \ldots e_k) \\ &= P(\langle b,e\rangle_1, \langle b,e\rangle_2, \ldots \langle b,e\rangle_k) \\ &= \prod_{k=1}^{K} P(\langle b,e\rangle_k \mid \langle b,e\rangle_1^{k-1}) \end{aligned} \quad (3)$$

which provides an alternative to the phoneme-based approach for resolving equations (1) and (2) by eliminating the intermediate phonemic representation.

Unlike the noisy-channel model, the joint source-channel model does not try to capture how source names can be mapped to target names, but rather how source and target names can be generated simultaneously. In other words, a joint probability model is estimated that can be easily marginalized in order to yield conditional probability models for both transliteration and back-transliteration.

Suppose that we have a Bengali name $\alpha = x_1 x_2 \ldots x_m$ and an English transliteration $\beta = y_1 y_2 \ldots y_n$ where x_i, i = 1: m are Bengali transliteration units and y_j, j = 1: n are English transliteration units. An English transliteration unit may correspond to zero, one or more than one transliteration unit in Bengali. Often the values of m and n are different.



where there exists an alignment γ with $\langle b,e\rangle_1 = \langle x_1,y_1\rangle$; $\langle b,e\rangle_2 = \langle x_2x_3, y_2\rangle$; …. and $\langle b,e\rangle_k = \langle x_m,y_n\rangle$. A transliteration unit correspondence <b, e> is called a transliteration pair. Thus B2E transliteration can be formulated as

$$\overline{\beta} = \underset{\beta,\,\gamma}{\operatorname{argmax}}\, P(\alpha, \beta, \gamma) \quad (4)$$

and similarly the E2B back-transliteration as

$$\overline{\alpha} = \underset{\alpha, \gamma}{\operatorname{argmax}} P(\alpha, \beta, \gamma) \quad (5)$$

An *n*-gram transliteration model is defined as the conditional probability or transliteration probability of a transliteration pair $<b, e>_k$ depending on its immediate *n* predecessor pairs:

$$P(B, E) = P(\alpha, \beta, \gamma)$$

$$= \prod_{k=1}^{K} P(<b, e>_k \mid <b, e>_{k-n+1}^{k-1}) \quad (6)$$

3 Proposed Models and Evaluation Scheme

Machine transliteration has been viewed as a sense disambiguation problem. A number of transliteration models have been proposed that can generate the English transliteration from a Bengali word that is not registered in any bilingual or pronunciation dictionary. The Bengali word is divided into Transliteration Units (TU) that have the pattern C^+M, where C represents a vowel or a consonant or conjunct and M represents the vowel modifier or matra. An English word is divided into TUs that have the pattern C*V*, where C represents a consonant and V represents a vowel. The TUs are considered as the lexical units for machine transliteration. The system considers the Bengali and English contextual information in the form of collocated TUs simultaneously to calculate the plausibility of transliteration from each Bengali TU to various English candidate TUs and chooses the one with maximum probability. This is equivalent to choosing the most appropriate sense of a word in the source language to identify its representation in the target language. The system learns the mappings automatically from the bilingual training corpus guided by linguistic features. The output of this mapping process is a decision-list classifier with collocated TUs in the source language and their equivalent TUs in collocation in the target language along with the probability of each decision obtained from a training corpus. The machine transliteration of the input Bengali word is obtained using direct orthographic mapping by identifying the equivalent English TU for each Bengali TU in the input and then placing the English TUs in order. The various proposed models differ in the nature of collocational stastistics used during machine transliteration process: monogram model with no context, bigram model with previous (with respect to the current TU to be transliterated) source TU as the context, bigram model with next source TU as the context, bigram model with previous source and target TUs as the context (this is the joint source channel model), trigram model with previous and next source TUs as the context and the modified joint source-channel model with previous and next source TUs and the previous target TU as the context.

● Model A

In this model, no context is considered in either the source or the target side. This is essentially the monogram model.

$$P(B,E) = \prod_{k=1}^{K} P(<b,e>_k)$$

● Model B

This is essentially a bigram model with previous source TU, i.e., the source TU occurring to the left of the current TU to be transliterated, as the context.

$$P(B,E) = \prod_{k=1}^{K} P(<b,e>_k \mid b_{k-1})$$

●Model C

This is essentially a bigram model with next source TU, i.e., the source TU occurring to the right of the current TU to be transliterated, as the context.

$$P(B,E) = \prod_{k=1}^{K} P(<b,e>_k \mid b_{k+1})$$

● Model D

This is essentially the joint source-channel model where the previous TUs in both the source and the target sides are considered as the context. The previous TU on the target side refers to the transliterated TU to the immediate left of the current target TU to be transliterated.

$$P(B,E) = \prod_{k=1}^{K} P(<b,e>_k \mid <b,e>_{k-1})$$

● Model E

This is basically the trigram model where the previous and the next source TUs are considered as the context

$$P(B,E) = \prod_{k=1}^{K} P(<b,e>_k \mid b_{k-1}, b_{k+1})$$

● Model F

In this model, the previous and the next TUs in the source and the previous target TU are considered as the context. This is the modified joint source-channel model .

$$P(B,E) = \prod_{k=1}^{K} P(<b,e>_k \mid <b,e>_{k-1}, b_{k+1})$$

The performance of the system is evaluated in terms of Transliteration Unit Agreement Ratio (TUAR) and Word Agreement Ratio (WAR) following the evaluation scheme in (Goto et al., 2003). The evaluation parameter Character Agreement Ratio in (Goto et al., 2003) has been modified to Transliteration Unit Agreement Ratio as vowel modifier matra symbols in Bengali words are not independent and must always follow a consonant or a conjunct in a Transliteration Unit. Let, B be the input Bengali word, E be the English transliteration given by the user in open test and E' be the system generates the transliteration. TUAR is defined as, TUAR = (L-Err)/ L, where L is the number of TUs in E, and Err is the number of wrongly transliterated TUs in E' generated by the system. WAR is defined as, WAR= (S-Err$'$) / S, where S is the test sample size and Err$'$ is is the number of erroneous names generated by the system (when E' does not match with E). Each of these models has been evaluated with linguistic knowledge of the set of possible conjuncts and diphthongs in Bengali and their equivalents in English. It has been observed that the Modified Joint Source Channel Model with linguistic knowledge performs best in terms of Word Agreement Ratio and Transliteration Unit Agreement Ratio.

4 Bengali-English Machine Transliteration

Translation of named entities is a tricky task: it involves both translation and transliteration. Transliteration is commonly used for named entities, even when the words could be translated [জনতা দল (*janata dal*) is translated to *Janata Dal* (literal translation) although জনতা *(Janata)* and দল (*Dal)* are vocabulary words]. On the other hand যাদবপুর বিশ্ববিদ্যালয় (*jadavpur viswavidyalaya*) is translated to *Jadavpur University* in which যাদবপুর (*Jadavpur)* is transliterated to *Jadavpur* and বিশ্ববিদ্যালয় (*viswavidyalaya*) is translated to *University.*

A bilingual training corpus has been kept that contains entries mapping Bengali names to their respective English transliterations. To automatically analyze the bilingual training corpus to acquire knowledge in order to map new Bengali names to English, TUs are extracted from the Bengali names and the corresponding English names, and Bengali TUs are associated with their English counterparts.

Some examples are given below:

অভিনন্দন (*abhinandan*) → [অ । ভি । ন । ন্দ । ন]
abhinandan → [a | bhi | na | nda | n]

কৃষ্ণমূর্তি (*krishnamoorti*) → [কৃ । ষ্ণ । মূ । র্তি]
krishnamurthy → [kri | shna | mu | rthy]

শ্রীকান্ত (*srikant*) → [শ্রী । কা । ন্ত]
srikant → [sri | ka | nt]

After retrieving the transliteration units from a Bengali-English name pair, it associates the Bengali TUs to the English TUs along with the TUs in context.

For example, it derives the following transliteration pairs or rules from the name-pair:

রবীন্দ্রনাথ (*rabindranath)* → rabindranath

Source Language				**Target Language**	
previous TU	TU	next TU		previous TU	TU
-	র	বী	↔	-	ra
র	বী	ন্দ্র	↔	ra	bi
বী	ন্দ্র	না	↔	bi	ndra
ন্দ্র	না	থ	↔	ndra	na
না	থ	-	↔	na	th

But, in some cases, the number of transliteration units retrieved from the Bengali and English words may differ. The [বৃজমোহন (*brijmohan*) ↔ brijmohan] name pair yields 5 TUs in Bengali side and 4 TUs in English side [বৃ । জ । মো । হ । ন ↔ bri । jmo । ha । n]. In such cases, the system cannot align the TUs automatically and linguistic knowledge is used to resolve the confusion. A knowledge base that contains a list of Bengali conjuncts and diphthongs and their possible English representations has been kept. The hypothesis followed in the present work is that *the problem TU in the English side has always the maximum length.* If more than one English TU has the same length, then *system starts its analysis from the first one*. In the above example, the TUs *bri* and *jmo* have the same length. The system interacts with the knowledge base and ascertains that *bri* is valid and *jmo* cannot be a valid TU in English since there is no corresponding conjunct representation in Bengali. So *jmo* is split up into 2 TUs *j* and *mo*, and the system aligns the 5 TUs as [বৃ । জ । মো । হ । ন ↔ bri । j । mo । ha । n]. Similarly, [লোকনাথ (*loknath*) ↔ loknath] is initially split as [লো । ক । না । থ] ↔ lo । kna । th], and then as [lo । k । na । th] since *kna* has the maximum length and it does not have any valid conjunct representation in Bengali.

In some cases, the knowledge of Bengali diphthong resolves the problem. In the following example, [রা । ই । মা (*raima*) ↔ rai । ma], the number of TUs on both sides do not match. The English TU *rai* is chosen for analysis as its length is greater than the other TU *ma*. The vowel sequence *ai* corresponds to a diphthong in Bengali that has two valid representations < আই, ঐ >. The first representation signifies that a matra is associated to the previous character followed by the character ই. This matches the present Bengali input. Thus, the English vowel sequence *ai* is separated from the TU *rai* (rai → r । ai) and the intermediate form of the name pair appears to be [রা । ই । মা (*raima*) ↔ r । ai । ma]. Here, a *matra* is associated with the Bengali TU that corresponds to English TU *r* and so there must be a vowel attached with the TU *r*. TU *ai* is further splitted as *a* and *i* (ai → a । i) and the first one (i.e. *a)* is assimilated with the previous TU (i.e. *r*) and finally the name pair appears as: [রা । ই । মা (*raima*) ↔ ra । i । ma].

In the following two examples, the number of TUs on both sides does not match.

[দে । ব । রা । জ (*devraj*) ↔ de । vra । j]

[সো । ম । না । থ (*somnath*) ↔ so । mna । th]

It is observed that both *vr* and *mn* represent valid conjuncts in Bengali but these examples contain the constituent Bengali consonants in order and not the conjunct representation. During the training phase, if, for some conjuncts, examples with conjunct representation are outnumbered by examples with constituent consonants representation, the conjunct is removed from the linguistic knowledge base and training examples with such conjunct representation are moved to a Direct example base which contains the English words and their Bengali transliteration. The above two name pairs can then be realigned as

[দে । ব । রা । জ (*devraj*) ↔ de । v । ra । j]

[সো । ম । না । থ (*somnath*) ↔ so । m । na । th]

Otherwise, if such conjuncts are included in the linguistic knowledge base, training examples with constituent consonants representation are to be moved to the Direct example base.

The Bengali names and their English transliterations are split into TUs in such a way that, it results in a one-to-one correspondence after using the linguistic information. But in some cases there exits zero-to-one or many-to-one relationship. An example of Zero-to-One relationship [Φ → h] is the name-pair [আ । ল্লা (*alla*) ↔ a । lla । h] while the name-pair [আ । ই । ভি (*aivy*) ↔ i । vy] is an example of Many-to-One relationship [আ, ই → i]. These bilingual examples should also be included in the Direct example base.

In some cases, the linguistic knowledge apparently solves the mapping problem, but not always. From the name-pair [বরখা (*barkha*) ↔ barkha], the system initially generates the mapping [ব । র । খা ↔ ba । rkha] which is not one-to-one. Then it consults the linguistic knowledge base and breaks up the transliteration unit as (*rkha* → *rk* । *ha*) and generates the final

aligned transliteration pair [ব । র । খা ↔ ba । rk । ha] (since it finds out that *rk* has a valid conjunct representation in Bengali but not *rkh*), which is an incorrect transliteration pair to train the system. It should have been [ব । র । খা ↔ ba । r । kha]. Such type of errors can be detected by following the alignment process from the target side during the training phase. Such training examples may be either manually aligned or maintained in the Direct Example base.

5 Results of the Proposed Models

Approximately 6000 Indian person names have been collected and their English transliterations have been stored manually. This set acts as the training corpus on which the system is trained to generate the collocational statistics. These statistics serve as the decision list classifier to identify the target language TU given the source language TU and its context. The system also includes the linguistic knowledge in the form of valid conjuncts and diphthongs in Bengali and their English representation.

All the models have been tested with an open test corpus of about 1200 Bengali names that contains their English transliterations. The total number of transliteration units (TU) in these 1200 (Sample Size, i.e., S) Bengali names is 4755 (this is the value of L), i.e., on an average a Bengali name contains 4 TUs. The test set was collected from users and it was checked that it does not contain names that are present in the training set. The total number of transliteration unit errors (Err) in the system-generated transliterations and the total number of words erroneously generated ($Err^/$) by the system have been shown in Table 1 for each individual model. The models are evaluated on the basis of the two evaluation metrics, Word Agreement Ratio (WAR) and Transliteration Unit Agreement Ratio (TUAR). The results of the tests in terms of the evaluation metrics are shown in Table 2. The modified joint source-channel model (Model F) that incorporates linguistic knowledge performs best among all the models with a Word Agreement Ratio (WAR) of 69.3% and a Transliteration Unit Agreement Ratio (TUAR) of 89.8%. The joint source-channel model with linguistic knowledge (Model D) has not performed well in the Bengali-English machine transliteration whereas the trigram model (Model E) needs further attention as its result are comparable to the modified joint source-channel model (Model F). All the models were also tested for back-transliteration, i.e., English to Bengali transliteration, with an open test corpus of 1000 English names that contain their Bengali transliterations. The results of these tests in terms of the evaluation metrics WAR and TUAR are shown in Table 3. It is observed that the modified joint source-channel model performs best in back-transliteration with a WAR of 67.9% and a TUAR of 89%.

Model	Error in TUs (Err)	Error words ($Err^/$)
A	990	615
B	795	512
C	880	532
D	814	471
E	604	413
F	486	369

Table 1: Value of Err and $Err^/$ for each model (B2E transliteration)

Model	WAR (in %)	TUAR (in %)
A	48.8	79.2
B	57.4	83.3
C	55.7	81.5
D	60.8	82.9
E	65.6	87.3
F	69.3	89.8

Table 2: Results with Evaluation Metrics (B2E transliteration)

Model	WAR (in %)	TUAR (in %)
A	49.6	79.8
B	56.2	83.8
C	53.9	82.2
D	58.2	83.2
E	64.7	87.5
F	67.9	89.0

Table 3: Results with Evaluation Metrics (E2B transliteration)

6. Conclusion

It has been observed that the modified joint source-channel model with linguistic knowledge performs best in terms of Word Agreement Ratio (WAR) and Transliteration Unit Agreement Ratio (TUAR). Detailed examination of the

evaluation results reveals that Bengali has separate short and long vowels and the corresponding matra representation while these may be represented in English by the same vowel. It has been observed that most of the errors are at the *matra* level i.e., a short matra might have been replaced by a long matra or vice versa. More linguistic knowledge is necessary to disambiguate the short and the long vowels and the matra representation in Bengali. The system includes conjuncts and diphthongs as part of the linguistic knowledge base. Triphthongs or tetraphthongs usually do not appear in Indian names. But, inclusion of them will enable the system to transliterate those few names that may include them. The models are to be trained further on sets of additional person names from other geographic areas. Besides person names, location and organization names are also to be used for training the proposed models.

Acknowledgement

Our thanks go to Council of Scientific and Industrial Research, Human Resource Development Group, New Delhi, India for supporting Sudip Kumar Naskar under Senior Research Fellowship Award (9/96(402) 2003-EMR-I).

References

Abdul Jaleel Nasreen and Leah S. Larkey. 2003. *Statistical Transliteration for English-Arabic Cross Language Information Retrieval*. Proceedings of the Twelfth International Conference on Information and Knowledge Management (CIKM 2003), New Orleans, USA, 139-146.

Al-Onaizan Y. and Knight K. 2002a. *Named Entity Translation: Extended Abstract.* Proceedings of the Human Language Technology Conference (HLT 2002), 122-124.

Al-Onaizan Y. and Knight K.2002b. *Translating Named Entities Using Monolingual and Bilingual Resources.* Proceedings of the 40th Annual Meeting of the ACL (ACL 2002), 400-408.

Al-Onaizan Y. and Knight K. 2002c. *Machine Transliteration of Names in Arabic Text.* Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages.

Arbabi Mansur, Scott M. Fischthal, Vincent C. Cheng, and Elizabeth Bar. 1994. *Algorithms for Arabic name transliteration*. IBM Journal of Research and Development, 38(2): 183-193.

Crego J.M., Marino J.B. and A. de Gispert. 2005. *Reordered Search and Tuple Unfolding for Ngram-based SMT.* Proceedings of the MT-Summit X, Phuket, Thailand, 283-289.

Marino J. B., Banchs R., Crego J. M., A. de Gispert, P. Lambert, J. A. Fonollosa and M. Ruiz, *Bilingual N-gram Statistical Machine Translation.* Proceedings of the MT-Summit X, Phuket, Thailand, 275-282.

Goto I., N. Kato, N. Uratani, and T. Ehara. 2003. *Transliteration considering Context Information based on the Maximum Entropy Method*. Proceeding of the MT-Summit IX, New Orleans, USA, 125–132.

Haizhou Li, Zhang Min, Su Jian. 2004. *A Joint Source-Channel Model for Machine Transliteration*. Proceedings of the 42nd Annual Meeting of the ACL (ACL 2004), Barcelona, Spain, 159-166.

Jung Sung Young, Sung Lim Hong, and Eunok Paek. 2000. *An English to Korean Transliteration Model of Extended Markov Window*. Proceedings of COLING 2000, 1, 383-389.

Knight K. and J. Graehl. 1998. *Machine Transliteration*, Computational Linguistics, 24(4): 599-612.

Meng Helen M., Wai-Kit Lo, Berlin Chen and Karen Tang. 2001. *Generating Phonetic Cognates to handle Name Entities in English-Chinese Cross-language Spoken Document Retrieva*l. Proceedings of the Automatic Speech Recognition and Understanding (ASRU) Workshop, Trento, Italy.

Stalls, Bonnie Glover and Knight K. 1998. *Translating names and technical terms in Arabic text*. Proceedings of the COLING/ACL Workshop on Computational Approaches to Semitic Languages, Montral, Canada, 34-41.

Virga Paola and Sanjeev Khudanpur. 2003. *Transliteration of Proper Names in Crosslingual Information Retrieval*. Proceedings of the ACL 2003 Workshop on Multilingual and Mixed-language Named Entity Recognition, Sapporo, Japan, 57-60.