

Learning to find transliteration on the Web

Chien-Cheng Wu

Department of Computer Science
National Tsing Hua University
101 Kuang Fu Road, Hsin chu, Taiwan
d9283228@cs.nthu.edu.tw

Jason S. Chang

Department of Computer Science
National Tsing Hua University
101 Kuang Fu Road, Hsin chu, Taiwan
jschang@cs.nthu.edu.tw

This prototype demonstrate a novel method for learning to find transliterations of proper nouns on the Web based on query expansion aimed at maximizing the probability of retrieving transliterations from existing search engines. Since the method we used involves learning the morphological relationships between names and their transliterations, we refer to this IR-based approach as *morphological query expansion for machine transliteration*. The *morphological query expansion* approach is general in scope and can be applied to translation and transliteration, but we focus on transliteration in this paper.

Many texts containing proper names (e.g., “The cities of Mesopotamia prospered under Parthian and Sassanian rule.”) are submitted to machine translation services on the Web every day, and there are also service on the Web specifically target transliteration of proper names, including *CHINET* (Kwok et al. 2005) and *Livetrans* (Lu, Chien, and Lee 2004).

Machine translation systems on the Web such as *Yahoo Translate* (babelfish.yahoo.com) and *Google Translate* (translate.google.com/translate_t.g) typically use a bilingual dictionary that is either manually compiled or learned from a parallel corpus. However, such dictionaries often have insufficient coverage of proper names and technical terms, leading to poor translation due to out of vocabulary problem. The OOV problems of machine translation or cross language information retrieval can be handled more effectively by learning to find transliteration on the Web.

Consider Sentence 1 containing three place names.

1. *The cities of Mesopotamia prospered under Parthian and Sassanian rule.*

2. 城市繁榮下 *parthian* 達米亞、*sassanian* 統治。
3. 美索不達米亞城市在 巴底亞 和 薩珊 統治下繁榮起來。

Google Translate produce Sentence 2, leaving “Parthian” and “Sassanian” not translated. A good response might be a translation like Sentence 3 where all place names have appropriate transliterations (underlined). These transliterations can be more effectively retrieved from mixed code Web pages by extend each of the place names into a query (e.g., “Parthian NEAR 巴”). Intuitively by requiring one of likely prefix transliteration morphemes (e.g., “巴” or “帕” for “par-“ names), we can bias the search engine towards retrieving the correct transliterations (e.g., “巴底亞” and “帕提亞”) in snippets of many top-ranked documents.

The method involves pairing up the prefixing morphemes between name and transliteration in a set of train data, calculating the statistical association for these pair, and selecting pairs with a high degree of statistical association. The results of this training stage are morphological relationships between prefixes and postfixes of names and transliterations. At run time, a given name is automatically extended into a query with relevant prefixing morphemes, then the query is submit to some search engine. After retrieving snippets from a search engine, the system extract transliterations from the snippets based on redundancy, proximity between name and transliteration, and cross language morphological relationships of prefix and postfix.

We present a new machine transliteration system based on information retrieval and morphological query expansion. The system automatically

learns to extend the proper names into a query expected to retrieve and extract transliterations of the proper names. Consider the case of transliteration of “Parthian.” The system looks at possible prefixes of the given name, including *p-*, *pa-*, *par-*, and *part-*, and determine determines the best *n* query expansions (e.g., “Parthian 巴,” “Parthian 帕”). These effective expansions automatically during training by analyzing a collection of 23,615 place names and transliterations pairs.

We evaluated the prototype system using a list of 500 proper names. The results show that 60% of the time there are sufficient relevant data on the Web to carry out effective machine transliteration based on IR and morphological query expansion. Of many results returned by the system, the top 1, two and three results are 0.88, 0.93, and 0.94. By performing query expansion, the system improves the recall rate from 0.48 to 0.60.

The results indicate that most names and transliteration counterparts can often be found on the Web and the proposed method are very effective in retrieving and extracting transliterations based on a statistical machine transliteration model trained on a bilingual name list. Our demonstration prototype shows alternative transliterations in use on the Web and snippets of such usage, so that the user can easily validate these transliterations.

The prototype supports:

- Searching and extracting transliterations of a given term
- Listing alternative transliterations on the Web
- Listing alternative transliteration in a local dictionary
- Browsing of snippets containing for each alternative transliteration
- Saving transliterations in a local dictionary
- Selecting and saving transliteration in snippets to a local dictionary

The method explored here can be extended as an alternative way to support such MT subtasks as back transliteration (Knight and Graehl 1998) and noun phrase translation (Koehn and Knight 2003). Finally, for more challenging tasks, such as handling sentences, the improvement of translation quality

probably will also be achieved by combining this IR-based approach and statistical machine translation. For example, a preprocessing unit may replace the proper names in a sentence with transliterations (e.g., mixed code text such as Sentence 4) *on the fly* or by looking up a local dictionary before sending it off to MT for finally translation.

4. *The cities of 美索不達米亞 prospered under 巴底亞 and 薩珊 rule.*

Morphological query expansion represents an innovative way to capture cross-linguistic relations in name transliteration. The method is independent of the bilingual lexicon content making it easy to adopt to other proper names such person, product, or organization names. This approach is useful in a number of machine translation subtasks, including name transliteration, back transliteration, named entity translation, and terminology translation.

References

- Y. Cao and H. Li. (2002). *Base Noun Phrase Translation Using Web Data and the EM Algorithm*, In Proc. of COLING 2002, pp.127-133.
- K. Knight, J. Graehl. (1998). *Machine Transliteration*. In Journal of Computational Linguistics 24(4), pp.599-612.
- P. Koehn, K. Knight. (2003). *Feature-Rich Statistical Translation of Noun Phrases*. In Proc. of ACL 2003, pp.311-318.
- KL Kwok, P Deng, N Dinstl, HL Sun, W Xu, P Peng, *CHINET: a Chinese name finder system for document triage*. Proceedings of 2005 International Conference on Intelligence, 2005.
- T. Lin, J.C. Wu, and J. S. Chang. (2004). *Extraction of Name and Transliteration in Monolingual and Parallel Corpora*. In Proc. of AMTA 2004, pp.177-186.
- WH Lu, LF Chien, HJ Lee. *Anchor text mining for translation of Web queries: A transitive translation approach*. ACM Transactions on Information Systems (TOIS), 2004.
- M. Nagata, T. Saito, and K. Suzuki. (2001). *Using the Web as a bilingual dictionary*. In Proc. of ACL 2001 DD-MT Workshop, pp.95-102.

Demonstration Script



The system runs under Microsoft Windows as a local application program. It opens up a form and accepts a name for translation. The user can choose to search on the Web or to look up in the local dictionary for results previously obtained and edited.



The system now expanded the query and sending them to a search engine to retrieve snippets containing transliterations.



The system extracted transliterations from the snippets and showed the list of most likely answers.



The user can click each answer and view a list of snippets where the transliteration appears.

The user finds and selects “帕提亞人” in the snippets for “安息人”



The user can do a number of things with the transliteration list:

1. View the snippets associated with each transliteration
2. Delete a transliteration
3. Save the list along with the snippet
4. Select a string in the snippets and add it to the answer list by clicking the right bottom

The transliteration “帕提亞人” is added to the list



Ngram	Char	Map_no	Ngramno	Prob
alla	亞	2	11	0.1818
alla	阿	9	11	0.8182
anti	外	1	11	0.0909
anti	安	10	11	0.9091
bart	巴	11	11	1.0000
bata	八	1	11	0.0909
bata	巴	10	11	0.9091
belo	比	2	11	0.1818
belo	貝	9	11	0.8182
beth	伯	1	11	0.0909
beth	貝	10	11	0.9091
buch	巴	2	11	0.1818

This slide shows the relationships between the prefixes of names and translations.

Ngram	Char	Map_no	Ngramno	Prob
ague	圭	4	11	0.3636
ague	格	6	11	0.5455
ague	蓋	1	11	0.0909
alen	侖	1	11	0.0909
alen	倫	2	11	0.1818
alen	連	8	11	0.7273
andi	地	1	11	0.0909
andi	迪	7	11	0.6364
andi	第	2	11	0.1818
andi	德	1	11	0.0909

This slide shows the relationships between the postfixes of names and translations.

Cluster	Morpheme	Phonetic symbol
八	八	ㄅㄨㄚˋ
八	巴	ㄅㄨㄚˋ
八	把	ㄅㄨㄚˋ
八	拔	ㄅㄨㄚˋ
八	罷	ㄅㄨㄚˋ
八	霸	ㄅㄨㄚˋ
白	白	ㄅㄞˊ
白	百	ㄅㄞˊ
白	伯	ㄅㄞˊ
白	波	ㄅㄞˊ
白	泊	ㄅㄞˊ
白	勃	ㄅㄞˊ
白	柏	ㄅㄞˊ
白	玻	ㄅㄞˊ
白	博	ㄅㄞˊ
白	搏	ㄅㄞˊ
白	白	ㄅㄞˊ
白	百	ㄅㄞˊ
白	拜	ㄅㄞˊ

The performance can improved by clustering transliteration user can click each answer and view a list of snippets where the transliteration appears.