Statistical Machine Transliteration Baselines for NEWS 2018

Snigdha Singhania[1], **Minh Nguyen**[1], **Hoang Gia Ngo**[1], **Nancy F. Chen**[2]

[1]National University of Singapore, Singapore
{singhaniasnigdha,nguyen.binh.minh92,ngohgia}@u.nus.edu
[2]Singapore University of Technology and Design, Singapore
nancychen@alum.mit.edu

Abstract

This paper reports the results of our transliteration experiments conducted on NEWS 2018 Shared Task dataset. We focus on creating the baseline systems trained using two open-source, statistical transliteration tools, namely Sequitur and Moses. We discuss the pre-processing steps performed on this dataset for both the systems. We also provide a re-ranking system which uses top hypotheses from Sequitur and Moses to create a consolidated list of transliterations. The results obtained from each of these models can be used to present a good starting point for the participating teams.

1 Introduction

Transliteration is defined as the phonetic translation of words across languages (Knight and Graehl, 1998; Li et al., 2009). It can be considered as a machine translation problem at the character level. Transliteration converts words written in one writing system (source language, e.g., English) into phonetically equivalent words in another writing system (target language, e.g., Hindi) and is often used to translate foreign names of people, locations, organizations, and products (Gia et al., 2015). With names comprising over 75 percent of the unseen words (Bhargava and Kondrak, 2011), they are a challenging problem in machine translation, multilingual information retrieval, corpus alignment and other natural language processing applications. More so, studies suggest that cross-lingual information retrieval performances can improve by as much as 50 percent if the system is provided with suitably transliterated named entities (Larkey et al., 2003).

In this paper, we run two baseline transliteration experiments and report our results on the NEWS 2018 Shared Task dataset. A re-ranking model using linear regression has also been provided in an attempt to combine hypotheses from both the baselines. Song et al. (2010) proposed that the performance of a transliteration system is expected to improve when the output candidates are re-ranked, as the Shared Task considers only the top-1 hypothesis when evaluating a system. Our re-ranking approach which uses the union of Sequitur and Moses hypotheses results in the top-1 word accuracy for all language pairs to be either an improvement or lie in their respective Moses and Sequitur accuracy range, excluding English-to-Thai, English-to-Chinese and English-to-Vietnamese where the results are relatively poorer.

The rest of this paper is structured as follows. Section 2 contains a summary of the datasets used for the transliteration task. Section 3 describes the two well-known statistical transliteration methods adopted; first, a joint-source channel approach using Sequitur, and second, a phrase-based statistical machine translation approach using Moses. Section 4 focuses on the experimental setup, re-ranking approach, and documents the results obtained. Finally, Section 5 summarizes the paper.

2 Data

The corpus sizes of each of the data partitions, namely training, development and test for the 19 language pairs used in the transliteration experiments is summarized in Table 1.

3 Methods

In this section, we describe the two software tools used for the transliteration experiment: Sequitur, which is based on the joint source-channel model

Proceedings of the Seventh Named Entities Workshop, pages 74–78
Melbourne, Australia, July 20, 2018.

Task ID	Training	Development	Test
T-EnTh	30781	1000	1000
B-ThEn	27273	1000	1433
T-EnPe	13386	1000	1000
B-PeEn	15677	1000	908
T-EnCh	41318	1000	1000
B-ChEn	32002	1000	1000
T-EnVi	3256	500	500
M-EnBa	13623	1000	1000
M-EnHi	12937	1000	1000
T-EnHe	10501	1000	523
M-EnKa	10955	1000	1000
M-EnTa	10957	1000	1000
B-HeEn	9447	1000	590
T-ArEn	31354	1000	1000
T-EnKo	7387	1000	1000
T-EnJa	28828	1000	1000
B-JnJk	10514	1000	1000
B-EnPe	11204	1000	1000
T-PeEn	6000	1000	1000

Table 1: Corpus Size for the 19 language pairs, where En: English, Th: Thai, Pe: Persian, Ch: Chinese, Vi: Vietnamese, Ba: Bangla, Hi: Hindi, He: Hebrew, Ka: Kannada, Ta: Tamil, Ar: Arabic, Ko: Korean, Ja: Japanese Katakana, Jn: English, Jk: Japanese Kanji.

and Moses, which adopts phrase-based statistical machine translation. It should be noted that identical settings were used for all 19 language pairs.

3.1 Joint Source-Channel Model

The Joint Source-Channel Model was first studied by Li et al. (2004), where a direct orthographic mapping was proposed for transliteration. Given a pair of languages, for example English and Hindi, where *e* and *h* are representative of their transliteration units, respectively; the transliteration process is nding the alignment for sub-sequences of the input string, *E* and the output string, *H* (Pervouchine et al., 2009), and can be represented for an *n*-gram model as

$$\begin{aligned} P(E,H) &= P(e_1, e_2, ..., e_k, h_1, h_2, ..., h_k) \\ &= P(< e_1, h_1 >, ..., < e_k, h_k >) \\ &= \prod_{i=1}^{k} P(< e, h >_i \mid < e, h >_{i-n+1}^{i-1}) \end{aligned} \tag{1}$$

where *k* is number of alignment units. *P(E, H)* is, thus, the joint probability of the *i*-th alignment pair, which depends on *n* previous pairs in the sequence.

Sequitur is a data-driven translation tool, originally developed for grapheme-to-phoneme conversion by Bisani and Ney (2008). It is applicable to several monotonous sequence translation tasks and hence is a popular tool in machine transliteration. It is different from many translation tools, as it is able to train a joint *n*-gram model from unaligned data. Higher order *n*-grams are trained iteratively from the smaller ones — first, a unigram model is trained, which is then used for a bigram model, and so on. We report results on a 5-gram Sequitur model in this paper.

3.2 Phrase-Based Statistical Machine Translation (PB-SMT)

Phrase-based machine translation model breaks the source sentence into phrases and translates these phrases in the target language before combining them to produce one final translated result (Brown et al., 1993; Collins, 2011). Its use can be extended in the field of transliteration — as transliteration is defined as a translation task at the character level (Koehn et al., 2007). The best transliteration sequence, H^{best}, in the target language is generated by multiplying the probabilities of the transliteration model, *P* and the language model, *P(E | H)*, along with their respective weights, α and β, as

$$\begin{aligned} H^{best} &= argmax_{H \epsilon h} P(H|E) \\ &= argmax_{H \epsilon h} \alpha P(E|H) \times \beta P(H) \end{aligned} \tag{2}$$

where *h* is the set of all phonologically correct words in the target orthography.

Moses is the statistical translation tool, which adopts the Phrase-Based Statistical Machine Translation approach. GIZA++ is used for aligning the word pairs and KenLM is used for creating the *n*-gram language models. We create 5-gram language models using the target language corpus. The decoders log-linear model is tuned using MERT.

3.3 Hypothesis Re-ranking

Song et al. (2010) proposed that re-ranking the output candidates is expected to boost transliteration accuracy, as the Shared Task considers only the top-1 hypothesis when evaluating the accuracy of the system. We adopt the following re-ranking approach in an attempt to improve over the individual Moses and Sequitur results.

Moses + Sequitur: We conduct an experiment to analyze the outcome when using hypotheses from both Sequitur and Moses, where a linear combination of their corresponding scores is used to rank the consolidated hypothesis list. The feature set consists of 10 scores from lexical reordering, language modelling, word penalty, phrase penalty, and translation from Moses and 1 confidence score from Sequitur. We use constrained decoding to obtain Moses scores for Sequitur transliterations which do not occur in the top-n Moses hypotheses. A linear regression model similar to that adopted by Shao et al. (2015) is used for re-ranking. For each transliteration, we use the edit distance of the hypothesis from the reference as the output of the linear regression model, following Wang et al. (2015). The hypotheses are ranked in increasing order of their calculated edit distance. The linear regression model can be mathematically represented using:

$$ED = c + \sum_{i=1}^{10} \alpha_i x_i \quad (3)$$

where *ED* is the edit distance calculated by the regression model, c is the intercept, and α_i and x_i are the coefficient and value of the i-th feature. As the edit distance between the hypothesis and reference is a measure of their similarity, it is seen as an effective parameter which can be used to re-rank the different hypotheses. It should be noted that these re-ranking experiments were performed after the Shared Task deadline and are not included in the official results submitted to the workshop.

4 Experiments

4.1 Experimental Setup for Sequitur

As an inherent grapheme-to-phoneme converter, the target language is broken down into its phonetic letter representation (phonemes), which are individual target language characters in a transliteration task. An example from the English-Hindi corpus is shown in Figure 1.

Input (English)	Transliteration (Hindi)
AFRICA	अ फ ़ ् र ी क ा

Figure 1: An example of data pre-processing in Sequitur from the English-Hindi corpus where the English word is AFRICA and Hindi representation is अफ़्रीका.

4.2 Experimental Setup for Moses

For this experiment, we augment word representations with boundary markers (^ for the start of the word and $ for the end of the word). Adding boundary markers ensures that character position is encoded in these word representations, which is otherwise ignored in PB-SMT models (Kunchukuttan and Bhattacharyya, 2015). This significantly improves transliteration accuracy for languages (e.g., all Indian languages) which have different characters for identical phonological symbols depending on where (initial, medial or terminal position) they occur in a word. Figure 2 shows an example of how the strings are represented after pre-processing for Moses.

Input (English)	Transliteration (Hindi)
^ A F R I C A $	^ अ फ ़ ् र ी क ा $

Figure 2: An example of data pre-processing (augmented with word boundary markers) in Moses from the English-Hindi corpus where the English word is *AFRICA* and Hindi representation is *अफ़्रीका*.

4.3 Results

Results from Moses and Sequitur on the test set are included in Tables 2 and 3. Table 2 includes top-1 accuracy results, while Table 3 summarizes the mean F-scores, for outcomes from each of Sequitur, Moses, and the consolidated re-ranking model on the hidden test partition. The top-1 hypothesis from the (Moses + Sequitur) re-ranked model is found to be the top-1 Sequitur and top-1 Moses transliteration in 61.93% and 61.06% instances, on average; of which the Sequitur and Moses results are identical in 45.62% instances. 22.63% of the time, on average, the top-1 re-ranked hypothesis is neither the top-1 from Moses nor Sequitur. These numbers do not include the English-to-Persian and Persian-to-English (with Western names) datasets, on account of the encoding mismatch between their test set with their training and development set, which is discussed later in this section.

From observing the accuracy results reported in Table 2, Sequitur reports best results on 5 language pairs — English-to-Thai, English-to-Vietnamese, English-to-Tamil, English-to-Japanese and English-to-Persian (with Persian

Task ID	Sequitur	Moses	Re-ranked
T-EnTh	**14.10**	13.90	13.50
B-ThEn	22.33	22.89	**26.59**
T-EnPe	**0.10**	**0.10**	**0.10**
B-PeEn	0.00	**0.11**	**0.11**
T-EnCh	26.20	**26.30**	24.90
B-ChEn	17.50	17.90	**18.80**
T-EnVi	**45.00**	43.40	40.40
M-EnBa	38.20	40.70	**41.10**
M-EnHi	30.03	**33.33**	31.83
T-EnHe	16.83	**17.59**	17.40
M-EnKa	28.41	26.90	**30.02**
M-EnTa	**18.22**	16.01	17.73
B-HeEn	6.78	**9.16**	8.47
T-ArEn	33.80	35.00	**37.50**
T-EnKo	25.90	26.10	**29.20**
T-EnJa	**31.83**	29.13	31.73
B-JnJk	51.70	**60.30**	57.20
B-EnPe	**61.00**	55.60	57.10
T-PeEn	65.80	65.60	**66.40**

Table 2: Word accuracies (%) from Moses and Sequitur models reported on the test set.

Task ID	Sequitur	Moses	Re-ranked
T-EnTh	**0.759759**	0.751033	0.756556
B-ThEn	0.804144	0.806737	**0.823464**
T-EnPe	**0.216715**	0.200054	0.203888
B-PeEn	0.007387	**0.307681**	0.297896
T-EnCh	**0.650861**	0.648604	0.639682
B-ChEn	0.784957	0.792034	**0.805242**
T-EnVi	**0.872989**	0.858727	0.857129
M-EnBa	0.871288	**0.879262**	0.873197
M-EnHi	0.836694	0.842555	**0.843902**
T-EnHe	0.796416	0.799957	**0.801067**
M-EnKa	0.840973	0.836202	**0.848025**
M-EnTa	0.820962	0.817579	**0.822778**
B-HeEn	0.720478	0.733852	**0.739240**
T-ArEn	0.896376	0.896873	**0.900685**
T-EnKo	**0.674653**	0.671095	0.671618
T-EnJa	**0.780412**	0.773722	0. 777001
B-JnJk	0.759595	**0.785229**	0.771079
B-EnPe	**0.928553**	0.918301	0.925398
T-PeEn	**0.947587**	0.943719	0.946168

Table 3: Mean F-scores from Moses and Sequitur models reported on the test set.

names) while Moses works best for another 5 — namely, English-to-Chinese, English-to-Hindi, English-to-Hebrew, Hebrew-to-English, and English-to-Kanji. The combined re-ranking of Moses + Sequitur improves the top-1 accuracy for 7 language pairs, which are Thai-to-English, Chinese-to-English, English-to-Bengali, English-to-Kannada, Arabic-to-English, English-to-Korean and Persian-to-English (with Persian names).

Further, it is observed that English-to-Persian and Persian-to-English (with Western names) perform very poorly as 66.92% and 67.53% Persian characters in the test set, respectively, were not present in either the training or the development set. The model is thus unable to predict transliterations for these characters, which occurs very frequently in the test set and hence report 100% error rates. The same language pair, however, performs significantly better (55-65% accuracy) for Persian names where the test set introduces no new tokens from the data used to train the transliteration models.

5 Summary

The two systems based on the joint source-channel and phrase-based statistical approaches are baseline systems for the NEWS 2018 shared task. For all our experiments we have adopted a language independent approach, wherein each language pair is processed automatically from the character sequence representation supplied for the shared tasks, with no language specific treatment for any of the language pairs.

References

Aditya Bhargava and Grzegorz Kondrak. 2011. How do you pronounce your name?: improving g2p with transliterations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 399–408. Association for Computational Linguistics.

Maximilian Bisani and Hermann Ney. 2008. Joint-sequence models for grapheme-to-phoneme conversion. *Speech communication*, 50(5):434–451.

Peter F Brown, Vincent J Della Pietra, Stephen A Della Pietra, and Robert L Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263–311.

Michael Collins. 2011. Statistical machine translation: Ibm models 1 and 2. *Columbia Columbia Univ*.

Ngo Hoang Gia, Nancy F Chen, Nguyen Binh Minh, Bin Ma, and Haizhou Li. 2015. Phonology-

augmented statistical transliteration for low-resource languages. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Kevin Knight and Jonathan Graehl. 1998. Machine transliteration. *Computational Linguistics*, 24(4):599–612.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Anoop Kunchukuttan and Pushpak Bhattacharyya. 2015. Data representation methods and use of mined corpora for indian language transliteration. In *Proceedings of the Fifth Named Entity Workshop*, pages 78–82.

Leah S Larkey, Nasreen AbdulJaleel, and Margaret Connell. 2003. What's in a name?: Proper names in arabic cross language information retrieval. In *ACL Workshop on Comp. Approaches to Semitic Languages*. Citeseer.

Haizhou Li, A Kumaran, Vladimir Vladimire Pervouchine, and Min Zhang. 2009. Report of news 2009 machine transliteration shared task. In *Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration*, pages 1–18. Association for Computational Linguistics.

Haizhou Li, Min Zhang, and Jian Su. 2004. A joint source-channel model for machine transliteration. In *Proceedings of the 42nd Annual Meeting on association for Computational Linguistics*, page 159. Association for Computational Linguistics.

Vladimir Pervouchine, Haizhou Li, and Lin Bo. 2009. Transliteration alignment. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 136–144. Association for Computational Linguistics.

Yan Shao, Jörg Tiedemann, and Joakim Nivre. 2015. Boosting english-chinese machine transliteration via high quality alignment and multilingual resources. In *Proceedings of the Fifth Named Entity Workshop*, pages 56–60.

Yan Song, Chunyu Kit, and Hai Zhou. 2010. Reranking with multiple features for better transliteration. In *Proceedings of the 2010 Named Entities Workshop*, pages 62–65. Association for Computational Linguistics.

Yu-Chun Wang, Chun-Kai Wu, and Richard Tzong-Han Tsai. 2015. Ncu iisr english-korean and english-chinese named entity transliteration using different grapheme segmentation approaches. In *Proceedings of the Fifth Named Entity Workshop*, pages 83–87.