Low-Resource Machine Transliteration Using Recurrent Neural Networks of Asian Languages

Ngoc Tan Le
Universite du Quebec a Montreal / Canada
le.ngoc_tan@uqam.ca

Fatiha Sadat
Universite du Quebec a Montreal / Canada
sadat.fatiha@uqam.ca

Abstract

Grapheme-to-phoneme models are key components in automatic speech recognition and text-to-speech systems. With low-resource language pairs that do not have available and well-developed pronunciation lexicons, grapheme-to-phoneme models are particularly useful. These models are based on initial alignments between grapheme source and phoneme target sequences. Inspired by sequence-to-sequence recurrent neural network-based translation methods, the current research presents an approach that applies an alignment representation for input sequences and pre-trained source and target embeddings to overcome the transliteration problem for a low-resource languages pair. We participated in the NEWS 2018 shared task for the English-Vietnamese transliteration task.

1 Introduction

Transliteration means the phonetic translation of the words in a source language (*e.g. English*) into equivalent words in a target language (*e.g. Vietnamese*). It entails transforming a word from one writing system (the *"source word"*) to a phonetically equivalent word in another writing system (the *"target word"*) (Knight and Graehl, 1998). This transformation requires a large set of rules defined by expert linguists to determine how the phonemes are aligned and to take into account the phonological system of the target language. Many language pairs have adopted various rules for transliteration over time, and most transliteration depends on the origin of a word (Waxmonsky and Reddy, 2012).

In recent work on sequence-to-sequence neural network-based machine translation, the input vocabulary is large. Moreover, statistics for many words must be sparsely estimated (Sutskever et al., 2014; Jean et al., 2014). To deal with this linguistics aspect, neural network-based approaches use continuous-space representations of words or word embeddings, in which words that occur in similar context tend to be close to each other in representational space. The benefits of using neural networks, particularly, recurrent neural networks, to deal with sparse problem are very clear.

We have observed that the state-of-the-art grapheme-to-phoneme methods were based on the use of grapheme-phoneme mappings (Oh et al., 2006; Bisani and Ney, 2008; Duan et al., 2016). However, recurrent neural networks approaches do not require any alignment information. In this study, we propose a novel method to build a low-resource machine transliteration system, using RNN-based models and alignment information for input sequences. Given a new word in the source language that does not exist in the bilingual pronunciation dictionary, this system automatically predicts the phonemic representation of a word in the target language. We are interested in solving out-of-vocabulary words for machine translation systems, such as proper nouns or technical terms, for a low-resource language pair, in this case English and Vietnamese.

The structure of the article is as follows: Section 2 presents the state of the art on machine transliteration. In section 3, we describe our proposed approach. Then, in section 4, we present our experiments, compare our system's performance with other systems. Finally, in section 5, we present our conclusions and perspectives for future research.

2 Related Work

Transliteration can be considered as a subtask of machine translation, when we need to translate source graphemes into target phonemes. In other words, an alignment model needs to be constructed first, and the translation model is built

Proceedings of the Seventh Named Entities Workshop, pages 95–100
Melbourne, Australia, July 20, 2018.

on the basis of the alignments. Transliterating a word from the language of its origin to a foreign language is called *Forward Transliteration*, while transliterating a loan-word written in a foreign language back to the language of its origin is called *Backward Transliteration* (Karimi et al., 2011).

Statistical techniques based on large parallel transliteration corpora work well for rich-resource languages but low-resource languages do not have the luxury of such resources. For such languages, rule-based transliteration is the only viable option.

From 2009 to 2018, various transliteration systems were proposed during the Named Entities Workshop evaluation campaigns[1] (Duan et al., 2016). These campaigns consist in transliterating from English into languages with a wide variety of writing systems, including Hindi, Tamil, Russian, Kannada, Chinese, Korean, Thai and Japanese. We can see that the romanization of non-Latin writing systems remains a complex computational task that depends crucially on which language is involved. Through this workshop, much progress has been made in methodologies for resolving the transliteration of proper nouns. We see the emergence of different approaches, such as grapheme-to-phoneme conversion (Finch and Sumita, 2010; Ngo et al., 2015), based on statistics like machine translation (Laurent et al., 2009; Nicolai et al., 2015) and neural networks (Finch et al., 2016; Shao and Nivre, 2016; Thu et al., 2016). Other work used attention-less sequence-to-sequence models for the transliteration task (Yao and Zweig, 2015). One study used a bidirectional Long Short-Term Memory (LSTM) models together with input delays for grapheme-to-phoneme conversion (Rao et al., 2015).

Another important challenge with the extraction of named entities and automatic transliteration is related to the vast variety of writing systems. All these difficulties are aggravated by the lack of bilingual pronunciation dictionaries for proper nouns, ambiguous transcriptions and orthographic variation in a given language. In addition to transliteration generation systems, there are also transliteration mining systems that try to obtain parallel transliteration pairs from comparable corpora (Klementiev and Roth, 2006; Kumaran et al., 2010; Sajjad et al., 2017; Tran et al., 2016; Udupa et al., 2009).

In our literature review, we found a few cases in which Vietnamese had been studied for the transliteration task. (Cao et al., 2010) applied the statistical-based approach as machine translation in the transliteration task for the English-Vietnamese low-resource language pair, with a performance of 63 BLEU points. (Ngo et al., 2015) proposed a statistical model for English and Vietnamese, with a phonological constraint on syllables. Their system performed better than the rule-based baseline system, with a 70% reduction in error rates. (Le and Sadat, 2017) explored RNN, particularly, LSTM, in the transliteration task for French and Vietnamese. Their results showed that the RNN-based system performed better than the baseline system, which was based on a statistical approach. In this research, we propose a new approach by using alignment representation for input sequences and pre-trained source/target embeddings in the input layer in order to build a neural network-based transliteration system to solve the problem of scattered data due to a low-resource language.

3 Methodology

Our proposed approach for an efficient transliteration consists of three main steps: *(1) pre-processing, (2) modification of the input sequences based on alignment representation and (3) creation of an RNN-based machine transliteration*. The whole pipeline is illustrated in Figure 1.

(1) Firstly, the learning data is pre-processed with normalization in lowercasing, removing the hyphens separating syllables and segmenting all syllables at the character level.

(2) Secondly, we extract the alignment output from the bilingual pronunciation dictionary and modify the input sequences based on the alignment results (Figure 1).

(3) Then we train an RNN-based machine transliteration (Figure 2).

4 Experiments

4.1 Configuration

To evaluate the efficiency of our proposed transliteration system in low resource settings, we used a bilingual pronunciation dictionary that has been provided by the NEWS 2018 shared task[2]. The

[1] http://workshop.colips.org/news2016/

[2] http://workshop.colips.org/news2018/documents/news2018whitepaper.pdf

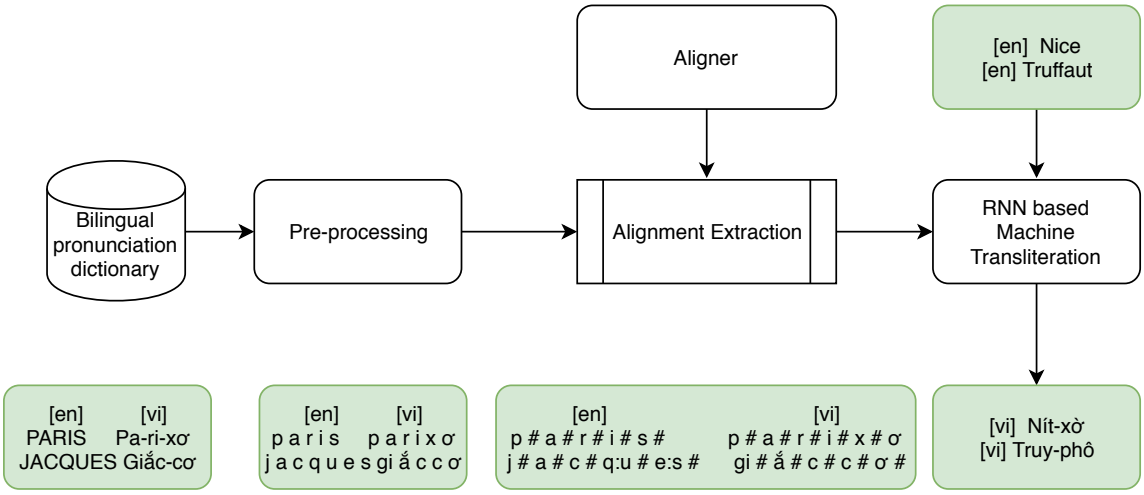

Figure 1: The architecture of machine transliteration for a low-resource language pair dealing with bilingual named entities.

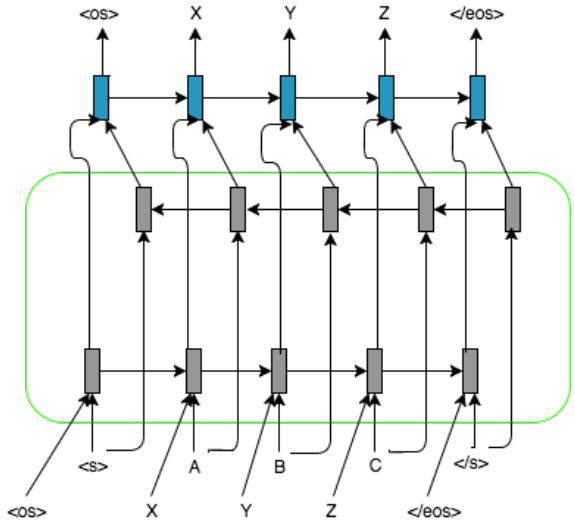

Figure 2: Our RNN-based model architecture with encoder-decoder bi-directional LSTM and alignment representation on input sequences. We use <s> and </s>, <os> and </eos> markers to pad the grapheme/phoneme sequences to a fixed length.

learning data comprise 3,256 pairs of bilingual English-Vietnamese named entities pairs, 500 pairs for the development set and 500 pairs for the testing set. We found that most of the named entities were persons, locations and organizations. To overcome the problem of the scattering of learning data, we performed the pre-processing step with segmentation of all syllables at the character level and presented the whole dataset in lowercase.

To deal with the alignment representation, we used the *m-2-m aligner*[3] toolkit (Jiampojamarn et al., 2007) to align the training data at the character level. We chose $m = 2$ (bigram-align) for all experiments; this means that a maximum of two graphemes on the source side will be aligned with a maximum of two phonemes on the target side. For the pre-trained source and target embeddings, we applied the word2vec[4] toolkit (Mikolov et al., 2013) with a dimension of 64, a continuous space window size of 5 and the 'skip-gram' option.

We applied the *nmt-keras*[5] toolkit to train our transliteration model for the English-Vietnamese language pair. In the transliteration system configuration, we used two-layer encoder-decoder bi-directional LSTM cells (Hochreiter and Schmidhuber, 1997) for the RNN model, with a 64-dimension projection layer to encode the input sequences and 128 nodes in each hidden layer. We used the *'Adam'* optimizer to learn the weights of the network with a default learning rate of 0.001. For decoding, the beam search was assigned the size of 6. All the RNN hyper-parameters were determined by tuning on the development set. This implementation is based on Python *Theano* (Al-Rfou et al., 2016), which allows for efficient training on both central processing units (CPU) and graphics processing units (GPU).

[3] https://github.com/letter-to-phoneme/m2m-aligner/

[4] https://code.google.com/archive/p/word2vec/

[5] https://github.com/lvapeab/nmt-keras/

4.2 Evaluation

In this work, we built a machine transliteration method which was inspired by neural machine translation. Hence, we applied different evaluation metrics such as *BiLingual Evaluation Understudy (BLEU)* (Papineni et al., 2002), *Translation Error Rate (TER)* (Snover et al., 2009), and *Phoneme Error Rate (PER)*.

To evaluate our proposed approach, we implemented five systems (Table 1):

(1) Baseline system A : *phrase-based statistical machine translation (pbSMT)*.
We implemented a *pbSMT* system with *Moses*[6] (Koehn et al., 2007). We used *mGIZA* (Gao and Vogel, 2008) to align the corpus at the character level, and *SRILM* (Stolcke et al., 2002) to create a character-based 5-gram language model for the target language.

(2) Baseline system B : *multi-joint sequence model for grapheme-to-phoneme convertion*.
We applied the *Sequitur-G2P*[7] toolkit to train a transliteration model.

(3) System 1 : *encoder-decoder bidirectional + attention mechanism*.

(4) System 2 : *encoder-decoder bidirectional + attention mechanism + alignment representation for input sequences*.

(5) System 3 : *encoder-decoder bidirectional + attention mechanism + alignment representation for input sequences + pre-trained source and target embeddings*.

The difference between the two baseline systems' performance is minor. Baseline system B seems slightly more efficient than baseline system A, with a gain of +4.40 BLEU points, as well as reduced translation errors (TER), at -3.58 points and phoneme errors (PER), at -6.20 points (Table 1).

By comparing the two baseline systems and systems 1, 2 and 3 (*our proposed approach*), we note significant results up to 68.60 points for BLEU, and reductions in TER and PER up to 15.92 and 30.03 points, respectively (Table 1).

In addition, system 3 performed better than systems A and B, with gains of +7.30 and +2.90 BLEU points, reductions of -8.16 and -4.58 TER points, -14.17 and -7.97 PER points, respectively (Table 1).

In general, the proposed approach performed the transliteration task very well, with significant gains, and reduced the phoneme error rate. We observed that the output quality of the proposed approach, based on recurrent neural networks, was more fluid, coherent and had fewer errors than other systems, that use statistical-based approaches (Table 2).

All the experimental results showed that using the alignment representation and the pre-trained source and target embeddings resulted in significant advances over other methods.

5 Conclusions and perspectives

In this paper, we presented a novel approach for machine transliteration in low research settings, that combines several techniques based on neural networks - encoder-decoder, attention mechanism, alignment representation for input sequences and pre-trained source and target embeddings - in machine transliteration systems.

In the future work, we intend to test our proposed approach with a larger bilingual pronunciation dictionary as well as to study other approaches such as semi-supervised or non-supervised.

Acknowledgements

We thank the anonymous reviewers for their insightful comments.

References

Rami Al-Rfou, Guillaume Alain, Amjad Almahairi, Christof Angermueller, Dzmitry Bahdanau, Nicolas Ballas, Frédéric Bastien, Justin Bayer, Anatoly Belikov, Alexander Belopolsky, et al. 2016. Theano: A Python framework for fast computation of mathematical expressions. *arXiv preprint arXiv:1605.02688* 472 (2016), 473.

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

Nam X Cao, Nhut M Pham, and Quan H Vu. 2010. Comparative analysis of transliteration techniques based on statistical machine translation and joint-sequence model. In *Proceedings of the 2010 Symposium on Information and Communication Technology*. Association for Computing Machinery, 59–63.

[6] http://www.statmt.org/moses/

[7] https://www-i6.informatik.rwth-aachen.de/web/Software/g2p.html

Experiments	BLEU ↑	TER ↓	PER ↓
Baseline system A (pbSMT)	61.30	24.08	44.20
Baseline system B (Sequitur-G2P)	65.70	20.50	38.00
System 1 (encoder-decoder + attention mechanism)	66,68	16,70	31,50
System 2 (encoder-decoder + attention mechanism + alignment representation)	67,57	16,23	30,63
System 3 (encoder-decoder + attention mechanism + alignment representation + pre-trained source target embeddings)	**68,60**	**15,92**	**30,03**

Table 1: Evaluation of scoring for all systems : BLEU, TER and PER.

PARIS			MANHATTAN		
No	**TOP-5**	**Probability**	**No**	**TOP-5**	**Probability**
1	**p a r i x ʊ**	**0,633242**	**1**	m a n h á t t â n	0,321082
2	p a r í t	0,153536	**2**	**m a n h á t t a n**	**0,288677**
3	b a r i	0,065151	**3**	m â n h á t t â n	0,080221
4	b a r í t	0,037314	**4**	m â n h á t t a n	0,072125
5	b a r í t x ʊ	0,028526	**5**	m a h á t t â n	0,058193

Table 2: Illustration of the transliteration predictions of the named entities obtained by our proposed approach before the re-ranking of the list of k-best results, with the top-5 ($k = 5$) first best results for the named entities : *PARIS* and *MANHATTAN*

Xiangyu Duan, Rafael E Banchs, Min Zhang, Haizhou Li, and A Kumaran. 2016. Report of NEWS 2016 Machine Transliteration Shared Task. *ACL 2016* (2016), 58–72.

Andrew Finch, Lemao Liu, Xiaolin Wang, and Eiichiro Sumita. 2016. Target-Bidirectional Neural Models for Machine Transliteration. *ACL 2016* (2016), 78–82.

Andrew Finch and Eiichiro Sumita. 2010. Transliteration using a phrase-based statistical machine translation system to re-score the output of a joint multigram model. In *Proceedings of the 2010 Named Entities Workshop*. Association for Computational Linguistics, 48–52.

Qin Gao and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*. Association for Computational Linguistics, 49–57.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (Nov. 1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Sebastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2014. On using very large target vocabulary for neural machine translation. *arXiv preprint arXiv:1412.2007* (2014).

Sittichai Jiampojamarn, Grzegorz Kondrak, and Tarek Sherif. 2007. Applying Many-to-Many Alignments and Hidden Markov Models to Letter-to-Phoneme Conversion. In *HLT-NAACL*, Vol. 7. 372–379.

Sarvnaz Karimi, Falk Scholer, and Andrew Turpin. 2011. Machine transliteration survey. *ACM Computing Surveys (CSUR)* 43, 3 (2011), 17.

Alexandre Klementiev and Dan Roth. 2006. Weakly supervised named entity transliteration and discovery from multilingual comparable corpora. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 817–824.

Kevin Knight and Jonathan Graehl. 1998. Machine transliteration. *Computational Linguistics* 24, 4 (1998), 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*. Association for Computational Linguistics, 177–180.

A Kumaran, Mitesh M Khapra, and Haizhou Li. 2010. Report of NEWS 2010 transliteration mining shared task. In *Proceedings of the 2010 Named Entities Workshop*. Association for Computational Linguistics, 21–28.

Antoine Laurent, Paul Deléglise, Sylvain Meignier, and France Spécinov-Trélazé. 2009. Grapheme to phoneme conversion using an SMT system. In *Proceedings of INTERSPEECH, ISCA*. 708–711.

Ngoc Tan Le and Fatiha Sadat. 2017. A Neural Network Transliteration Model in Low Resource Settings. In *Proceedings of the 16th International Conference of Machine Translation Summit, September 18-22 2017, Nagoya, Japan, volume 1, Research Track*. 337–345.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations.. In *hlt-Naacl*, Vol. 13. 746–751.

Hoang Gia Ngo, Nancy F Chen, Binh Minh Nguyen, Bin Ma, and Haizhou Li. 2015. Phonology-augmented statistical transliteration for low-resource languages.. In *Interspeech*. 3670–3674.

Garrett Nicolai, Bradley Hauer, Mohammad Salameh, Adam St Arnaud, Ying Xu, Lei Yao, and Grzegorz Kondrak. 2015. Multiple system combination for transliteration. In *Proceedings of NEWS 2015 The Fifth Named Entities Workshop*. 72–79.

Jong-Hoon Oh, Key-Sun Choi, and Hitoshi Isahara. 2006. A machine transliteration model based on correspondence between graphemes and phonemes. *ACM Transactions on Asian Language Information Processing (TALIP)* 5, 3 (2006), 185–208.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 311–318.

Kanishka Rao, Fuchun Peng, Haşim Sak, and Françoise Beaufays. 2015. Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 4225–4229.

Hassan Sajjad, Helmut Schmid, Alexander Fraser, and Hinrich Schütze. 2017. Statistical models for unsupervised, semi-supervised and supervised transliteration mining. *Computational Linguistics* (2017).

Yan Shao and Joakim Nivre. 2016. Applying Neural Networks to English-Chinese Named EntityTransliteration. In *Sixth Named Entity Workshop, joint with 54th ACL*.

Matthew G Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. 2009. TER-Plus: paraphrase, semantic, and alignment enhancements to Translation Edit Rate. *Machine Translation* 23, 2-3 (2009), 117–127.

Andreas Stolcke et al. 2002. SRILM-an extensible language modeling toolkit.. In *Interspeech*, Vol. 2002. 2002.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*. 3104–3112.

Ye Kyaw Thu, Win Pa Pa, Yoshinori Sagisaka, and Naoto Iwahashi. 2016. Comparison of Grapheme–to–Phoneme Conversion Methods on a Myanmar Pronunciation Dictionary. *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing 2016* (2016), 11–22.

Phuoc Tran, Dien Dinh, and Hien T Nguyen. 2016. A Character Level Based and Word Level Based Approach for Chinese-Vietnamese Machine Translation. *Computational intelligence and neuroscience* 2016 (2016).

Raghavendra Udupa, K Saravanan, A Kumaran, and Jagadeesh Jagarlamudi. 2009. Mint: A method for effective and scalable mining of named entity transliterations from large comparable corpora. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 799–807.

Sonjia Waxmonsky and Sravana Reddy. 2012. G2P conversion of proper names using word origin information. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 367–371.

Kaisheng Yao and Geoffrey Zweig. 2015. Sequence-to-sequence neural net models for grapheme-to-phoneme conversion. *arXiv preprint arXiv:1506.00196* (2015).