Comparison between Neural and Statistical translation after transliteration of Algerian Arabic Dialect

Imane GUELLIL
Ecole Supérieure d'Informatique ESI Alger
Ecole préparatoire des sciences et techniques Alger
i_guellil@esi.dz
i.guellil@epsta.dz

Faical AZOUAOU
Ecole Supérieure d'Informatique ESI Alger
f_azouaou@esi.dz

Mourad ABBAS
Centre de Recherche Scientifique et Technique pour le Développement de la Langue Arabe (CRSTDLA)
m_abbas04@yahoo.fr

Abstract

Research on Arabic Dialect Treatment has recently become important in the literature. Although most work on these dialects considers only the messages or the portion of text written in Arabic letters, another style of writing has emerged on social media. This style is known by Arabizi and combines between Latin letters and numbers. To address this emergent problem in the context of automatic translation, we present an Arabic dialect translation system composed by two modules: Transliteration and translation. We develop each module with a statistical and a neural model. To test our system, we used the Algerian portion of a multi-dialectal Arabic corpus named PADIC. Experimental results show that a good transliteration improves the translation results. Moreover, the neural transliteration gives better results than the statistical transliteration. However, the statistical translation still gives better results that the neural translation.

1 Introduction

Machine Translation (MT) represents an active researcher area (Chand 2016). Just recently, a new approach has emerged that involves neural networks. This approach is known as Neural Machine Translation (NMT) (Sutskever et al. 2014; Cho et al. 2014a; Bahdanau et al. 2014). Unfortunately, the work on NMT has not focused yet on Arabic language and its dialects. Among all the work on NMT, we were able to find only one paper that describes NMT on the Arabic language (Almahairi et al. 2016) and no work involving NMT on Arabic Dialects.

Nowadays, users in social media write in this way:

1) By using only Arabic letters for example, " حبيت نسقسيكم شحال يدير ايفون6من فضلكم", which means, "I want to ask you, what is the price of iphone6 please.

2) By combining between Latin letters and numbers. For example: "walahi rabi ykon fi el3awn" which means: "I swear god will help you. This way of writing recognized by "Arabizi", (Darwish 2014).The work in (Bies et al. 2014) considers Arabizi as a challenge for Arabic NLP research. To address this challenge, we consider Arabizi Transliteration as the first module (or as pre-processing

step) of Arabic dialect treatment where the analyzed messages combining between the Arabizi and the Arabic letters. We survey a lot of work on Statistical Machine transliteration (SMTR) (van der Wees et al. 2016; Al-Badrashiny et al. 2014; Darwish 2014) and other combing between transliteration and translation (May et al. 2014) and (van der Wees et al. 2016). However, the literature has not contained any work related to Neural Machine Transliteration (NMTR) of Arabizi or related to. To address this problem, we propose an Arabic dialect translation system composed of two components or modules: The first one for Arabizi transliteration and the second one for Arabic dialect translation.

2 Related work

The work of (May et al. 2014) and (van der Wees et al. 2016) present an Arabizi to English Statistical Machine Translation. Despite the fact that the two works do not focus on transliteration of Arabizi to Arabic but also evaluate the performance of MT system, they differ in two points: 1) the first one constructs a transliterated corpus semi-automatically, with input from experts, while the second one constructs it automatically. 2) The first one learns weights of character from an Arabizi-Arabic text while the second one uses uniform weights.
However, we have not found any work that combines NMTR and SMT or NMT for Arabizi.

3 The Arabic dialect translation framework

The general idea of this approach is to transliterate an Arabizi corpus with SMTR and NMTR techniques and translate the transliterated corpus.

3.1 The transliteration step

To transliterate a given text written in Arabizi to the same text written in the Arabic alphabet, we follow four main sub-steps:

1) We construct a parallel Arabizi corpus containing 6233 sentences. We based our work on PADIC (Meftouh et al. 2015) (which is written in Arabic letters), which we transliterated to Arabizi. To do that, we first define a rule-based algorithm to automatically transliterate Arabic Dialect written in Arabic letters to Arabizi form. This algorithm transforms the letter (ع) to the number (3), the letter (غ) to the two letters (gh),…etc. Unfortunately, at this stage, we can only correct 1300 sentences.

2) Based on the work of (Darwish 2014), we divide each sentence to a set of word and each word to a set of characters, so we work at the character level.

3) We apply an SMT-based phrase on our data. These data are first trained using a language model. The language model is built with the target language (in our case, Arabic Dialect written with Arabic letters). For training the transliteration model, we run a character based-alignment. We finish by the tuning process, for determining the best results for each transliteration pair.

4) We also apply to the same data an NMT model. In this paper, we opt to use RNN Encoder-decoder model. The RNN Encoder-decoder proposed by (Cho et al. 2014a) and (Sutskever et al. 2014). The choice to use an RNN Encoder-decoder is mainly due to the fact that this model is considered as the simplest version of neural machine translation. To train this model, we firstly replace some unknown characters by the term “unk”. We use a development set separated from the training set to measure how well the model generalizes during training. Finally we use an external lexicon indicating the mapping between characters and their probabilities. To create this lexicon, we use a word alignment tool (character-based)(Neubig 2016).Neural Machine Transliteration based on a character level.

In this paper, we choose to use RNN Encoder-decoder model. To train this model, we first replace some unknown character by the term “unk” then, we use a development set separating from training set to measure how well the model is generalizing during the training. Finally, we use an external lexicon indicating mapping between character and their probabilities. To create this lexicon, we use a character Alignment (Neubig 2016).

3.2 The translation step

The main idea of this step is to translate Arabic Dialect to MSA. This will allow us, in the future, to consider MSA as a pivot and translate to English or French. We assume that each sentence is written in Arabic letters only or Arabizi only. We do not treat, in this paper, the case where we find an Arabic letter and Arabizi in the same sentence. We leave this problem for future work. This component could take as an input arabizi messages after transliteration or the messages written with Arabic letters. So it can receive as the input messages provided from our Arabic dialect corpus or a set of messages that we transliterate before (so the output of the transliteration component). In this step, we follow three main sub-steps. 1) We begin by reassembling the words of the transliterated corpus. This is due to the fact that transliteration is word-based level and translation is phrase-based level. However, we do not need to reassemble in the case of Arabic dialect translation (when corpus written with Arabic letters), as shownin Fig.1.2) We apply an SMT model to the resulting sen-tences As in the transliteration task, we have to build the language model, train it by running a word-level Alignment and call the tuning process.

3) We also apply an NMT model to the same sentences. We also use the use RNN Encoder-decoder model. We follow the same steps as the transliteration, so we detect the unknown words and train the model and create an aligned lexicon. The unique difference compared the transliteration is that the model is phrase-based and not word-based. model.

4 Experiments and results

Our System is composed by two components: The transliteration and the translation component. For each one, we apply the statistical and neural models. Concerning Statistical model, we use Moses toolkit (Koehn et al. 2007), with KenLM (Heafield 2011) as language model and GIZA++ as alignment tool (Och and Ney 2000). Concerning Neural model, based on (Neubig 2016), we use Lamtram toolkit (Neubig 2015), which is the combination of the of the two model(Bahdanau et al. 2014) and (Luong et al. 2015). Before utilizing lamtram toolkit, we have to install dynet library.

As shown in Table 1, we conduct our experiments on 4 distinct training data sets. They differ in size. For each data set, we present the transliteration and the translation results. For the transliteration, we consider the statistical (SMTR) and Neural (NMTR) transliteration. For translation too, we consider statistical (SMT) and neural (NMT) translation. We observe that SMT gives better results than NMT. Moreover, SMT work well where it is combined with SMTR.

To show the utility of proceeding to transliteration before translation, we conduct a SMT on the Arabizi corpus test without transliterate it. We carry out this experiment for the biggest training corpus (so 100% of the total size). We obtained a bleu score= 4.26 where the score after SMTR= 6.01 and the reference= 10.74.

Trainig corpus size	Transliteration	Translation	BLEU score
10%	Reference	SMT	6.31
		NMT	0.00
	SMTR	SMT	2.65
		NMT	0.00
	NMTR	SMT	2.40
		NMT	0.0
25%	Reference	SMT	8.02
		NMT	1.71
	SMTR	SMT	3.47
		NMT	0.00
	NMTR	SMT	4.49
		NMT	0.0
50%	Reference	SMT	10.02
		NMT	2.34
	SMTR	SMT	5.21
		NMT	0.0
	NMTR	SMT	4.21
		NMT	0.0
100%	Reference	SMT	**10.74**
		NMT	6.25
	SMTR	SMT	**6.01**
		NMT	4.54
	NMTR	SMT	3.94
		NMT	4.13

Table 1: SMT Vs NMT of Arabizi

5 Conclusion and Perspectives

We present and implement an approach composed by two components: Transliteration and translation. We consider the statistical and neural transliteration and translation. Through this paper, we observe that for a small corpus of Arabizi, neural machine transliteration gives better results than statistical transliteration, whereas statistical translation still gives better results than neural translation.

In future work, we will try to generalize this idea by testing our system on other corpora like on Cotterell et al. (Cotterell et al. 2014) corpora.

References

Al-Onaizan, Yaser, and Kevin Knight
2002 Machine transliteration of names in Arabic text. Proceedings of the ACL-02 workshop on Computational approaches to semitic languages, 2002, pp. 1-13. Association for Computational Linguistics.

Almahairi, Amjad, et al.
2016 First Result on Arabic Neural Machine Translation. arXiv preprint arXiv:1606.02680.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio
2014 Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

Bhalla, Deepti, Nisheeth Joshi, and Iti Mathur
2013 Rule based transliteration scheme for English to Punjabi. arXiv preprint arXiv:1307.4300.

Chalabi, Achraf, and Hany Gerges
2012 Romanized arabic transliteration.

Cho, Kyunghyun, et al.
2014a On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259.

Cho, Kyunghyun, et al.
2014b Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

Cotterell, Ryan, et al.
2014 An algerian arabic-french code-switched corpus. Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools Workshop Programme, 2014, pp. 34.

Darwish, Kareem
2014 Arabizi Detection and Conversion to Arabic. ANLP 2014:217.

Darwish, Kareem, and Walid Magdy
2014 Arabic information retrieval. Foundations and Trends® in Information Retrieval 7(4):239-342.

Dyer, Chris, Victor Chahuneau, and Noah A Smith
2013 A simple, fast, and effective reparameterization of ibm model 2, 2013. Association for Computational Linguistics.

Finch, Andrew, et al.
2015 Neural network transduction models in transliteration generation. Proceedings of NEWS 2015 The Fifth Named Entities Workshop, 2015, pp. 61.

Goldberg, Yoav
2015 A primer on neural network models for natural language processing. arXiv preprint arXiv:1510.00726.

Habash, Nizar, Abdelhadi Soudi, and Timothy Buckwalter
2007 On arabic transliteration. *In* Arabic computational morphology. Pp. 15-22: Springer.

Heafield, Kenneth
2011 KenLM: Faster and smaller language model queries. Proceedings of the Sixth Workshop on Statistical Machine Translation, 2011, pp. 187-197. Association for Computational Linguistics.

Jadidinejad, Amir H
2016 Neural Machine Transliteration: Preliminary Results. arXiv preprint arXiv:1609.04253.

Josan, Gurpreet Singh, and Gurpreet Singh Lehal
2010 A Punjabi to Hindi Machine Transliteration System. Computational Linguistics and Chinese Language Processing 15(2):77-102.

Joshi, Hardik, Apurva Bhatt, and Honey Patel
2013 Transliterated Search using Syllabification Approach. Forum for Information Retrieval Evaluation, 2013.

Kaur, Kamaljeet, and Parminder Singh
2014 Review of Machine Transliteration Techniques. International Journal of Computer Applications 107(20).

Kikuchi, Yuta, et al.
2016 Controlling output length in neural encoder-decoders. arXiv preprint arXiv:1609.09552.

Kingma, Diederik, and Jimmy Ba
2014 Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Koehn, Philipp, et al.
2007 Moses: Open source toolkit for statistical machine translation. Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions, 2007, pp. 177-180. Association for Computational Linguistics.

Luong, Minh-Thang, Hieu Pham, and Christopher D Manning
2015 Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025.

Malik, MG Abbas, et al.
2013 Urdu Hindi machine transliteration using SMT. WSSANLP-2013:43.

May, Jonathan, Yassine Benjira, and Abdessamad Echihabi
2014 An Arabizi-English social media statistical machine translation system. Proceedings of the 11th Conference of the

Association for Machine Translation in the Americas, 2014, pp. 329-341.
Meftouh, Karima, et al.
2015 Machine Translation Experiments on PADIC: A Parallel Arabic DIalect Corpus. The 29th Pacific Asia Conference on Language, Information and Computation, 2015.
Neubig, Graham
2015 lamtram: A toolkit for language and translation modeling using neural networks.
—
2016 Lexicons and minimum risk training for neural machine translation: Naist-cmu at wat2016. arXiv preprint arXiv:1610.06542.
Och, Franz Josef, and Hermann Ney
2000 Giza++: Training of statistical translation models.
Oh, Jong-Hoon, and Key-Sun Choi
2005 An ensemble of grapheme and phoneme for machine transliteration. International Conference on Natural Language Processing, 2005, pp. 450-461. Springer.
Sutskever, Ilya, Oriol Vinyals, and Quoc V Le
2014 Sequence to sequence learning with neural networks. Advances in neural information processing systems, 2014, pp. 3104-3112.
van der Wees, Marlies, Arianna Bisazza, and Christof Monz
2016 A Simple but Effective Approach to Improve Arabizi-to-English Statistical Machine Translation. WNUT 2016:43.
Zaidan, Omar F, and Chris Callison-Burch
2014 Arabic dialect identification. Computational Linguistics 40(1):171-202.