

Neural Poetry Translation

Marjan Ghazvininejad[†], Yejin Choi^{‡,♠}, and Kevin Knight[†]

[†]Information Sciences Institute & Computer Science Department
University of Southern California
{ghazvini, knight}@isi.edu

[‡]Paul G. Allen School of Computer Science & Engineering, University of Washington

[♠]Allen Institute for Artificial Intelligence
yejin@cs.washington.edu

Abstract

We present the first neural poetry translation system. Unlike previous works that often fail to produce any translation for fixed rhyme and rhythm patterns, our system always translates a source text to an English poem. Human evaluation ranks translation quality as acceptable 78.2% of the time.

1 Introduction

Despite recent improvements in machine translation, automatic translation of poetry remains a challenging problem. This challenge is partially due to the intrinsic complexities of translating a poem. As Robert Frost says “Poetry is what gets lost in translation”. Nevertheless, in practice poems have always been translated and will continue to be translated between languages and cultures.

In this paper, we introduce a method for automatic poetry translation. As an example, consider the following French poem:

French poem:

Puis je venais masseoir pr’ès de sa chaise
Pour lui parler le soir plus ‘a mon aise.

(Literally:

Then I came to sit near her chair
To discuss with her the evening more
at my ease.)

Our goal is to translate this poem into English, but also to obey target rhythm and rhyme patterns specified by the user, such as *2-line rhyming iambic pentameter*, ten syllables per line with alternating stress 0101010101, where 0 represents an unstressed syllable, and 1 represents a stressed syllable. Lines strictly rhyme if their pronunciations match from the final stressed vowel onwards; slant rhyming allows variation. Overall, this is a difficult task even for human translators.

In spite of recent works in automatic poetry generation (Oliveira, 2012; He et al., 2012; Yan

et al., 2013; Zhang and Lapata, 2014; Yi et al., 2017; Wang et al., 2016; Ghazvininejad et al., 2016, 2017; Hopkins and Kiela, 2017; Oliveira, 2017), little has been done on automatic poetry translation. Greene et al. (2010) use phrase-based machine translation techniques to translate Italian poetic lines to English-translation lattices. They search these lattices for the best translation that obeys a given rhythm pattern. Genzel et al. (2010) also use phrase-based machine translation technique to translate French poems to English ones. They apply the rhythm and rhyme constraints during the decoding process. Both methods report total failure in generating any translations with a fixed rhythm and rhyme format for most of the poems. Genzel et al. (2010) report that their method can generate translations in a specified scheme for only 12 out of 109 6-line French stanzas.

This failure is due to the nature of the phrase-based machine translation (PBMT) systems. PBMT systems are bound to generate translations according to a learned bilingual phrase table. These systems are well-suited to unconstrained translation, as often the phrase table entries are good translations of source phrases. However, when rhythm and rhyme constraints are applied to PBMT, translation options become extremely limited, to the extent that it is often impossible to generate *any* translation that obeys the poetic constraints (Greene et al., 2010). In addition, literal translation is not always desired when it comes to poetry. PBMT is bound to translate phrase-by-phrase, and it cannot easily add, remove, or alter details of the source poem.

In this paper, we propose the first neural poetry translation system and show its quality in translating French to English poems. Our system is much more flexible than those based on PBMT, and is always able to produce translations into any scheme. In addition, we propose two novel im-

provements to increase the quality of the translation while satisfying specified rhythm and rhyme constraints. Our proposed system generates the following translation for the French couplet mentioned above:

French poem:

Puis je venais masseoir pr'ès de sa chaise
Pour lui parler le soir plus 'a mon aise.

Our system:

And afterwards I came to sit together.
To talk about the evening at my pleasure.

2 Data

We use a French translation of Oscar Wilde's *Ballad of Reading Gaol* (Wilde, 2001) by Jean Guiloineau¹ as our input poem, and the original Wilde's poem as the human reference. This test set contains 109 6-line stanzas, 29 of which we use for development. For each stanza, we require our machine translation to produce odd lines with iambic tetrameter and even lines with iambic trimeter, with even lines (2, 4, 6) rhyming.

3 Proposed Method

3.1 Model A: Initial Model

Unconstrained Machine Translation. The base of our poetry translation system is an encoder-decoder sequence-to-sequence model (Sutskever et al., 2014) with a two-layer recurrent neural network (RNN) with long short-term memory (LSTM) units (Hochreiter and Schmidhuber, 1997). It is pre-trained on parallel French-English WMT14 corpus.² Specifically, we use 2-layer LSTM cells with 1000 hidden cells for each layer. For pre-training, we set the dropout ratio to 0.5. Batch size is set to 128, and the learning rate is initially set as 0.5 and starts to decay by 0.5 when the perplexity of the development set starts to increase. Gradients are clipped at 5 to avoid gradient explosion. We stop pre-training the system after 3 epochs. In order to adapt the translation system to in-domain data, we collect 16,412 English songs with their French translations and 12,538 French songs with their English translations (6M word tokens in total) as our training corpus,³ and continue training the system (warm start)⁴ with this dataset.

¹<https://bit.ly/2GN1ZGk>

²<http://www.statmt.org/wmt14/translation-task.html>

³<http://lyricstranslate.com/>

⁴We continue training the system while we set dropout ratio to 0.2, and keep the other settings fixed.

This encoder-decoder RNN model is used to generate the unconstrained translation of the poems.

Enforcing Rhythm in Translation. To enforce the rhythm constraint, we adopt the technique of Ghazvininejad et al. (2016). We create a large finite-state acceptor (FSA) that compactly encodes all word sequences that satisfy the rhythm constraint. In order to generate a rhythmic translation for the source poem, we constrain the possible LSTM translations with this FSA. To do so, we alter the beam search of the decoding phase of the neural translation model to only generate outputs that are accepted by this FSA.

Enforcing Rhyme in Translation. Ghazvininejad et al. (2016) fix the rhyme words in advance and build an FSA with the chosen rhyme words in place. Unlike their work, we do not fix the rhyme words in the FSA beforehand, but let the model choose rhyme words during translation. We do so by partitioning the vocabulary into rhyme classes and building one FSA for each class. This FSA accepts word sequences that obey the rhythm pattern and end with any word within the corresponding rhyme class. Then we translate each line of the source poem multiple times, once according to each rhyme class. In the final step, for each set of rhyming lines, we select a set of translations that come from the same rhyme class and have the highest combined translation score. In practice, we just make FSAs for each of the 100 most frequent rhyme classes (out of 1505), which covers 67% of actual rhyming word tokens in our development set.

3.2 Model B: Biased Decoding with Unconstrained Translation

Naive application of rhythm and rhyme constraints to the neural translation system limits the translation options of the system. Sometimes the beam search finds no related translation that satisfies the constraints, forcing the decoder to choose an unrelated target-language token. The system does not have a way to recover from this situation, and continues to generate a totally unrelated phrase. An example is rhythm- and rhyme-constrained translation of "Et buvait l'air frais jusqu'au soir" ("And drinking fresh air until the evening") to "I used to close my hair" by our initial system (Figure 1).

We therefore propose to use the output of *unconstrained* translation as a guideline for the constrained translation process. To do so, we encour-

age the words that appear in the unconstrained translation during the decoding step of the constrained one. We encourage by multiplying their RNN log probabilities by 5 during beam search. Figure 1 shows how this technique addresses the problem.

<p>French poem: Sans mains tordues, comme ces hommes, Ces pauvres hommes sans espoir, Qui osent nourrir l'espérance Dans le caveau du désespoir: Il regardait vers le soleil Et buvait l'air frais jusqu'au soir.</p> <p>Human reference: He did not wring his hands, as do Those witless men who dare To try to rear the changeling Hope In the cave of black Despair: He only looked upon the sun, And drank the morning air.</p> <p>Unconstrained machine translation: Like these men These poor men without hope, Who dare to feed the hope. In the vault of despair He was looking to the sun And drinking fresh air until the evening.</p> <p>Rhythmic and rhyming translation by model A: Without a crooked hand as men. These hopeless people there. Who dare to feed the expectations. Surrounded by despair. He only looking at the sun. <i>I used to close my hair.</i></p> <p>Rhythmic and rhyming translation by model B: Without a crooked hand as men. These hopeless people there. Who dare to feed the expectations. Surrounded by despair. He only looking at the sun. <i>Was drinking fresh of air.</i></p>

Figure 1: An example of poetry translation by models A and B. Biased decoding with unconstrained translation (Model B) produces a better translation compared to the baseline rhythm- and rhyme-constrained system.

3.3 Model C: Biased Decoding with All Potential Translation

Our poetry translation system is also challenged by rare words for which the system has not learned a good translation. The unconstrained system produces a special <UNK> token for these cases, but

the FSA does not accept <UNK>, as it is not pronounceable. We can let the system produce its next guess instead, but <UNK> is a sign that the translation system is not sure about the source meaning.

To overcome this problem, we use an idea similar to model B. This time, *in addition to encouraging the unconstrained translated words*, we encourage all potential translations of the foreign words. To get the potential translations, we use the translation table (t-table) extracted from parallel French-English training data using Giza++ (Och and Ney, 2003). This way, the system receives an external signal that guides it toward selecting better translations for the rare foreign word. We run five iterations of each of IBM models 1, 2, HMM, and 4 to get the t-table. An example of how this method improves the poem quality over model B can be observed in the fifth line of the poems in Figure 2.

4 Results

Our first experiment compares model A with model B. These systems generated non-identical translations for 77 (out of 80) of the test stanzas. We ask 154 Amazon mechanical turk judges to compare these translations (each pair of translations was compared twice). We present the judges with the French poem for reference and did not mention that the poems are computer generated. Judges have the option to prefer either of the poems or state they cannot decide. The results in Table 1 clearly show that the model B generates better translations.

In the second experiment, we compare model B with model C. We have 84 judges compare 42 different poems. Table 2 shows that judges preferred the outputs of model C by a 17.7% margin.

We also ask 238 judges to rank translations of all the 80 stanzas of the test set as very bad, bad, ok, good, and very good. Table 3 shows the distribution of this ranking. We see that 78.2% of the judges ranked the output ok or more (49.6% of the poems were ranked good or very good). Figure 3 shows an example of the poems ranked as very good.

5 Conclusion

In this paper we presented the first neural poetry translation system and provided two novel methods to improve the quality of the translations. We conducted human evaluations on generated po-

<p>French poem:</p> <p>Il ny avait que sable et boue Où s'était ouverte la tombe. Le long des murs de la prison On ne voyait aucune tombe. Un petit tas de chaux ardente Servait de linceul à cette ombre.</p> <p>Human reference:</p> <p>For where a grave had opened wide, There was no grave at all: Only a stretch of mud and sand By the hideous prison-wall, And a little heap of burning lime, That the man should have his pall.</p> <p>Unconstrained machine translation:</p> <p>There was only sand and mud Where the grave opened. Along the walls of prison We saw no grave A little pile of <UNK> <UNK> to this shadow.</p> <p>Rhythmic and rhyming translation by model B:</p> <p>But there was only sand and mud. To where the grave was laid. Along the walls of prison wall. We saw no masquerade. <i>A little lot of prostitutes.</i> They used to shroud this shade.</p> <p>Rhythmic and rhyming translation by model C:</p> <p>But there was only sand and mud. To where the grave was laid. Along the walls of prison wall. We saw no masquerade. <i>A little bunch of shiny lime.</i> They used to shroud this shade.</p>

Figure 2: An example of poetry translation by models B and C. Biased decoding with all potential translation (Model C) produces a better translation compared to Model B.

Method Name	User Preference
Model A	18.2%
Cannot Decide	19.5%
Model B	62.3%

Table 1: Users prefer translations generated by model A.

ems and showed that the proposed improvements highly improve the translation quality.

<p>French poem:</p> <p>Tels des vaisseaux dans la tempête, Nos deux chemins s'étaient croisés, Sans même un signe et sans un mot, Nous navions mot déclarer ; Nous n'étions pas dans la nuit sainte Mais dans le jour déshonoré.</p> <p>Human reference:</p> <p>Like two doomed ships that pass in storm We had crossed each others way: But we made no sign, we said no word, We had no word to say; For we did not meet in the holy night, But in the shameful day.</p> <p>Translation by our full system (model C):</p> <p>And like some ships across the storm. These paths were crossed astray. Without a signal nor a word. We had no word to say. We had not seen the holy night. But on the shameful day.</p>

Figure 3: A sample poem translated by our full system (Model C).

Method Name	User Preference
Model B	26.7%
Cannot Decide	28.9%
Model C	44.4%

Table 2: Users prefer translations generated by model C.

Very Bad	Bad	OK	Good	Very Good
5.9%	15.9%	28.6%	35.3%	14.3%

Table 3: Quality of the translated poems by model C.

Acknowledgments

We would like to thank the anonymous reviewers for their helpful comments. This work was supported in part by DARPA under the CwC program through the ARO (W911NF-15-1-0543), NSF (IIS-1524371), and gifts by Google and Facebook.

References

- Dmitriy Genzel, Jakob Uszkoreit, and Franz Och. 2010. Poetic statistical machine translation: rhyme and meter. In *Proceedings of EMNLP*.
- Marjan Ghazvininejad, Xing Shi, Yejin Choi, and Kevin Knight. 2016. Generating topical poetry. In *Proceedings of EMNLP*.

- Marjan Ghazvininejad, Xing Shi, Jay Priyadarshi, and Kevin Knight. 2017. Hafez: an interactive poetry generation system. In *Proceedings of ACL Demo Track*.
- Erica Greene, Tugba Bodrumlu, and Kevin Knight. 2010. Automatic analysis of rhythmic poetry with applications to generation and translation. In *Proceedings of EMNLP*.
- Jing He, Ming Zhou, and Long Jiang. 2012. Generating Chinese classical poems with statistical machine translation models. In *Proceedings of AAAI*.
- Sepp Hochreiter and Jurgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8).
- Jack Hopkins and Douwe Kiela. 2017. Automatically generating rhythmic verse with neural networks. In *Proceedings of ACL*.
- Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational linguistics* 29(1).
- Hugo Oliveira. 2012. PoeTryMe: a versatile platform for poetry generation. *Computational Creativity, Concept Invention, and General Intelligence* 1.
- Hugo Gonçalo Oliveira. 2017. A survey on intelligent poetry generation: Languages, features, techniques, reutilisation and evaluation. In *Proceedings of the 10th International Conference on Natural Language Generation*.
- Ilya Sutskever, Oriol Vinyals, and Quoc Le. 2014. Sequence to sequence learning with neural networks. In *proceedings of NIPS*.
- Qixin Wang, Tianyi Luo, Dong Wang, and Chao Xing. 2016. Chinese song iambics generation with neural attention-based model. In *Proceedings of IJCAI*.
- Oscar Wilde. 2001. *Ballad of Reading Gaol*. Electric Book Company.
- Rui Yan, Han Jiang, Mirella Lapata, Shou-De Lin, Xueqiang Lv, and Xiaoming Li. 2013. I, Poet: Automatic Chinese poetry composition through a generative summarization framework under constrained optimization. In *Proceedings of IJCAI*.
- Xiaoyuan Yi, Ruoyu Li, and Maosong Sun. 2017. Generating chinese classical poems with RNN encoder-decoder. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*.
- Xingxing Zhang and Mirella Lapata. 2014. Chinese poetry generation with recurrent neural networks. In *Proceedings of EMNLP*.