Translating Translationese: A Two-Step Approach to Unsupervised Machine Translation

Nima Pourdamghani[♣] **Nada Aldarrab**[♠] **Marjan Ghazvininejad**[♢]
Kevin Knight[♡] **Jonathan May**[♠]
[♣] Amazon [♠] USC Information Sciences Institute
[♢] Facebook AI Research [♡] DiDi Labs
nimpourd@amazon.com aldarrab@isi.edu ghazvini@fb.com
kevinknight@didiglobal.com jonmay@isi.edu

Abstract

Given a rough, word-by-word gloss of a source language sentence, target language natives can uncover the latent, fully-fluent rendering of the translation. In this work we explore this intuition by breaking translation into a two step process: generating a rough gloss by means of a dictionary and then 'translating' the resulting pseudo-translation, or 'Translationese' into a fully fluent translation. We build our Translationese decoder once from a mish-mash of parallel data that has the target language in common and then can build dictionaries on demand using unsupervised techniques, resulting in rapidly generated unsupervised neural MT systems for many source languages. We apply this process to 14 test languages, obtaining better or comparable translation results on high-resource languages than previously published unsupervised MT studies, and obtaining good quality results for low-resource languages that have never been used in an unsupervised MT scenario.

1 Introduction

Quality of machine translation, especially neural MT, highly depends on the amount of available parallel data. For a handful of languages, where parallel data is abundant, MT quality has reached quite good performance (Wu et al., 2016; Hassan et al., 2018). However, the quality of translation rapidly deteriorates as the amount of parallel data decreases (Koehn and Knowles, 2017). Unfortunately, many languages have close to zero parallel texts. Translating texts from these languages requires new techniques.

Hermjakob et al. (2018) presented a hybrid human/machine translation tool that uses lexical translation tables to gloss a translation and relies on human language and world models to propagate glosses into fluent translations. Inspired by that work, this work investigates the following question: Can we replace the human in the loop with more technology? We provide the following two-step solution to unsupervised neural machine translation:

1. Use a bilingual dictionary to gloss the input into a pseudo-translation or 'Translationese'.

2. Translate the Translationese into target language, using a model built in advance from various parallel data, with the source side converted into Translationese using Step 1.

The notion of separating adequacy from fluency components into a pipeline of operations dates back to the early days of MT and NLP research, where the inadequacy of word-by-word MT was first observed (Yngve, 1955; Oswald, 1952). A subfield of MT research that seeks to improve fluency given disfluent but adequate first-pass translation is *automatic post-editing* (APE) pioneered by Knight and Chander (1994). Much of the current APE work targets correction of black-box MT systems, which are presumed to be supervised.

Early approaches to unsupervised machine translation include decipherment methods (Nuhn et al., 2013; Ravi and Knight, 2011; Pourdamghani and Knight, 2017), which suffer from a huge hypothesis space. Recent approaches to zero-shot machine translation include pivot-based methods (Chen et al., 2017; Zheng et al., 2017; Cheng et al., 2016) and multi-lingual NMT methods (Firat et al., 2016a,b; Johnson et al., 2016; Ha et al., 2016, 2017). These systems are zero-shot for a specific source/target language pair, but need parallel data from source to a pivot or multiple other languages.

More recently, totally unsupervised NMT methods are introduced that use only monolingual data for training a machine translation system. Lample et al. (2018a,c), Artetxe et al. (2018), and

Yang et al. (2018) use iterative back-translation to train MT models in both directions simultaneously. Their training takes place on massive monolingual data and requires long time to train as well as careful tuning of hyperparameters.

The closest unsupervised NMT work to ours is by Kim et al. (2018). Similar to us, they break translation into glossing and correction steps. However, their correction step is trained on artificially generated noisy data aimed at simulating glossed source texts. Although this correction method helps, simulating noise caused by natural language phenomena is a hard task and needs to be tuned for every language.

Previous zero-shot NMT work compensates for a lack of source/target parallel data by either using source/pivot parallel data, extremely large monolingual data, or artificially generated data. These requirements and techniques limit the methods' applicability to real-world low-resource languages. Instead, in this paper we propose using parallel data from high-resource languages to learn 'how to translate' and apply the trained system to low resource settings. We use off-the-shelf technologies to build word embeddings from monolingual data (Bojanowski et al., 2017) and learn a source-to-target bilingual dictionary using source and target embeddings (Lample et al., 2018b). Given a target language, we train source-to-target dictionaries for a diverse set of high-resource source languages, and use them to convert the source side of the parallel data to Translationese. We combine this parallel data and train a Translationese-to-target translator on it. Later, we can build source-to-target dictionaries on-demand, generate Translationese from source texts, and use the pre-trained system to rapidly produce machine translation for many languages without requiring a single line of source-target parallel data.

We introduce the following contributions in this paper:

- Following Hermjakob et al. (2018), we propose a two step pipeline for building a rapid neural MT system for many languages. The pipeline does not require parallel data or parameter fine-tuning when adapting to new source languages.
- The pipeline only requires a comprehensive source to target dictionary. We show that this dictionary can be easily obtained using off-the shelf tools within a few hours.
- We use this system to translate test texts from 14 languages into English. We obtain better or comparable quality translation results on high-resource languages than previously published unsupervised MT studies, and obtain good quality results for low-resource languages that have never been used in an unsupervised MT scenario. To our knowledge, this is the first unsupervised NMT work that shows good translation results on such a large number of languages.

2 Method

We introduce a two-step pipeline for unsupervised machine translation. In the first step a source text is glossed into a pseudo-translation or Translationese, while in the second step a pre-trained model translates the Translationese into target. We introduce a fully unsupervised method for converting the source into Translationese, and we show how to train a Translationese to target system in advance and apply it to new source languages.

2.1 Building a Dictionary

The first step of our proposed pipeline includes a word-by-word translation of the source texts. This requires a source/target dictionary. Manually constructed dictionaries exist for many language pairs, however cleaning these dictionaries to get a word to word lexicon is not trivial, and these dictionaries often cover a small portion of the source vocabulary, focusing on stems and specifically excluding inflected variants. In order to have a comprehensive, word to word, inflected bi-lingual dictionary we look for automatically built ones.

Automatic lexical induction is an active field of research (Fung, 1995; Koehn and Knight, 2002; Haghighi et al., 2008; Lample et al., 2018b). A popular method for automatic extraction of bilingual dictionaries is through building cross-lingual word embeddings. Finding a shared word representation space between two languages enables us to calculate the distance between word embeddings of source and target, which helps us to find translation candidates for each word.

We follow this approach for building the bilingual dictionaries. For a given source and target language, we start by separately training source and target word embeddings S and T, and use the method introduced in (Lample et al., 2018b) to find a linear mapping W that maps the source

embedding space to the target: $SW = T$.

Lample et al. (2018b) propose an adversarial method for estimating W, where a discriminator is trained to distinguish between elements randomly sampled from WS and T, and W is trained to prevent the discriminator from making accurate classifications. Once the initial mapping matrix W is trained, a number of refinement steps is performed to improve performance over less frequent words by changing the metric of the space.

We use the trained matrix W to map the source embeddings into the space of the target embeddings. Then we find the k-nearest neighbors among the target words for each source word, according to the cosine distance metric. These nearest neighbors represent our translation options for that source word.

2.2 Source to Translationese

Once we have the translation options for tokens in the source vocabulary we can perform a word by word translation of the source into Translationese. However, a naive translation of each source token to its top translation option without considering the context is not the best way to go. Given different contexts, a word should be translated differently.

We use a 5gram target language model to look at different translation options for a source word and select one based on its context. This language model is trained in advance on large target monolingual data.

In order to translate a source sentence into Translationese we apply a beam search with a stack size of 100 and assign a score equal to $\alpha P_{LM} + \beta d(s,t)$ to each translation option t for a source token s, where P_{LM} is the language model score, and $d(s,t)$ is the cosine distance between source and target words. We set $\alpha = 0.01$ and $\beta = 0.5$

2.3 Translationese to Target

We train a transformer model (Vaswani et al., 2017) on parallel data from a diverse set of high-resource languages to translate Translationese into a fluent target. For each language we convert the source side of the parallel data to Translationese as described in Section 2.2. Then we combine and shuffle all the Translationese/target parallel data and train the model on the result. Once the model is trained, we can apply it to the Translationese coming from any source language.

We use the tensor2tensor implementation[1] of the transformer model with the `transformer_base` set of hyperparameters (6 layers, hidden layer size of 512) as our translation model.

3 Data and Parameters

For all our training and test languages, we use the pre-trained word embeddings[2] trained on Wikipedia data using fastText (Bojanowski et al., 2017). These embeddings are used to train bilingual dictionaries.

We select English as the target language. In order to avoid biasing the trained system toward a language or a specific type of parallel data, we use diverse parallel data on a diverse set of languages to train the Translationese to English system. We use Arabic, Czech, Dutch, Finnish, French, German, Italian, Russian, and Spanish as the set of out training languages.

We use roughly 2 million sentence pairs per language and limit the length of the sentences to 100 tokens. For Dutch, Finnish, and Italian we use Europarl for parallel data. For Arabic we use MultiUN. For French we use CommonCrawl. For German we use a mix of CommonCrawl (1.7M), and NewsCommentary (300K). The numbers in parenthesis show the number of sentences for each dataset. For Spanish we use CommonCrawl (1.8M), and Europarl (200K). For Russian we use Yandex (1M), CommonCrawl (800K), and NewsCommentary (200K), and finally for Czech we use a mix of ParaCrawl (1M), Europarl (640K), NewsCommentary (200K), and CommonCrawl (160K).

We train one model on these nine languages and apply it to test languages not in this set. Also, to test on each of the training languages we train a model where the parallel data for that language is excluded from the training data. In each experiment we use 3000 blind sentences randomly selected out of the combined parallel data as the development set.

We use the default parameters in (Lample et al., 2018b) to find the cross-lingual embedding vectors. In order to create the dictionary we limit the size of the source and target (English) vocabulary to 100K tokens. For each source token we find 20

[1] https://github.com/tensorflow/tensor2tensor

[2] https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

nearest neighbors in the target language. We use a 5gram language model trained on 4 billion tokens of Gigaword to select between the translation options for each token. We use Moses scripts for tokenizing and lowercasing the data. We do not apply BPE (Sennrich et al., 2015) on the data. In order to be comparable to Kim et al. (2018) we split German compound words only for the newstest2016 test data. We use the CharSplit[3] python package for this purpose. We use tensor2tensor's `transformer_base` hyperparameters to train the transformer model on a single gpu for each language.

4 Experiments

We report translation results on newstest2013 for Spanish, newstest2014 for French, and newstest2016 for Czech, German, Finnish, Romanian, and Russian. We also report results on the first 3000 sentences of GlobalVoices2015[4] for Dutch, Bulgarian, Danish, Indonesian, Polish, Portuguese, and Catalan. In each experiment we report the quality of the intermediate Translationese as well as the scores for our full model.

	fr-en	de-en	ru-en	ro-en
(Lample et al., 2018a)	14.3	13.3	-	-
(Artetxe et al., 2018)	15.6	10.2	-	-
(Yang et al., 2018)	15.6	14.6	-	-
(Lample et al., 2018c) (transformer)	24.2	21.0	9.1	19.4
(Kim et al., 2018)	16.5	17.2	-	-
Translationese	11.6	13.8	5.7	8.1
Full Model	21.0	18.7	12.0	16.3

Table 1: Comparing translation results on newstest2014 for French, and newstest2016 for Russian, German, and Romanian with previous unsupervised NMT methods. (Kim et al., 2018) is the method closest to our work. We report the quality of Translationese as well as the scores for our full model.

We compare our results against all the existing fully unsupervised neural machine translation methods in Table 1 and show better results on common test languages compared to all of them except Lample et al. (2018c) where, compared to their transformer model[5], we improve results for Russian, but not for other languages.

The first four methods that we compare against are based on back-translation. These methods require huge monolingual data and large training time to train a model per test language. The fifth method, which is most similar to our approach (Kim et al., 2018), can be trained quickly, but still is fine tuned for each test language and performs worse than our method. Unlike the previous works, our model can be trained once and applied to any test language on demand. Besides this, these methods use language-specific tricks and development data for training their models while our system is trained totally independent of the test language.

We also show acceptable BLEU scores for ten other languages for which no previous unsupervised NMT scores exist, underscoring our ability to produce new systems rapidly (Table 2).

	cs-en	es-en	fi-en	nl-en	bg-en
Translationese	7.4	12.7	3.8	16.9	10.0
Full Model	13.7	22.2	7.2	22.0	16.8
	da-en	id-en	pl-en	pt-en	ca-en
Translationese	13.6	7.4	8.3	15.2	10.1
Full Model	18.5	13.7	14.8	23.1	19.8

Table 2: Translation results on ten new languages: Czech, Spanish, Finnish, Dutch, Bulgarian, Danish, Indonesian, Polish, Portuguese, and Catalan

5 Conclusion

We propose a two step pipeline for building a rapid unsupervised neural machine translation system for any language. The pipeline does not require retraining the neural translation model when adapting to new source languages, which makes its application to new languages extremely fast and easy. The pipeline only requires a comprehensive source-to-target dictionary. We show how to easily obtain such a dictionary using off-the shelf tools. We use this system to translate test texts from 14 languages into English. We obtain better or comparable quality translation results on high-resource languages than previously published unsupervised MT studies, and obtain good quality results for ten

[3] https://github.com/dtuggener/CharSplit

[4] http://opus.nlpl.eu/GlobalVoices.php

[5] They present better results when combining their transformer model with an unsupervised phrase-based translation model.

other languages that have never been used in an unsupervised MT scenario.

Acknowledgements

The research is based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via AFRL Contract FA8650-17-C-9116 and by the Defense Advanced Research Projects Agency (DARPA) via contract HR0011-15-C-0115. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, DARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

References

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. In *Proc. ICLR*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Yun Chen, Yang Liu, Yong Cheng, and Victor OK Li. 2017. A teacher-student framework for zero-resource neural machine translation. *arXiv preprint arXiv:1705.00753*.

Yong Cheng, Yang Liu, Qian Yang, Maosong Sun, and Wei Xu. 2016. Neural machine translation with pivot languages. *arXiv preprint arXiv:1611.04928*.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016a. Multi-way, multilingual neural machine translation with a shared attention mechanism. *arXiv preprint arXiv:1601.01073*.

Orhan Firat, Baskaran Sankaran, Yaser Al-Onaizan, Fatos T Yarman Vural, and Kyunghyun Cho. 2016b. Zero-resource translation with multilingual neural machine translation. *arXiv preprint arXiv:1606.04164*.

Pascale Fung. 1995. Compiling bilingual lexicon entries from a non-parallel English-Chinese corpus. In *Workshop on Very Large Corpora*.

Thanh-Le Ha, Jan Niehues, and Alexander Waibel. 2016. Toward multilingual neural machine translation with universal encoder and decoder. *arXiv preprint arXiv:1611.04798*.

Thanh-Le Ha, Jan Niehues, and Alexander Waibel. 2017. Effective strategies in zero-shot neural machine translation. *arXiv preprint arXiv:1711.07893*.

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. 2008. Learning bilingual lexicons from monolingual corpora. In *Proc. ACL*.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, et al. 2018. Achieving human parity on automatic Chinese to English news translation. *arXiv preprint arXiv:1803.05567*.

Ulf Hermjakob, Jonathan May, Michael Pust, and Kevin Knight. 2018. Translating a language you don't know in the Chinese room. In *Proc. ACL, System Demonstrations*.

Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. 2016. Google's multilingual neural machine translation system: enabling zero-shot translation. *arXiv preprint arXiv:1611.04558*.

Yunsu Kim, Jiahui Geng, and Hermann Ney. 2018. Improving unsupervised word-by-word translation with language model and denoising autoencoder. In *Proc. EMNLP*.

Kevin Knight and Ishwar Chander. 1994. Automated postediting of documents. In *Proc AAAI*.

Philipp Koehn and Kevin Knight. 2002. Learning a translation lexicon from monolingual corpora. In *Proc. ACL workshop on Unsupervised lexical acquisition*.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. *arXiv preprint arXiv:1706.03872*.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018a. Unsupervised machine translation using monolingual corpora only. In *Proc. ICLR*.

Guillaume Lample, Alexis Conneau, MarcÁurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018b. Word translation without parallel data. In *Proc. ICLR*.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018c. Phrase-based & neural unsupervised machine translation. In *Proc. EMNLP*.

Malte Nuhn, Julian Schamper, and Hermann Ney. 2013. Beam search for solving substitution ciphers. In *Proc. ACL*.

Victor Oswald. 1952. Word-by-word translation. In *Proc. intervention à la Conférence du MIT*.

Nima Pourdamghani and Kevin Knight. 2017. Deciphering related languages. In *Proc. EMNLP*.

Sujith Ravi and Kevin Knight. 2011. Deciphering foreign language. In *Proc. ACL*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. NIPS*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2018. Unsupervised neural machine translation with weight sharing. In *Proc. ACL*.

Victor H. Yngve. 1955. Sentence-for-sentence translation. *Mechanical Translation*, 2(2):29–37.

Hao Zheng, Yong Cheng, and Yang Liu. 2017. Maximum expected likelihood estimation for zero-resource neural machine translation. In *Proc. IJCAI*.