NICT's Machine Translation Systems for the WMT19 Similar Language Translation Task

Benjamin Marie **Raj Dabre** **Atsushi Fujita**
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0289, Japan
{bmarie, raj.dabre, atsushi.fujita}@nict.go.jp

Abstract

This paper presents the NICT's participation in the WMT19 shared Similar Language Translation Task. We participated in the Spanish–Portuguese task. For both translation directions, we prepared state-of-the-art statistical (SMT) and neural (NMT) machine translation systems. Our NMT systems with the Transformer architecture were trained on the provided parallel data enlarged with a large quantity of back-translated monolingual data. Our primary submission to the task is the result of a simple combination of our SMT and NMT systems. According to BLEU, our systems were ranked second and third respectively for the Portuguese-to-Spanish and Spanish-to-Portuguese translation directions. For contrastive experiments, we also submitted outputs generated with an unsupervised SMT system.

1 Introduction

This paper describes the machine translation (MT) systems built for the participation of the National Institute of Information and Communications Technology (NICT) in the WMT19 shared Similar Language Translation Task. We participated in Spanish–Portuguese (es-pt) in both translation directions. We chose this language pairs to explore the potential of unsupervised MT for very close languages with large monolingual data, and to compare it with supervised MT systems trained on large bilingual data.

We participated under the team name "NICT." All our systems were *constrained*, i.e., we used only the parallel and monolingual data provided by the organizers to train and tune the MT systems. For both translation directions, we trained supervised neural MT (NMT) and statistical MT (SMT) systems, and combined them through n-best list reranking using different informative features as proposed by Marie and Fujita (2018a). This simple combination method, in conjunction with the exploitation of large back-translated monolingual data (Sennrich et al., 2016a), performed among the best MT systems in this task.

The remainder of this paper is organized as follows. Section 2 introduces the data preprocessing. Section 3 describes the details of our NMT and SMT systems, and also our unsupervised SMT systems. Then, the combination of NMT and SMT is described in Section 4. Empirical results produced with our systems are presented in Section 5, and Section 6 concludes this paper.

2 Data Preprocessing

2.1 Data

As parallel data to train our systems, we used all the provided data. As monolingual data, we used the provided "News Crawl" corpora that are sufficiently large and in-domain to train our unsupervised systems and be used for generating useful pseudo-parallel data through back-translation. To tune/validate our systems, we used the provided development data.

2.2 Tokenization, Truecasing, and Cleaning

We used the tokenizer and truecaser of `Moses` (Koehn et al., 2007). The truecaser was trained on one million tokenized lines extracted randomly from the monolingual data. Truecasing was then performed on all the tokenized data. For cleaning, we only applied the `Moses` script `clean-corpus-n.perl` to remove lines in the parallel data containing more than 80 tokens and replaced characters forbidden by `Moses`. Note that we did not perform any punctuation normalization. Table 1 presents the statistics of the parallel and monolingual data, respectively, after preprocessing.

Proceedings of the Fourth Conference on Machine Translation (WMT), Volume 3: Shared Task Papers (Day 2) pages 208–212
Florence, Italy, August 1-2, 2019.

Corpus	#sent. pairs		#sent. tokens	
	es	pt	es	pt
Parallel	3.41M	3.41M	87.38M	84.69M
Development	3,000	3,000	69,704	68,284
Monolingual	40.88M	7.61M	1.22B	171.15M

Table 1: Statistics of our preprocessed data.

3 MT Systems

3.1 NMT

For our NMT systems, we adopt the Transformer architecture (Vaswani et al., 2017). We chose `Marian` (Junczys-Dowmunt et al., 2018)[1] to train our NMT systems since it supports state-of-the-art features and is one of the fastest NMT frameworks publicly available. In order to limit the size of the vocabulary of the NMT models, we segmented tokens in the parallel data into sub-word units via byte pair encoding (BPE) (Sennrich et al., 2016b) using 30k operations. BPE segmentations were jointly learned on the training parallel data for the source and target languages. All our NMT systems were consistently trained on 4 GPUs,[2] with the parameters for `Marian` listed in Table 2. To improve translation quality, we added 5M synthetic sentence pairs, obtained through back-translating (Sennrich et al., 2016a) the first 5M sentences from the monolingual corpora, to the original parallel data for training. We performed NMT decoding with an ensemble of a total of four models according to the best BLEU (Papineni et al., 2002) scores on the development data produced by four independent training runs using the same training parameters.

3.2 SMT

We trained SMT systems using `Moses`.[3] Word alignments and phrase tables were obtained from the tokenized parallel data using `mgiza`. Source-to-target and target-to-source word alignments were symmetrized with the `grow-diag-final-and` heuristic. We also trained `MSLR` (monotone, swap, discontinuous-left, discontinuous-right) lexicalized reordering model. We trained one 4-gram language models on the entire monolingual data concatenated to the target side of the parallel data using `LMPLZ`

[1] https://marian-nmt.github.io/, version 1.6.0

[2] NVIDIA® Tesla® P100 16Gb.

[3] https://github.com/moses-smt/mosesdecoder/

```
--type transformer
--max-length 80
--mini-batch-fit --valid-freq
5000 --save-freq 5000
--workspace 8000 --disp-freq
500 --beam-size 12 --normalize
1 --valid-mini-batch 16
--overwrite --early-stopping
5 --cost-type ce-mean-words
--valid-metrics ce-mean-words
translation --keep-best
--enc-depth 6 --dec-depth
6 --transformer-dropout
0.1 --learn-rate 0.0003
--dropout-src 0.1
--dropout-trg 0.1 --lr-warmup
16000 --lr-decay-inv-sqrt
16000 --lr-report
--label-smoothing 0.1
--devices 0 1 2 3 --dim-vocabs
30000 30000 --optimizer-params
0.9 0.98 1e-09 --clip-norm 5
--sync-sgd --tied-embeddings
--exponential-smoothing
```

Table 2: Parameters of Marian used for training our NMT systems.

(Heafield et al., 2013). Our systems used the default distortion limit of 6. We tuned the SMT model weights with `KB-MIRA` (Cherry and Foster, 2012) and selected the weights giving the best BLEU score on the development data after 15 iterations.

3.3 Unsupervised SMT

We also built an SMT system, without any supervision, i.e., using only but all the provided monolingual data for training. We chose unsupervised SMT (USMT) over unsupervised NMT (UNMT) since previous work (Artetxe et al., 2018b) has shown that USMT slightly outperforms UNMT and that we expect USMT to work well for this language pair that involves only very few word reorderings.

We built USMT systems using a framework similar to the one proposed in Marie and Fujita (2018b). The first step of USMT is the induction of a phrase table from the monolingual corpora. We first collected phrases of up to six tokens from the monolingual News Crawl corpora

using word2phrase.[4] As phrases, we also considered all the token types in the corpora. Then, we selected the 300k most frequent phrases in the monolingual corpora to be used for inducing a phrase table. All possible phrase pairs are scored, as in Marie and Fujita (2018b), using bilingual word embeddings, and the 300 target phrases with the highest scores were kept in the phrase table for each source phrase. In total, the induced phrase table contains 90M (300k×300) phrase pairs. For this induction, bilingual word embeddings of 512 dimensions were obtained using word embeddings trained with fastText[5] and aligned in the same space using unsupervised Vecmap (Artetxe et al., 2018a). For each one of these phrase pairs a total of four scores, to be used as features in the phrase table were computed to mimic phrase-based SMT: forward and backward phrase and lexical translation probabilities. Finally, the phrase table was plugged into a Moses system that was tuned on the development data using KB-MIRA. We performed four refinement steps to improve the system using at each step 3M synthetic parallel sentences generated, from sentences randomly sampled from the monolingual data, by the forward and backward translation systems, instead of using only either forward (Marie and Fujita, 2018b) or backward translations (Artetxe et al., 2018b). We report on the performance of the systems obtained after the fourth refinement step.

4 Combination of NMT and SMT

Our primary submission for WMT19 is the result of a simple combination of NMT and SMT. Indeed, as demonstrated by Marie and Fujita (2018a), and despite the simplicity of the method used, combining NMT and SMT makes MT more robust and can significantly improve translation quality, even when SMT greatly underperforms NMT. Moreover, due to the very few word reorderings to perform and the morphological similarity between Spanish and Portuguese, we can expect SMT to perform closely to NMT while remaining different and complementary. Following Marie and Fujita (2018a), our combination of NMT and SMT works as follows.

4.1 Generation of n-best Lists

We first produced the six 100-best lists of translation hypotheses generated by four NMT left-to-right models individually, by their ensemble, and by one right-to-left model. Unlike Moses, Marian must use a beam of size k to produce a k-best list during decoding. However, using a larger beam size during decoding for NMT may worsen translation quality (Koehn and Knowles, 2017). Consequently, we also produced with Marian the 12-best lists and merged them with Marian's 100-best lists to obtain lists containing up to 112 hypotheses,[6] or up to 672 hypotheses after merging all the lists produced by NMT. In this way, we make sure that we still have hypotheses of good quality in the lists despite using a larger beam size. We also generated 100-best translation hypotheses with SMT.[7] Finally, we merged the lists produced by Marian and Moses.

4.2 Reranking Framework and Features

We rescored all the hypotheses in the resulting lists with a reranking framework using SMT and NMT features to better model the fluency and the adequacy of each hypothesis. This method can find a better hypothesis in these merged n-best lists than the one-best hypothesis originated by either Moses or Marian. We chose KB-MIRA as a rescoring framework and used a subset of the features proposed in Marie and Fujita (2018a). As listed in Table 3, it includes the scores given by the four left-to-right NMT models used to perform ensemble decoding (see Section 3.1). We also used as features the scores given by the right-to-left NMT model that we trained for each translation direction with the same parameters as left-to-right NMT models. The right-to-left NMT model achieving the best BLEU score on the development data, was selected, giving us another feature for each translation direction. All the following features we used are described in details by Marie and Fujita (2018a). We computed sentence-level translation probabilities using the lexical translation probabilities learned by mgiza during the training of our SMT systems. The language model trained for SMT was also used to score the transla-

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

[5] https://github.com/facebookresearch/fastText

[6] Note that we did not remove duplicated hypotheses that may appear, for instance, in both 12-best and 100-best lists.

[7] We used the option distinct in Moses to avoid duplicated hypotheses, i.e., with the same content but obtained from different word alignments, and consequently to increase diversity in the generated n-best lists.

Feature	Description
L2R (4)	Scores given by each of the 4 left-to-right `Marian` models
R2L (1)	Scores given by 1 right-to-left `Marian` models
LEX (4)	Sentence-level translation probabilities, for both translation directions
LM (1)	Scores given by the language model used by our SMT system
LEN (2)	Difference between the length of the source sentence and the length of the translation hypothesis, and its absolute value

Table 3: Set of features used by our reranking systems. The column "Feature" refers to the same feature name used in Marie and Fujita (2018a). The numbers in parentheses indicate the number of scores in each feature set.

System	es→pt		pt→es	
	dev	test	dev	test
SMT	55.6	-	60.4	-
NMT	53.8	-	61.3	-
Reranked SMT+NMT	57.2	53.3	61.9	59.9
USMT	51.4	47.9	57.9	54.9

Table 4: Results (BLEU). Since the translation reference of the test data was not released at the time of writing this paper, we could not compute BLEU scores on the test data for the configurations that we did not submit to the tasks and put "-" instead.

tion hypotheses. To account for hypotheses length, we added the difference, and its absolute value, between the number of tokens in the translation hypothesis and the source sentence.

The reranker was trained on n-best lists produced by decoding the same development data that we used to validate NMT system's training and to tune SMT's model weights.

5 Results

The results for both translation directions are presented in Table 4. As expected, we obtained very high BLEU scores that point out that the proximity between the two languages has a key role in the success of MT. Also, due to the many characteristics shared by both languages, especially regarding word orderings and morphology, we can observe that SMT performed as good as NMT. Combining SMT and NMT through reranking derived our best results with, for instance, a substantial improvement of 1.6 BLEU points for es→pt on the development data.

USMT also achieved very high BLEU scores: only 5.4 BLEU points below our primary model for es→pt on the test data. The USMT performance points out that training MT systems with large bilingual data may be unnecessary for very close languages, such as Spanish and Portuguese.

6 Conclusion

We participated in the Spanish–Portuguese translation task and compared a strong supervised MT system with USMT. While our supervised MT system significantly outperformed USMT, we showed that USMT for close languages has the potential to be a reasonable alternative since it can deliver a good translation quality without requiring manual creation of large parallel data for training.

Acknowledgments

We would like to thank the reviewers for their useful comments and suggestions. This work was conducted under the program "Research and Development of Enhanced Multilingual and Multipurpose Speech Translation Systems" of the Ministry of Internal Affairs and Communications (MIC), Japan.

References

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018b. Unsupervised statistical machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3632–3642, Brussels, Belgium. Association for Computational Linguistics.

Colin Cherry and George Foster. 2012. Batch tuning strategies for statistical machine translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 427–436, Montréal, Canada. Association for Computational Linguistics.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 690–696, Sofia, Bulgaria. Association for Computational Linguistics.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180. Association for Computational Linguistics.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Benjamin Marie and Atsushi Fujita. 2018a. A smorgasbord of features to combine phrase-based and neural machine translation. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, pages 111–124. Association for Machine Translation in the Americas.

Benjamin Marie and Atsushi Fujita. 2018b. Unsupervised neural machine translation initialized by unsupervised statistical machine translation. *CoRR*, abs/1810.12703.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

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 *Proceedings of the 30th Neural Information Processing Systems Conference*, pages 5998–6008.