NICT's Machine Translation Systems for CCMT-2019 Translation Task

Kehai Chen Rui Wang Masao Utiyama Eiichiro Sumita
{khchen,wangrui,mutiyama,eiichiro.sumita}@nict.go.jp
National Institute of Information and Communications Technology (NICT), Kyoto, Japan

1. Summary

Abstract

- This paper describes the NICT's neural machine translation systems for Chinese↔English directions in the CCMT-2019 shared news translation task.
- Our system makes use of techniques that have been proven to be most effective to improve the performance of NMT model, and thereby generates the primary submissions of Chinese↔English translation tasks.

Key points

- Using Marian Toolkit (v1.7.6) to build baseline translation model for Chinese↔English translation tasks.
- **Back-translation** technique is used to performe data augmentation for providing a large quantity of pseudo parallel data.
- **Fine-tuning** technique to further optimizate the trained NMT model on small development data.
- **Ensemble** technique focuses on making full use of diverse NMT models to generate better translations.

2. Datasets and Preprocessing

Datasets: pre-processed parallel data (left table) and pre-processed monolingual data (right table)

Language pair	#Sentence pairs	#Tokens	
		Chinese	English
Chinese↔English	24.8M	509.9M	576.2M

Language	#Sentences	#Tokens
English	338.7M	7.5B
Chinese	130.5M	2.3B

Pre-processing steps

- For Chinese and English monolingual data:
  - Empirical selecting the first ten million lines of the News Crawl 2019 English corpus
  - English: tokenizer and truecaser in Moses
  - Chinese: Jieba
  - Filtering out sentences longer than 80 tokens in the training data
  - Replacing characters forbidden by Moses

3. Experiments

Marian Toolkit (v1.7.6) for Chinese↔English

- 50,000 sub-word vocabularies
- Batch sizes of 4096 words
- The number of dimensions of input and output layers was set to 512
- The inner feed-forward neural network layer was set to 2048
- The number of attention heads was set to eight
- Warm-up steps of 16,000
- The label smoothing and attention dropout were set to 0.1
- The Adam optimizer was used to tune the parameters of the model
- Both encoder and decoder are stack of 6 SAN layers
- All models trained on four P100 GPUs
- Train till convergence on development data
- Early stopping and fine-tuning on development data
- Ensemble 5 independent runs for decoding
- Beam size and length penalty tuned on dev set

Main results (BLEU-cased)

#	System	ZH→EN	EN→ZH
1.	Single Transformer (base) model (w/o back-translation)	23.3	30.3
2.	Single Transformer (base) model (w/ back-translation)	25.3	31.8
3.	Single Transformer (base) model (w/ fine-tuning)	27.5	33.1
4.	ensembling five single models (w/ back-translation and fine-tuning)	31.0	34.5

Observation

- Back-translation obtains improvements by 2.0 BlEU scores (#2) over single Transformer (base) model (w/o back-translation) (#1)
- Large improvements with Single Transformer (base) model (w/ fine-tuning)
- Best results with ensemble five single models (w/ back-translation and fine-tuning)
- This results conformed that these three methods can incrementally improve translation performance.