Exploiting General Purpose Sequence Representations for Low Resource Neural Machine Translation

H.Jeung Han*, Sathish Indurthi*, Sangha Kim
Samsung Research

*Equal contribution

Introduction

- **Neural Machine Translation**
  - Neural Machine Translation system has achieved near human level performance with abundant parallel corpus. (but… with small dataset…?)
- **Previous work on Low Resource NMT**
  - NMT with monolingual data, multilingual NMT, etc.
  - Transfer learning and meta-learning with auxiliary high resource parallel corpus and fine-tuned on low resource corpus.
    1. They assume the availability of several high-resource languages pairs for pre-training
    2. Most of them also assume English as the target language in all the language pairs -> difficult to apply non-English language translation.
- **General purpose sequence representations**
  - Led strong improvements in numerous NLP tasks.
  - Several attempts to utilize such representations into NMT system.

Introduction

- **NMTwGSR** (Neural Machine Translation with General Purpose Sequence Representations system)
  - In the proposed model, not explicitly training the encoder for source language using general purpose sequence representations.
  - The trainable parameters are only from the decoder and the output layer.
  - Advantages
    1. It doesn't assume the availability of several high resource language pairs for pre-training as required by the previous approaches.
    2. The previously proposed techniques such as back-translation, multilingual-NMT, transfer/meta-learning training strategies are straight forward to integrated into to the proposed system when high resource is available for the corresponding target pair.
    3. It can be easily adopted to a new language pair coming from low resource languages, whenever the sequence representations are available for these in language models.
  - Extensive evaluation on four low-resource translation task.

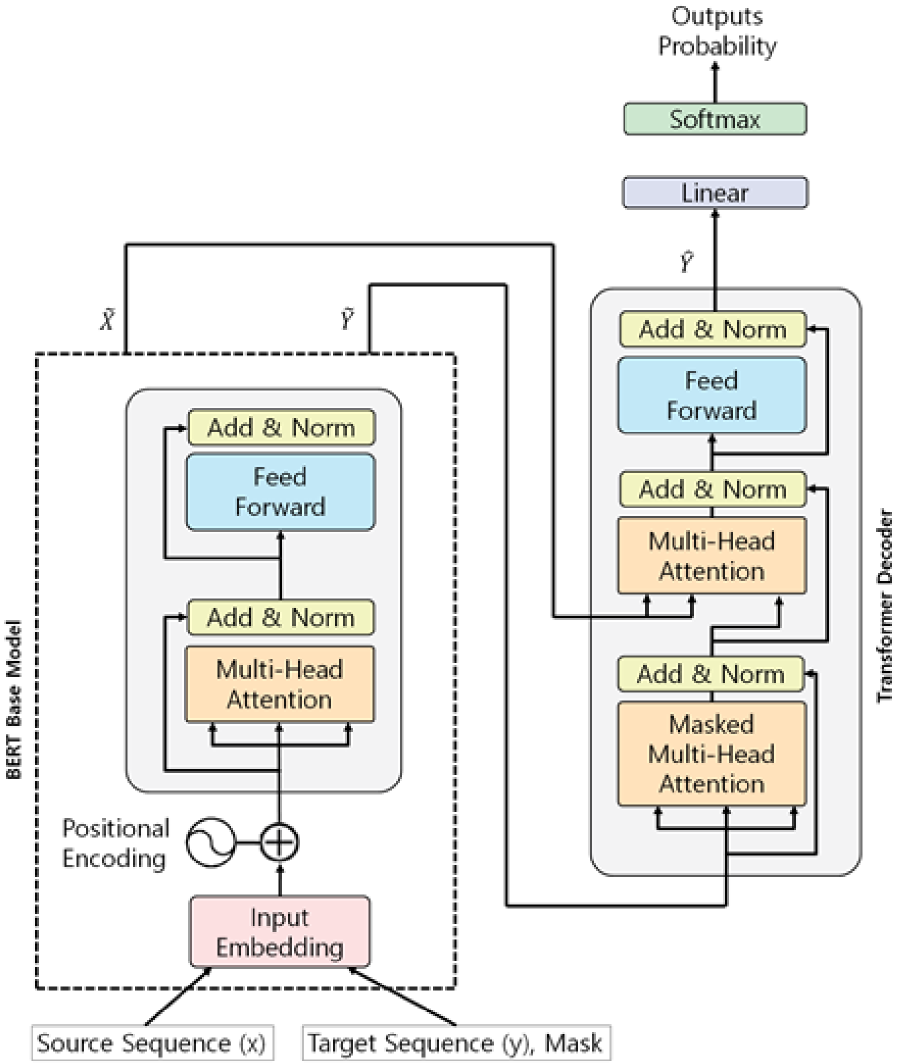
NMTwGSR

- **Model**
  - $\widetilde{\mathbf{X}} = BERT(\mathbf{x}) \in R^{m\times d}$.
  - $\widetilde{\mathbf{Y}} = BERT_{Masked}(\mathbf{y}, mask) \in R^{n\times d}$.
  - $\widehat{\mathbf{Y}} = Transformer_{decoder}(\widetilde{\mathbf{X}}, \widetilde{\mathbf{Y}})$.
  - $\mathrm{p}(w_t) = softmax(W_o \hat{y}_t + b_o)$

$W_o \in R^{V\times d}$ and b_o are learnable parameters.

V : output vocabulary size

$\hat{y}_t \in \widehat{\mathbf{Y}}$: output from decoder at time t.



Experiments Settings

- **Dataset and Implementation details**
  - 4 language pair (Romanian, Finnish, Turkish, Latvian -> English) from WMT
  - Low resource environment simulation by 5 times random sampling of 160k, 320k, 640k, 1280k tokens
  - BERT-Base, Multilingual Cased model, weight fixed
  - tensorflow, OpenNMT

Dataset	# Tokens (En)	# Sentences			
		Lv-En	Fi-En	Ro-En	Tr-En
Train	Full	4.46M	2.63M	0.61M	0.21M
	1280K	99.1K	58.8K	55.4K	60K
	640K	49.8K	29.4K	27.7K	30K
	320K	24.7K	14.7K	13.9K	15K
	160K	12.4K	7.3K	7K	7.5K
Dev	-	2K	3K	2K	3K
Test	-	2K	3K	2K	3K

Experimental Results

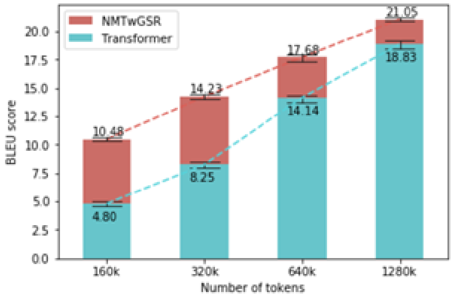
- **Training in Low Resource Environment**
  - BLEU score of model trained on sampled subset on 4 language in Table- Forward and Reverse direction

#Tok	Ro-En		En-Ro		Fi-En		En-Fi		Tr-En		En-Tr		Lv-En		En-Lv	
	Base	Ours	Base	Ours	Base	Ours	Base	Ours	Base	Ours	Base	Ours	Base	Ours	Base	Ours
Full	31.76	32.68	-	-	20.20	22.39	-	-	13.74	15.19	-	-	15.15	17.39	-	-
1280K	18.83	21.05	16.96	19.67	7.70	9.26	7.20	8.02	9.40	10.24	9.87	10.12	6.17	7.19	5.32	6.39
640K	14.14	17.68	12.70	16.38	5.53	7.62	3.67	4.58	5.08	7.76	4.58	6.43	3.80	5.22	3.01	4.30
320K	8.25	14.23	6.63	11.01	3.24	5.47	1.98	2.92	2.59	5.44	1.74	3.91	2.29	3.52	1.71	2.76
160K	4.80	10.48	4.20	7.97	1.04	3.12	0.53	1.91	1.44	3.49	1.02	2.15	0.85	2.15	0.96	1.72

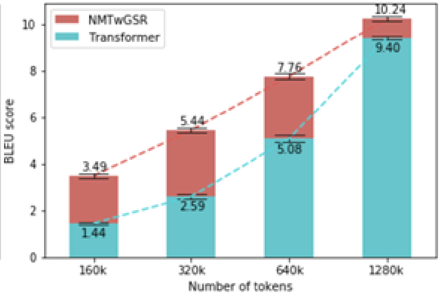
Table 1: Test BLEU results of models trained on 160k, 320k, 640k, and 1280k sampled subsets of four langauge paris both for forward and backward. The models trained on Full dataset is only presented with forward direction.

Experimental Results

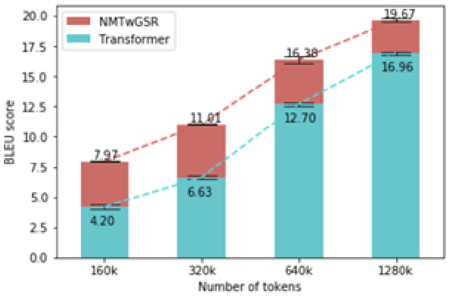
- **Training in Low Resource Environment**
  - Forward direction
  - Reverse direction



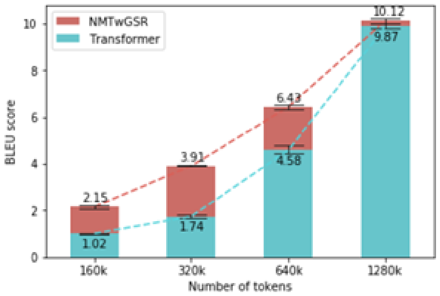
(a) Ro-En



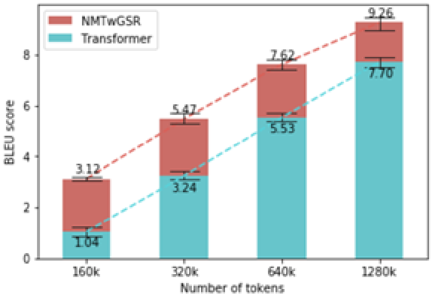
(a) Tr-En



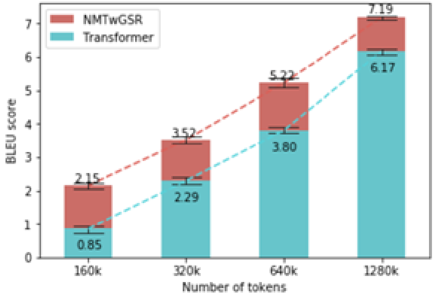
(b) En-Ro



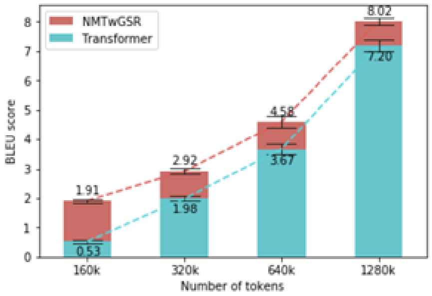
(b) En-Tr



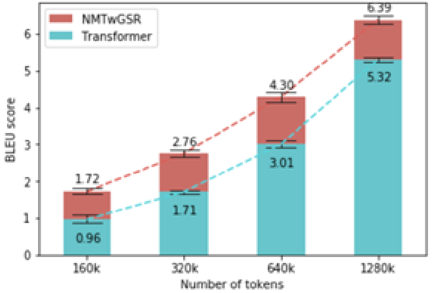
(c) Fi-En



(c) Lv-En



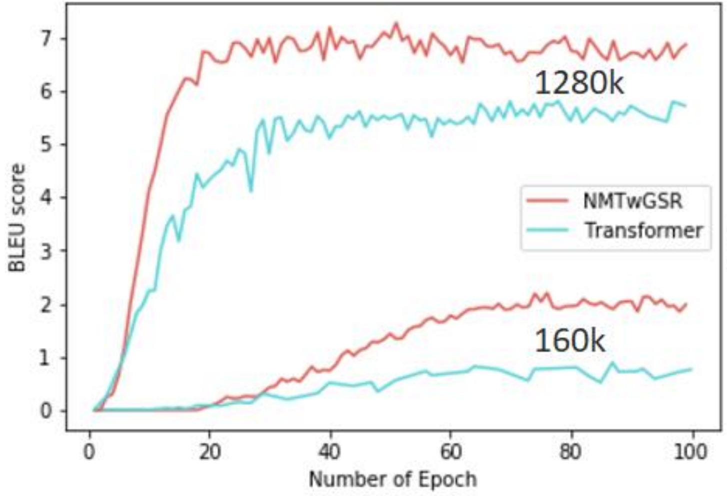
(d) En-Fi



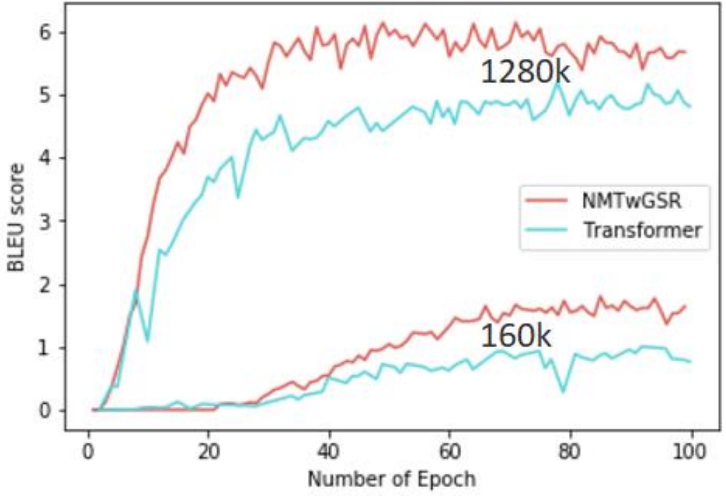
(d) En-Lv

Experimental Results

- **Learning curves**
  - The learning curves of BLEU scores for Lv-En and En-Lv pairs
  - In each plot, the upper learning curves are obtained using 1280k subset and lower learning curves are obtained using 160k subset.



(a) Lv-En



(b) En-Lv

Thank you