Paper Review

Title: PRE-TRAINED LANGUAGE MODEL REPRESENTATIONS FOR LANGUAGE GENERATION

by
Anurag Saraswat (M20CS066)
Parsa Revanth (M20CS058)

Date: May 23, 2021

1 SUMMARY

The authors examined different strategies to integrate the pretrained language models to the sequence to sequence neural architectures. They examined these strategies on abstractive summarization and machine translation tasks. For evaluating the strategies on the machine translation task, BLEU (bilingual evaluation understudy) [4] score is used and for evaluation of strategies on the abstractive summarization task, ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [3] scores are used. The authors considered two models in the baseline sequence to sequence model based on the sharing of the tokens. If in the baseline sequence to sequence model the tokens of both encoder inputs and decoder inputs, outputs are shared then they considered that model as SHARED. If in the baseline sequence to sequence model the tokens of the only decoder inputs and outputs are shared then they consider that model as SHDEMB.

There are two types of pre-training models authors used which are pretrained Bi-transformer and pretrained unidirectional language model. Depending on the token representation and parameter updates there are two approaches which are ELMo augmentation approach and fine tuning approach. There are a total of six models on which the authors calculated the BLEU score in machine translation task and ROUGE score in abstractive summarisation task. The six models are SHARED model, SRC-ELMO model, SRC-FT model, TGT-ELMO model, TGT-FT model, SRC-ELMO combined with SHDEMB model. The experimental results shows that using the pretrained tokens for the encoder will increase the BLEU score and ROUGE score [2].

2 ARCHITECTURES/MODELS DISCUSSED IN THE PAPER

Baseline sequence to sequence model is also considered as two different types based on the embeddings sharing between encoder and decoder

(1) Decoder input and output embeddings are shared then it is SHDEMB model

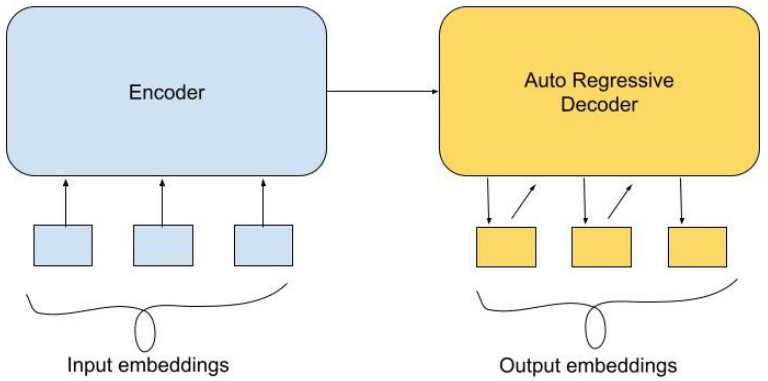

Fig. 1. SHDEMB

(2) If the dictionary is shared that is token embeddings for both encoder inputs and decoder input, output are shared then it is SHARED model

Based on the type of pretraining used there are two types of models,

(1) Pre-trained Bi-Transformer used for the generating the embeddings and later those embeddings are passed to the encoder

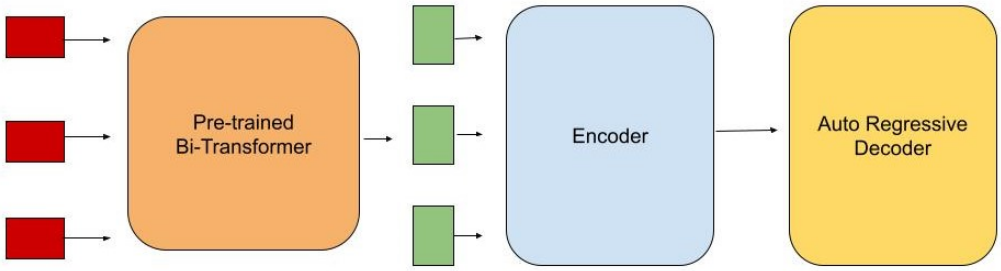

Fig. 2. Architecture using pre-trained Bi-Transformer

The red boxes represent the input embeddings. The green boxes represent the embeddings corresponding to the input red boxes generated by passing through pertained bi-transforemer.

(2) Pre-trained unidirectional language model used for the generating the embeddings at the decoder side

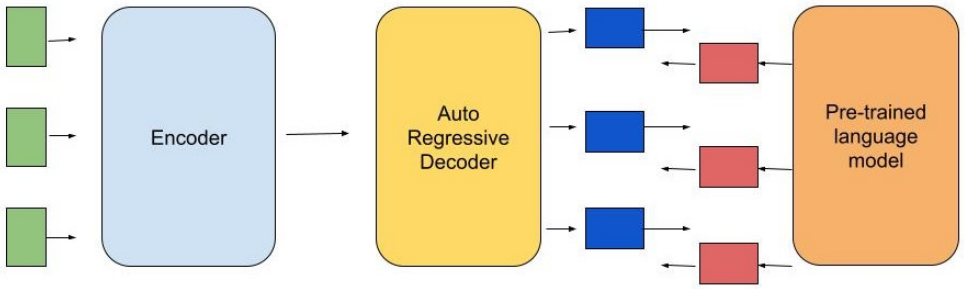

Fig. 3. Architecture using pre-trained unidirectional language model

The green boxes represent the input embeddings. The BLEU boxes represent the output embeddings from the auto regressive decoder. The red boxes represent the embeddings generated by passing the output embeddings to pretarined language model which are then passed to the decoder.

Depending on the type of token representation and parameter updates there are two styles in each architecture

(1) **ELMo augmentation:** We get the contextualized word embeddings based on the language model representations without adjusting the actual language model parameters. Layer normalisation is applied to the outputs of the language model layer before computing the ELMo vectors [5]

(2) **Fine tuning approach:** Fine-tuning the pre-trained representations adjusts the language model parameters by the learning signal of the end-task. We replace learned input word embeddings in the encoder network with the output of the language model [1] [6]

Models created and used in the paper are,

(1) **SHARED model:** In this the tokens of both encoder and decoder are shared and use baseline sequence to sequence architecture.

(2) **SRC-ELMO model:** In this the pretrained bi-transformer model is used. The token representation is the linear combination of representations of all the layers. The parameter update

is the linear combination of weights are learned but the bi-transformer parameters are not updated

(3) **SRC-FT model:** In this the pretrained bi-transformer model is used. The token representation is the representation from the top most layer. The parameter update is the linear combination of weights are learned and the bi-transformer parameters are also updated

(4) **TGT-ELMO model:** In this the pretrained unidirectional language model is used. The token representation is the linear combination of representations of all the layers. The parameter update is the linear combination of weights are learned but the language model parameters are not updated

(5) **TGT-FT model:** In this the pretrained unidirectional language model is used. The token representation is the representation from the top most layer. The parameter update is the linear combination of weights learned and the language model parameters are also updated.

(6) **SRC-ELMO + SHDEMB model:** This is the special case model which is used because experimentally the SRC-ELMO model gave the best scores of both BLEU and ROUGE. SHDEMB is the model where the decoder input output embeddings are shared.

3 EXPERIMENTS:

The authors used the CNN-DailyMail dataset for evaluating the abstractive summarisation task on all the models and the WMT'18 English newscrawl dataset is used for the machine translation task.

The machine translation task is performed for six models.

(1) For the SHARED model the BLEU delta wrt baseline for the dataset of size 160K is around 3 but it decreases as the dataset size increases gradually and for the dataset size of 5186K BLEU delta wrt baseline is close to 0.

(2) For the SRC-ELMO model the BLEU delta wrt baseline for the dataset of size 160K is around 3.8 but it decreases as the dataset size increases gradually and for the dataset size of 5186K BLEU delta wrt baseline is close to 1.

(3) For the SRC- FT model the BLEU delta wrt baseline for the dataset of size 160K is around 2.9 but it decreases as the dataset size increases gradually and for the dataset size of 5186K BLEU delta wrt baseline is close to 0.2.

(4) For the TGT-ELMO model the BLEU delta wrt baseline for the dataset of size 160K is around 0.3 but it goes to negative values for 320K and 640K size dataset and becomes close to zero for the 5186K size dataset.

(5) For the TGT-FT model the BLEU delta wrt baseline for the dataset of size 160K is around 2 but it decreases as the dataset size increases gradually and for the dataset size of 5186K BLEU delta wrt baseline is close to -1.

(6) For the SRC-ELMO + SHDEMB model the BLEU delta wrt baseline for the dataset of size 160K is 5.8 but it decreases as the dataset size increases gradually and for the dataset size of 5186K BLEU delta wrt baseline is 0.9.

Based on the BLEU scores the authors inferred that SRC-ELMO + SHDEMB model is the best model. They used that model for abstractive summarization task and yielded a ROUGE-1, ROUGE-2, and ROUGE-L scores as 41.56, 18.94, 38.47 respectively as opposed to the baseline architecture attaining the scores as 40.07, 17.61, and 36.78 respectively.

4 CRITICAL DISCUSSION

The first inference is that not all possible combinations of models give the best evaluation metric for any tasks. The model where we use the pretrained model embeddings from a bi-transformer with the token representation as the linear combination of representations of all the layers and

the parameter update as the linear combination of weights are learned but the bi-transformer parameters are not updated. Also the above mentioned model combined with the model which uses the decoder input and output embeddings sharing in the model gave the best metrics evaluated. The time taken by the models is 10%-14% more than the baseline sequence to sequence model is the limitation. The authors considered the model which has more BLEU score and used it as the best model and did abstractive summarization task on it. They didn't calculate the ROUGE score for the remaining models.

5 CONCLUSIONS

We conclude that pretraining token embeddings or pretarining language representations are helpful in getting the better metric scores. It also depends on where we are using the pretrained embeddings in the architecture because pretrained embeddings used for encoder gave best results. Ironically the pretrained embeddings used near decoder deterred as the size of the dataset increases and after certain dataset size the baseline sequence to sequence model gave better metrics than pretrained embeddings used near decoder models.

REFERENCES

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[2] S. Edunov, A. Baevski, and M. Auli. Pre-trained language model representations for language generation. *arXiv preprint arXiv:1903.09722*, 2019.

[3] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[4] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[5] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[6] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. 2018.