BERT Technology introduced in 3-minutes

Suleiman Khan, Ph.D. Follow
Feb 1 · 4 min read

Google BERT is a pre-training method for natural language understanding that performs various NLP tasks better than ever before.

BERT works in two steps, First, it uses a large amount of unlabeled data to learn a language representation in an unsupervised fashion called **pre-training**. Then, the pre-trained model can be **fine-tuned** in a supervised fashion using a small amount of labeled trained data to perform various supervised tasks. Pre-training machine learning models have already seen success in various domains including image processing and natural language processing (NLP).

BERT stands for **B**idirectional **E**ncoder **R**epresentations from **T**ransformers. It is based on the transformer architecture (released by Google in 2017). The general transformer uses an encoder and a decoder network, however, as BERT is a pre-training model, it only uses the encoder to learn a latent representation of the input text.

Photo by Franki Chamaki on Unsplash

· · ·

Technology

BERT stacks multiple transformer encoders on top of each other. The transformer is based on the famous multi-head attention module which has shown substantial success in both vision and language tasks. For a review of attention see.

> *BERT's state-of-the-art performance is based on two things. First, novel pre-training tasks called* ***Masked Langauge Model(MLM)*** *and* ***Next Sentense Prediction (NSP)****. Second, a lot of data and compute power to train BERT.*

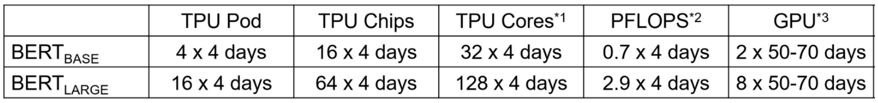
MLM makes it possible to perform bidirectional learning from the text, i.e. it allows the model to learn the context of each word from the words appearing both *before and after it*. This was not possible earlier! The previous state-of-the-art methods called Generative Pre-training used left-to-right training and ELMo used shallow bidirectionality.

The MLM pre-training task converts the text into tokens and uses the token representation as an input and output for the training. A random subset of the tokens (15%) are masked, i.e. hidden during the training, and the objective function is to predict the correct identities of the tokens. This is in contrast to traditional training methodologies which used either unidirectional prediction as the objective or used both left-to-right and right-to-left training to approximate bidirectionality. The NSP task allows BERT to learn relationships between sentences by predicting if the next sentence in a pair is the true next or not. For this 50% correct pairs are supplemented with 50% random pairs and the model trained. BERT trains both MLM and NSP objectives simultaneously.

. . .

Data and TPU/GPU Runtime

BERT was trained using 3.3 Billion words total with 2.5B from Wikipedia and 0.8B from BooksCorpus. The training was done using TPU, GPU estimates are shown below.



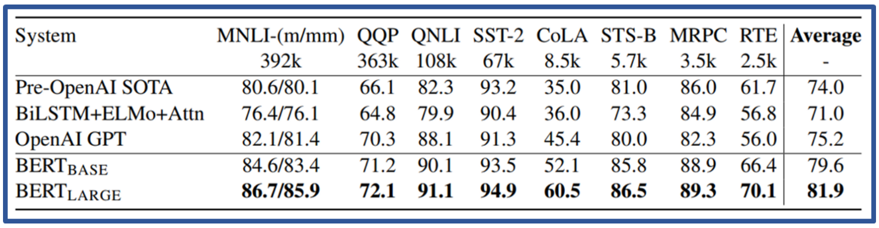
	TPU Pod	TPU Chips	TPU Cores^{*1}	PFLOPS^{*2}	GPU^{*3}
$\text{BERT}_{\text{BASE}}$	4 x 4 days	16 x 4 days	32 x 4 days	0.7 x 4 days	2 x 50-70 days
$\text{BERT}_{\text{LARGE}}$	16 x 4 days	64 x 4 days	128 x 4 days	2.9 x 4 days	8 x 50-70 days

Training devices and times for BERT; used TPU and estimated for GPU.

Fine-tuning was done using 2.5K to 392K labeled samples. Importantly, datasets above 100K training samples showed robust performance over various hyper-parameters. Each fine-tuning experiment runs within 1 hour on a single cloud TPU and few hours on GPU.

Results

BERT outperforms 11 state-of-the-art NLP tasks with large margins. The tasks fall in three main categories, text classification, textual entailment, and Q/A. On two of the tasks SQUAD and SWAG, BERT is the first to outperform the human level performance!



System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.9	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	88.1	91.3	45.4	80.0	82.3	56.0	75.2
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.1	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**91.1**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**81.9**

BERT results from the paperhttps://arxiv.org/abs/1810.04805

Using BERT in your analysis

BERT is available as open source: https://github.com/google-research/bert and pre-trained for 104 languages with implementations in TensorFlow and Pytorch.

It can be fine-tuned for several types of tasks, such as text classification, text similarity, question and answer, text labeling such as parts of speech, named entity recognition etc. However, pre-training BERT can be computationally expensive unless you use TPU's or GPU's similar to the Nvidia V100.

BERT folks have also released a single multi-lingual model trained on entire Wikipedia dump of 100 languages. Multilingual BERT is has a few percent lower performance than those trained for a single language.

Critique

The BERT masking strategy in MLM biases the model towards the actual word. The impact of this bias on the training is not shown.

References

[1] https://cloud.google.com/tpu/docs/deciding-pod-versus-tpu

[2] Assuming second generation TPU, 3rd generation is 8 times faster. https://en.wikipedia.org/wiki/Tensor_processing_unit

[3] http://timdettmers.com/2018/10/17/tpus-vs-gpus-for-transformers-bert/