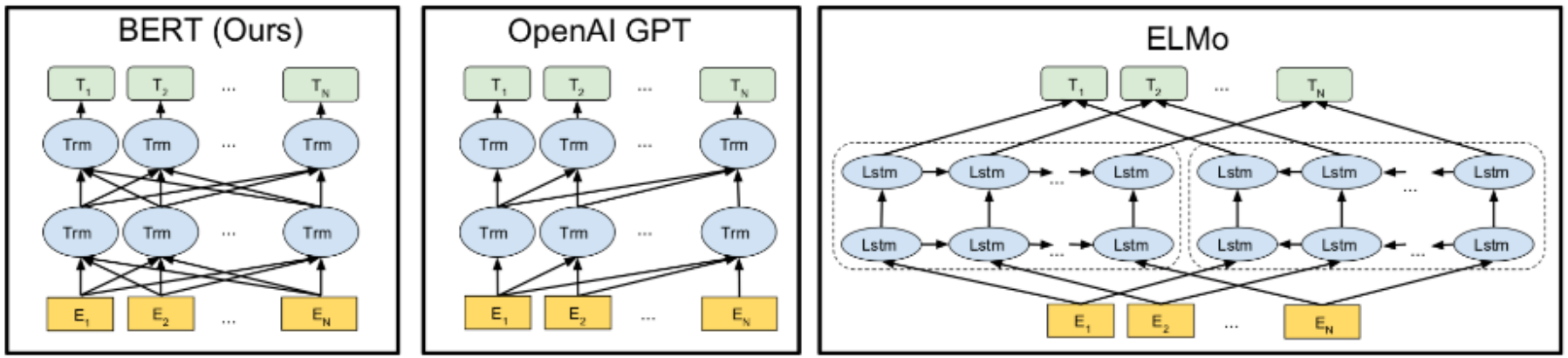

NVIDIA在BERT领域的支持分享
NVIDIA
2019年4月

议题

- Bert概述
- Nvidia对于Bert的加速 – 训练
- Nvidia对于Bert的加速 – 推理

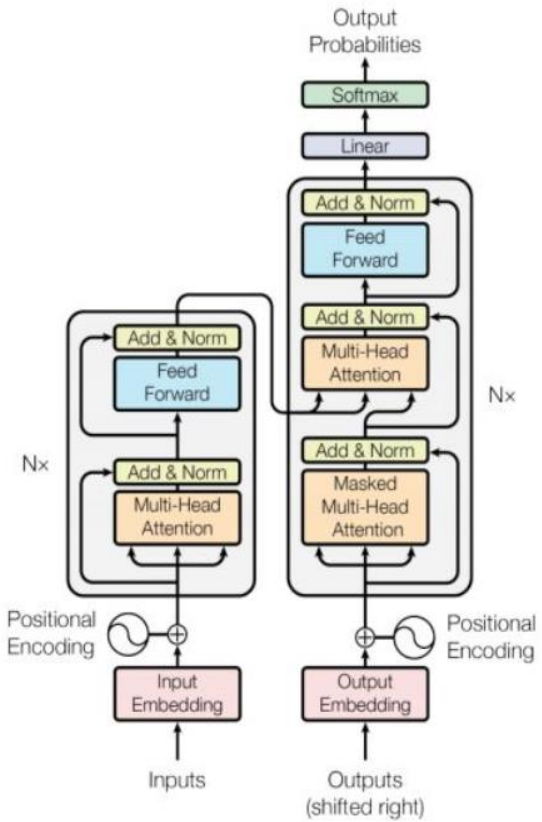
BERT概述 - 模型及其意义

- BERT - Bidirectional Encoder Representations from Transformers，双向Transformer的Encoder，如左图



- BERT的“里程碑”意义：证明了一个非常深的模型（ Transformer ）可以显著提高NLP任务的准确率，而这个模型可以从无标记数据集中预训练得到
- 特点：成熟模型 + 大数据量， Google 风格的暴力模型

BERT概述 – TRANSFORMER模型结构



Transformer模型结构:

- 用全Attention的结构代替了传统的LSTM
- 成熟模型:在翻译任务已经取得好成绩
- 可以多层叠加:
  - Google Base模型:12层，用于微调
  - Google Large模型:24层，用于预训练

Google Bert训练数据集:

- BooksCorpus: 8 亿词量
- Wikipedia: 25 亿词量

BERT概述 - 应用场景

NLP任务

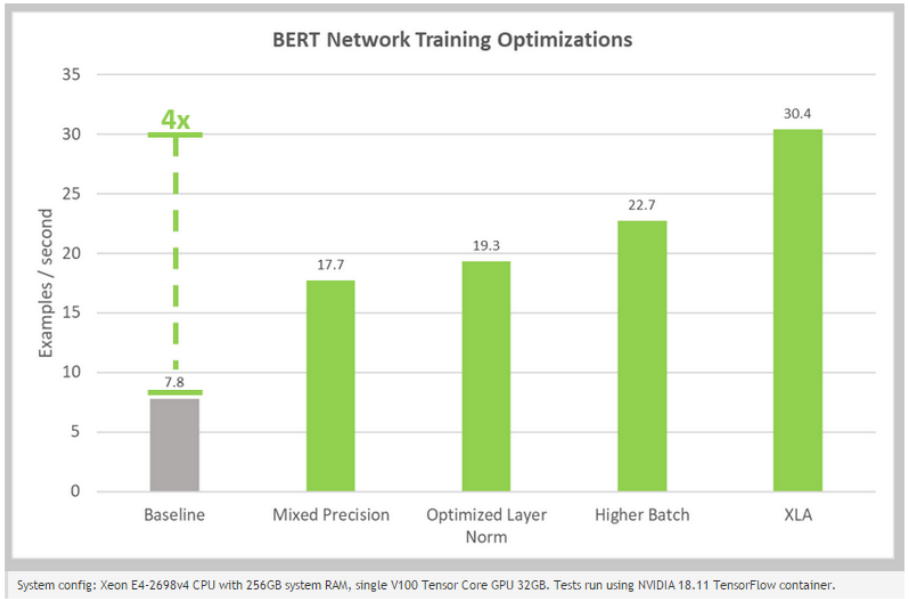
- 序列标注：分词/pos tag/语义标注...
- 分类任务：文本分类/情感计算...
- 句子关系判断：Entailment/QA/自然语言推理...
- 生成式任务：机器翻译/文本摘要...

System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	Average -
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
$BERT_{BASE}$	84.6/83.4	71.2	90.1	93.5	52.1	85.8	88.9	66.4	79.6
$BERT_{LARGE}$	**86.7/85.9**	**72.1**	**91.1**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**81.9**

业务场景

- 搜索分词
- 文本/评论分类
- 情感识别
- 搜索/推荐中的特征判断
- 个性化推荐语
- 问答/人机对话 / 智能客服
- 机器翻译 / 语音合成
- 。。。。。。

NVIDIA对于BERT的加速 - 训练



Base Line: ~7.81 samples/s

@BERT_LARGE, @Sequence Length=512, @bs=8, @V100 32GB

Using NVIDIA V100 Tensor Core GPUs :

1) Mixed precision training -> speedup of 2.3x;
2) Layer Norm in cuDNN BN primitive -> speedup of 1.09x;
3) Large BS(8->16) -> speed up of 1.18x;
4) XLA -> speed up of 1.34x.

TensorFlow version:

4x acceleration version on BERT training: https://github.com/google-research/bert/pull/255

70% extension on BERT training(Horovod): https://github.com/thorjohnsen/bert/tree/gpu_optimizations

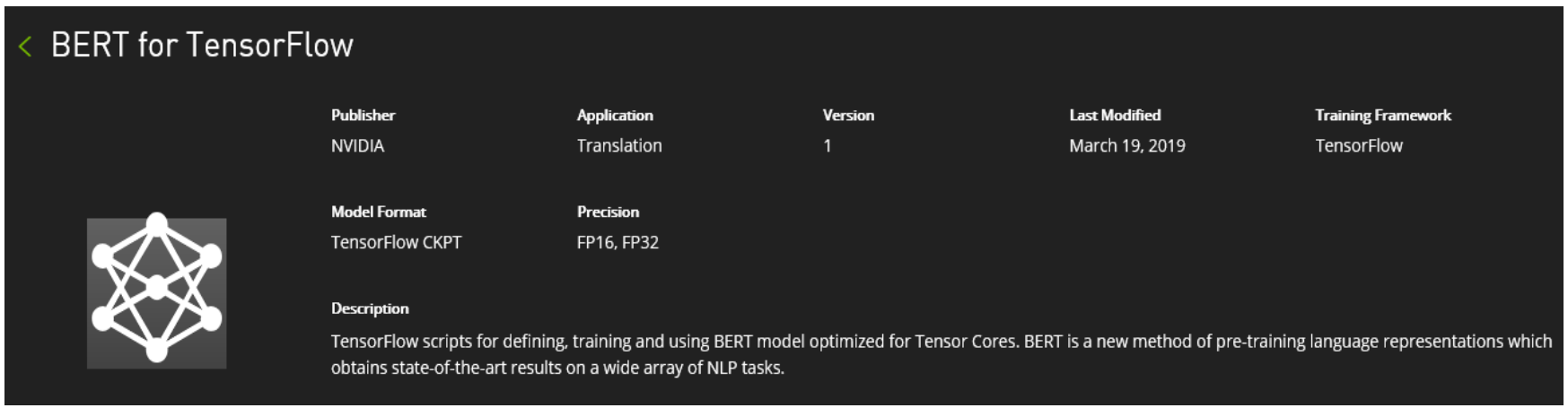
PyTorch version: without XLA optimization

2.5x acceleration version on BERT training: https://github.com/huggingface/pytorch-pretrained-BERT/pull/116

NVIDIA对于BERT的加速 - 训练

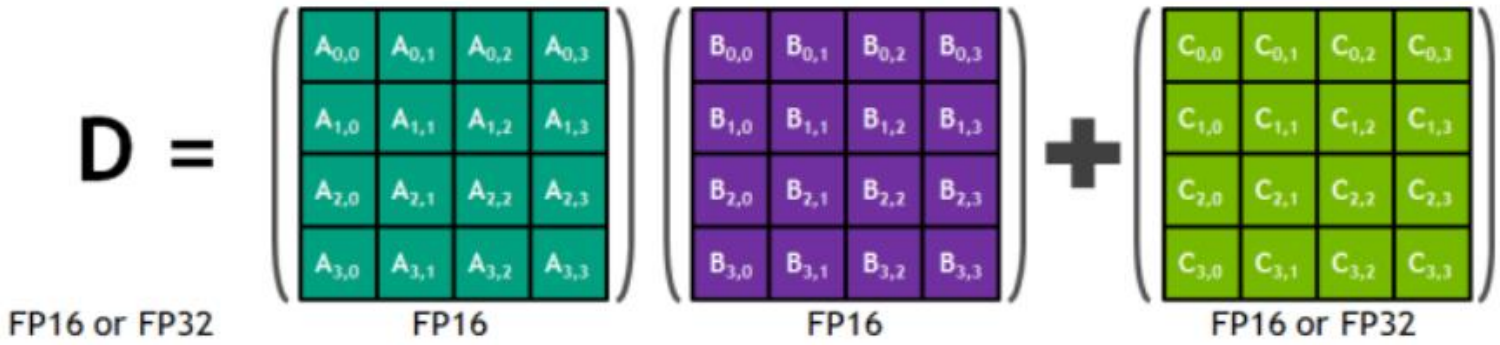
https://ngc.nvidia.com/catalog/model-scripts/nvidia:bert_for_tensorflow/

NGC 19.03+ TensorFlow container



1. Mixed precision support with TensorFlow Automatic Mixed Precision (TF-AMP), which enables mixed precision training without any changes to the code-base by performing automatic graph rewrites and loss scaling controlled by an environmental variable.
2. Scripts to download dataset for
   Pretraining - Wikipedia, BookCorpus
   Fine Tuning - SQuaD (Stanford Question Answering Dataset), Pretrained Weights from Google
3. Custom fused CUDA kernels for faster computations
4. Multi-GPU/Multi-Node support using Horovod
5. XLA support (experimental).

混合精度训练的基础 – TENSOR CORE

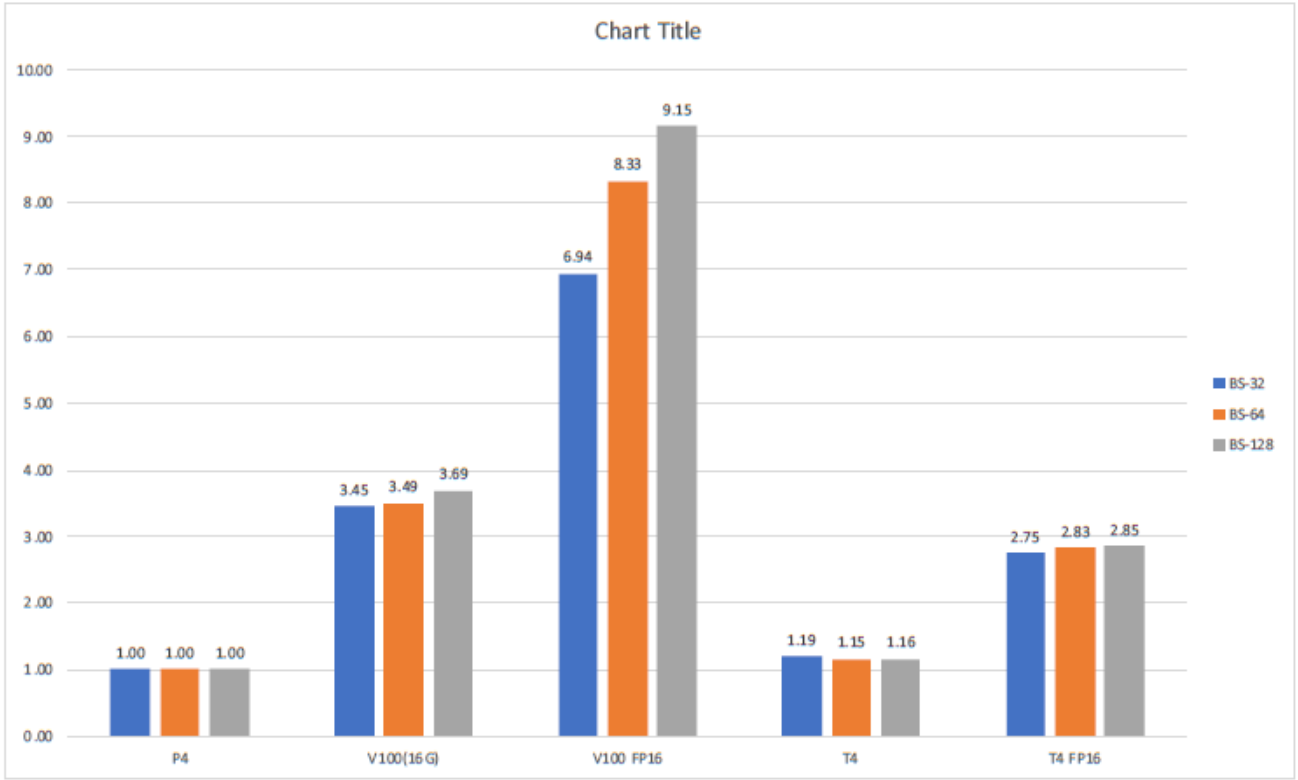


- Specialize for 4x4 matrix multiply and accumulate
- Total 640 tensor cores in NVIDIA Tesla V100 GPU, theoretical max performance: **125 TFLOPS**
  - 8-12x speedup vs. FP32
  - 2-3x end-2-end speedup of training task
- More info about Tensor Core and NVIDIA Volta GPU, please refer to: https://devblogs.nvidia.com/inside-volta/

自动混合精度训练 - NVIDIA AMP

- Automatic Mixed Precision for Deep Learning: https://developer.nvidia.com/automatic-mixed-precision
- AMP for Tensorflow: https://devblogs.nvidia.com/nvidia-automatic-mixed-precision-tensorflow/
  - Graph optimization and automatic loss scaling
  - Only available in NVIDIA Tensorflow docker image currently:
    - Register at NVIDIA GPU CLOUD: http://ngc.nvidia.com
    - PULL the container: docker pull nvcr.io/nvidia/tensorflow:19.03-py3
- AMP for PyTorch with APEX: https://devblogs.nvidia.com/apex-pytorch-easy-mixed-precision-training/
  - Open source on github: https://github.com/NVIDIA/apex
- AMP for MXNet:
  - Work in Progress. PR: https://github.com/apache/incubator-mxnet/pull/14173

NVIDIA对BERT的加速 - 推理

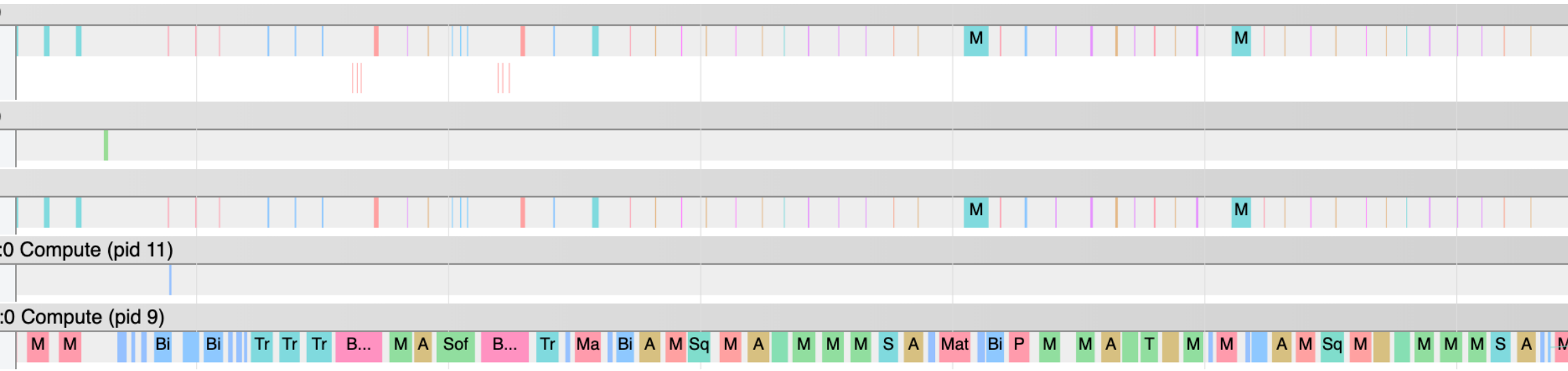


- FP16
- TRT / TFTRT
- 深度优化op

batch_size=128
hidden_size=768
max_pos_embed=512
num_attention_heads=12
num_hidden_layers=12
max_seq_len=128
samples=1725
ngc:tf-19.01

NVIDIA对BERT的加速 - 推理

- batch_size = 1, seq_len = 64, Base Model:



- Inference慢的原因分析:CPU频繁调用GPU kernel成为瓶颈
- op developed by Nvidia DevTech: 针对Transformer layer的加速
  - 适用场景: batch size 小(1/2/4/8/16), 句子长度比较短(Seq.len = 16/3264/128), with FP32
  - 加速效果:50%+, 具体依赖于Transformer层数、CPU性能、GPU型号


NVIDIA