OPENSEQ2SEQ: A DEEP LEARNING TOOLKIT FOR SPEECH RECOGNITION, SPEECH SYNTHESIS, AND NLP

Boris Ginsburg, Vitaly Lavrukhin, Igor Gitman, Oleksii Kuchaiev, Jason Li, Vahid Noroozi, Ravi Teja Gadde, Chip Nguyen

18/10/2018

SUMMARY

OpenSeq2Seq is a TensorFlow toolkit based on sequence-to-sequence paradigm

- Modular design
- Mixed precision training optimized for Volta GPU
- Scalable multi-GPU training with Horovod

State-of-the art models

- Automatic Speech Recognition: DeepSpeech2, Wave2Letter
- Language Translation: GNMT, Transformer, ConvS2S
- Text-To-Speech: Tacotron2

WHY WE BUILT OPENSEQ2SEQ

The project started in January 2018, when we found that

- Adding float16 ("mixed precision") support to TF models is difficult:
  - TF approach based on "custom getters" practically didn't work for RNN, consumed large amount of memory, and was error-prone
- Multi-GPU training scales badly:
  - "native" TF tower-based scaling is inferior to Horovod (doesn't use NCCL, hard to implement multi-node training)
- There were no state-of-the art ASR models (DeepSpeech2, Wave2Letter) in TensorFlow

Need new high-level modular toolkit to solve these problems

OpenSeq2Seq: Modular Design

Model

Implements: parsing parameters, multi-GPU details, creating data_layer, optimizer

Abstract: graph creation, metrics logging, eval, infer

EncoderDecoderModel

Implements: graph creation - builds and connects Encoder, Decoder and Loss.

Abstract: metrics logging, eval, infer

...

Text2Text

Implements: metrics logging, eval, infer

Speech2Text

Implements: metrics logging, eval, infer

...

DataLayer

Implements: parsing/setting parameters

Abstract: tf.data object creation

Text2Text

Implements: tf.data object

...

Encoder

Implements: parsing/setting parameters accepts DayaLayer output

Abstract: graph creation

DeepSpeech2Encoder

Implements: graph creation

...

Decoder

Implements: parsing/setting parameters accepts Encoder output

Abstract: graph creation

RnnDecoderWithAttention

Implements: graph creation

...

Loss

Implements: parsing/setting parameters accepts Decoder output

Abstract: graph creation

CTCLoss

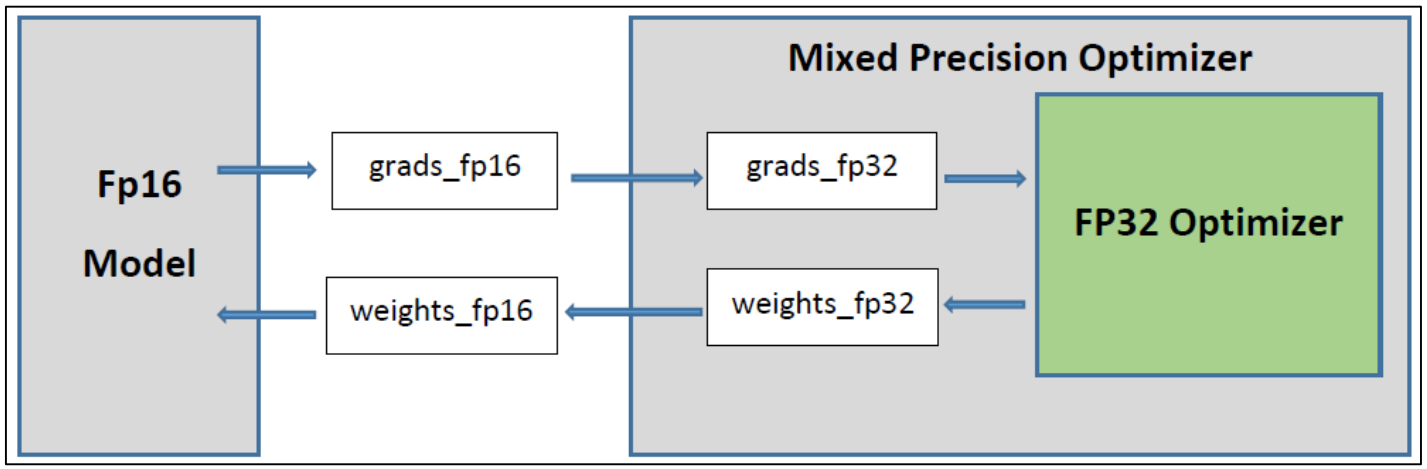
Implements: graph creation

...

```
def train(train_model, eval_model=None)
def eval(eval_model)
def infer(infer_model)
```

OpenSeq2Seq: Mixed Precision Training - 1

Optimizer wrapper



- Loss scaling with automatic back-off
- LARC (Layer-wise Automatic Rate Control) to speed-up and stabilize initial stage of training

OpenSeq2Seq: Mixed Precision Training - 2

TensorFlow Regularization

```
def l2_regularizer(scale):
  def l2(weights):
    return scale * nn.l2_loss(weights)
  return l2
```

OpenSeq2Seq: Mixed Precision Training - 2

Regularization wrapper

```
def mp_regularizer_wrapper(regularizer):
  def func_wrapper(weights):
    if weights.dtype.base_dtype == tf.float16:
      tf.add_to_collection('REGULARIZATION_FUNCTIONS’,
                           (weights, regularizer))
      # disabling the inner regularizer
      return None
    else:
      return regularizer(weights)
  return func_wrapper
```

```
reg_funcs = tf.get_collection('REGULARIZATION_FUNCTIONS’)

reg = tf.contrib.layers.apply_regularization(
    reg_funcs[var.name],[fp32_var]),

fp32_grad += tf.gradients(reg, fp32_var)[0]
```

OpenSeq2Seq: Multi-GPU training and inference

- **Tower-based approach**
  - Building a separate graph for each GPU
  - Data feeding with separate data layer per each GPU
  - No NCCL
- **Horovod-based approach**
  - Multi-GPU training is straightforward: optimizer wrapper and broadcast hook
  - It relies on NCCL for fast inter-GPUs communications

OpenSeq2Seq: Available models

Available models (with checkpoints and training recipes):

machine translation: GNMT, **Transformer**, **ConvS2S**

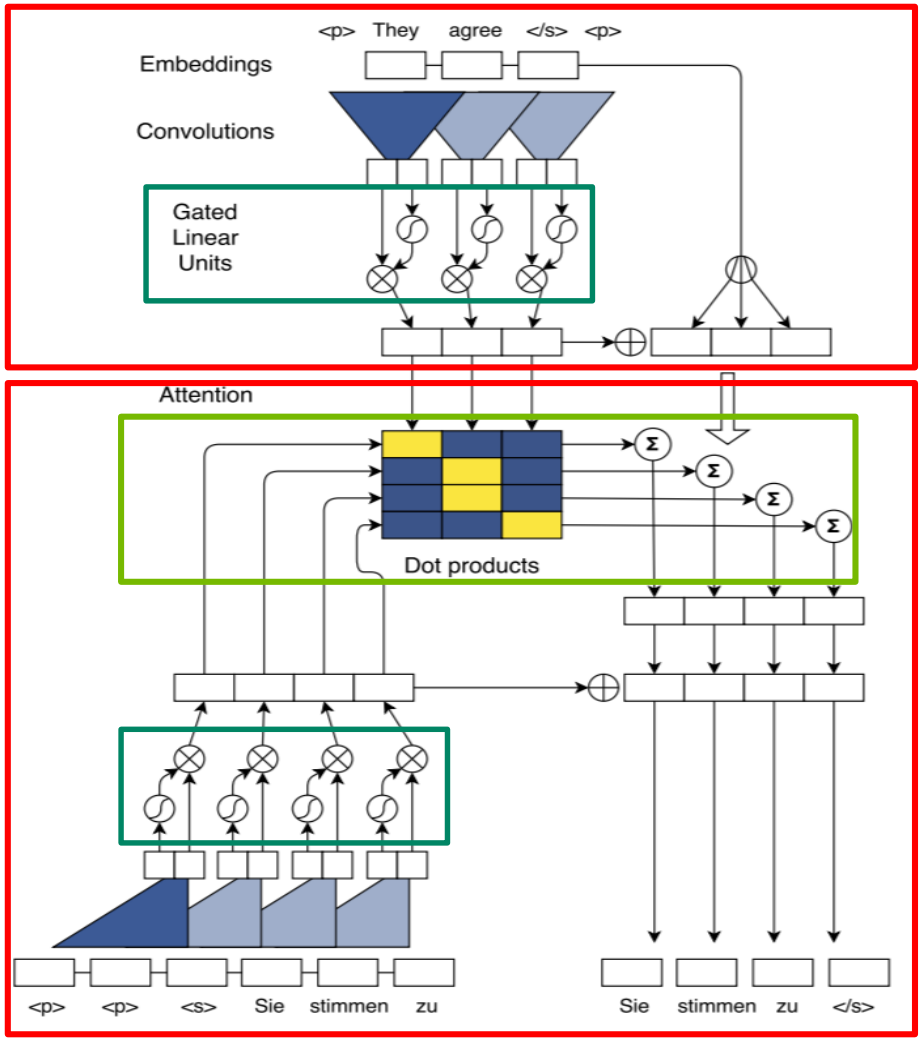
speech recognition: DeepSpeech2, **Wave2Letter**

text-to-speech: Tacotron-2

image classification: ResNets, AlexNet

Many different encoders / decoders / losses
to combine and play with!

Machine Translation: ConvS2S



- BLEU= 25 on News2014 EN-DE (better than GNMT)
- Convolutional encoder
- Convolutional decoder
  - Don't need inputs from previous steps➔parallel training
  - Easy control on maximum dependency length
- Gated Linear Units (GLU): $h(X) = (X*W+b) \otimes \sigma(X*V+c)$
- Separate attention for each decoder layer
- Position embedding
- Weight Normalization

Jonas Gehring et al. "Convolutional Sequence to Sequence Learning" *arXiv preprint arXiv:1705.03122* (2017)

ConvS2S: Mixed Precision Speed-up, 1 GPU

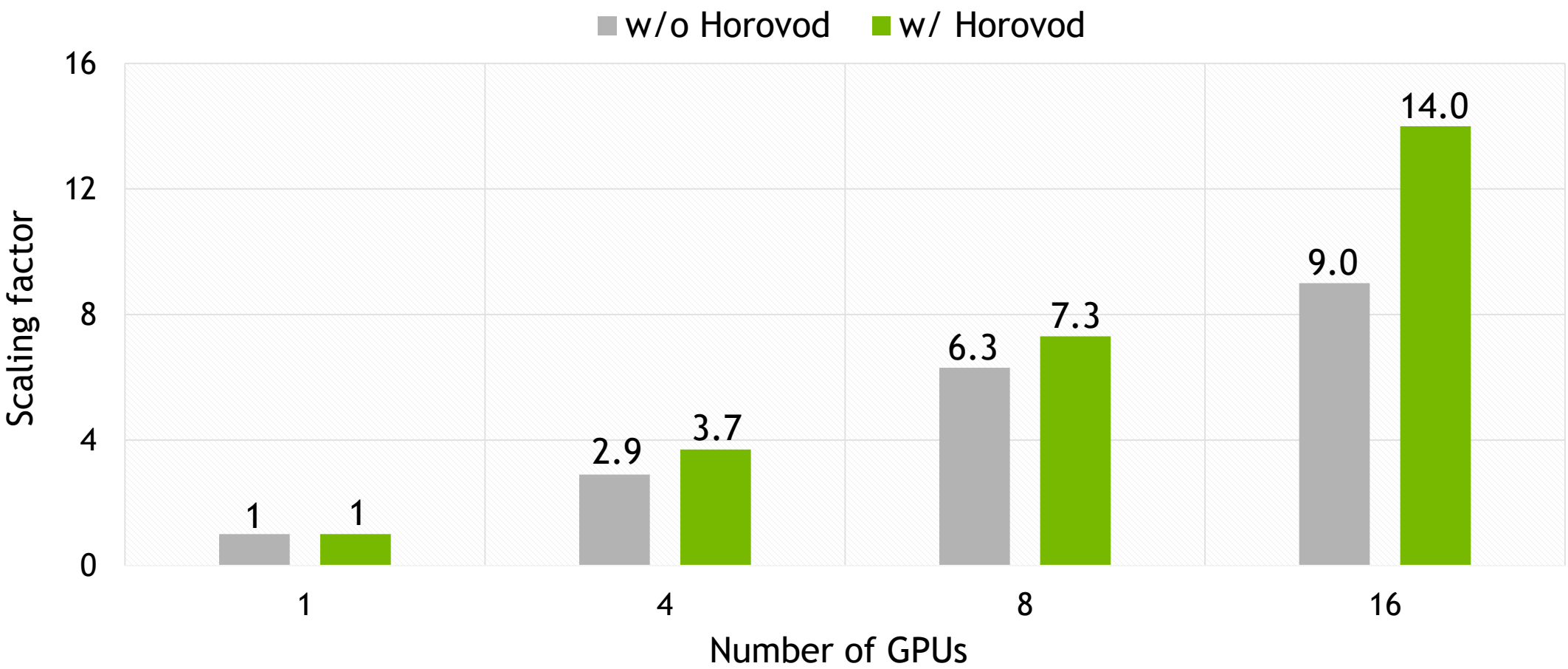
Batch/GPU	Speed-up
128	2.4x
256	**2.6x**

GPU: Single Quadro GV100 with 32GB Memory

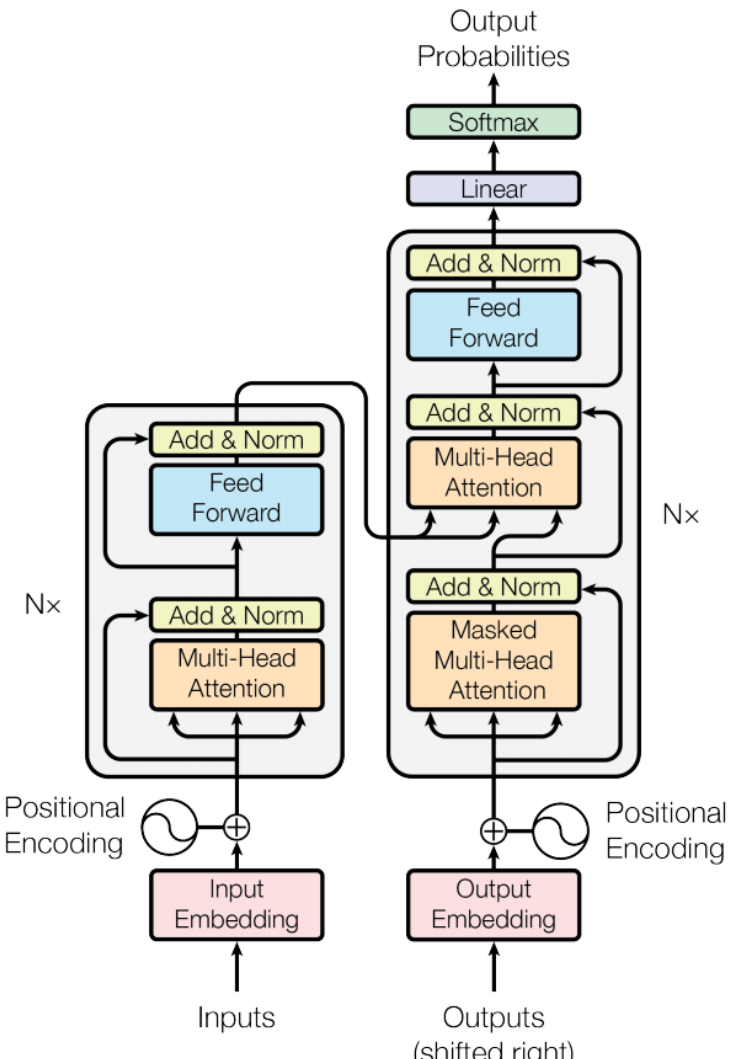
Driver version: 384.146, CUDA version: 9.0, cuDNN version: 7.0

TensorFlow: 18.06, Horovod: 0.13.10

ConvS2S: Multi-GPU scalability



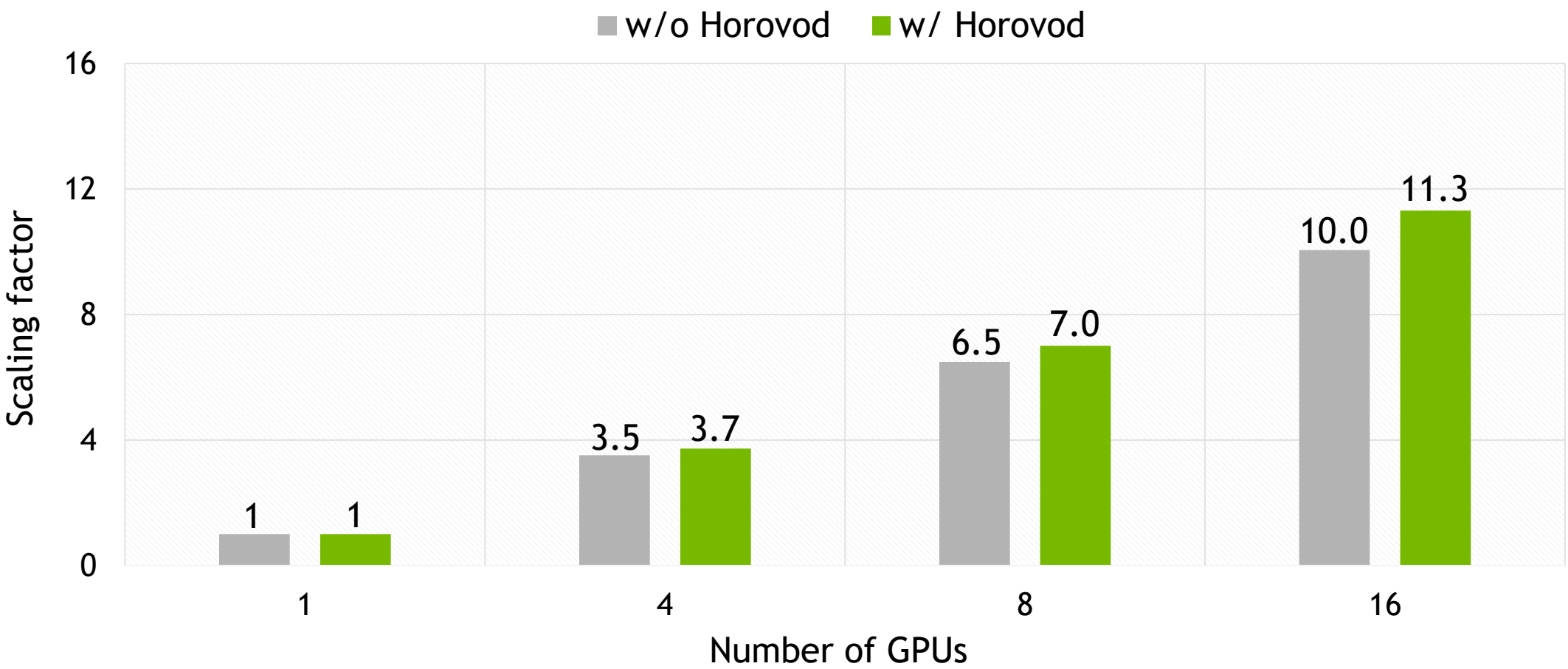
Machine Translation: Transformer



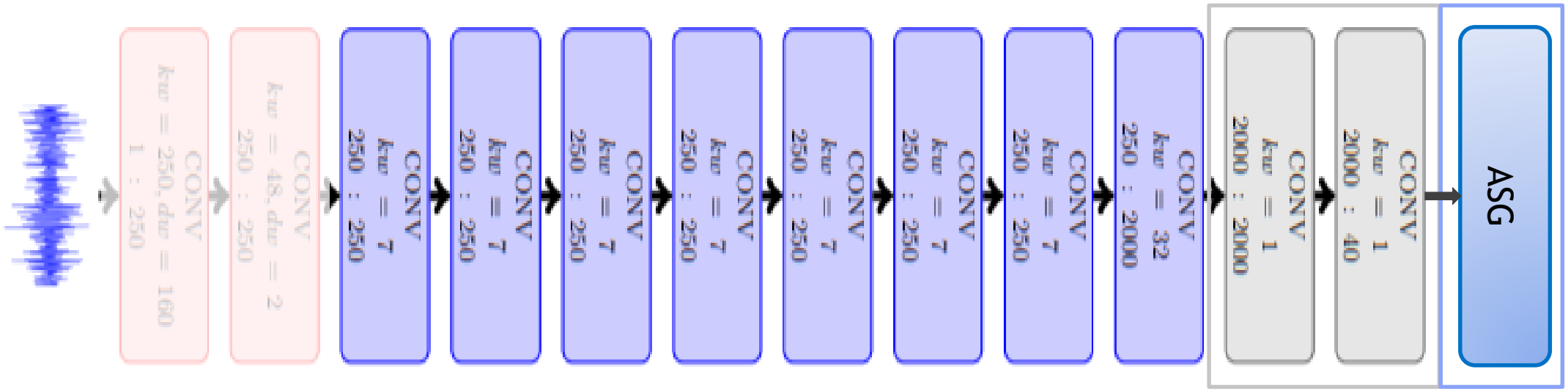
- BLEU = 28.0 on News2014 EN-DE (better than ConvS2S)
- No recurrent layers
- No convolutional layers
- Attention based
- SentencePiece tokenization (byte-pair-encoding)
- Layer Normalization
- Mixed precision training is **1.9x** faster (than float32)

Ashish Vaswani et al. “Attention is all you need"
arXiv preprint arXiv:1706.03762 (2017)

Transformer: Multi-GPU scalability



Speech Recognition: Wave2Letter



Wave2Letter V2:

- Gated Linear Units (GLU)
- Weight Normalization
- Gradient clipping
- ASG loss

Collobert, Ronan, Christian Puhrsch, and Gabriel Synnaeve. "Wav2letter: an end-to-end convnet-based speech recognition system." *arXiv preprint arXiv:1609.03193* (2016).

15

Wave2Letter+: Architecture modifications

- BatchNorm instead of Weight Norm
- Clipped ReLU instead of GLU
- Standard CTC loss
- LARC instead of Gradient clipping

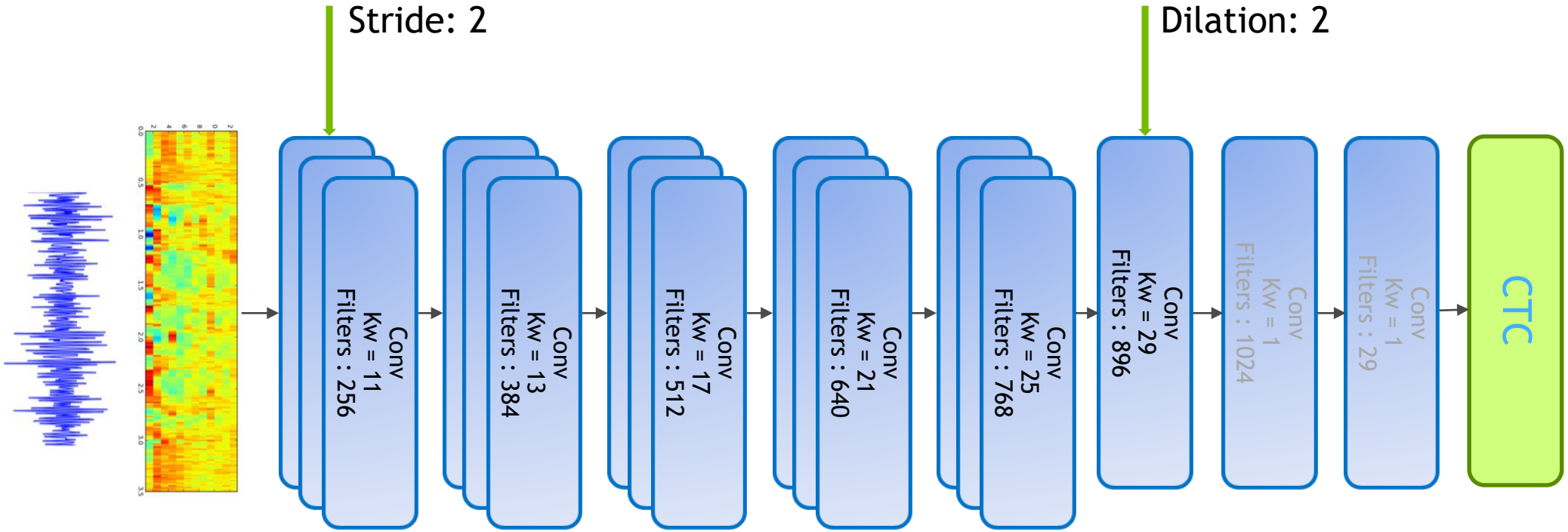
- Dilated Convolution:
  - Increase a context for the predictions

- Strided Convolution:
  - Decreases the length in time by 2x
  - Significant reduction in training time

Model	WER, %
Base (B)	10.53
B + Dilated Conv	**10.18**
B + Strided Conv	**8.59**

Trained for 50 epochs

Speech Recognition: Wave2Letter+



Wave2Letter+: Accuracy

LibriSpeech Dataset	WER, % Greedy Decoding	WER, % Beam Search with LM*
dev-clean	6.67	4.77
test-clean	6.58	4.92
dev-other	18.68	13.88
test-other	19.61	15.01

Training Configuration:

- 8 GPUs (NGC), 64 batch size
- Mixed Precision
- 200 Epochs, 44 Hours

* Language model is a standard 4-gram OpenSLR LM

Wave2Letter+: Mixed Precision Speed-Up, 1 GPU

Batch/GPU	Speed-up
160	3.22x
192	3.18x
352	**3.58x**

GPU : Single Quadro GV100-32GB
Driver Version : 384.146
CUDA Version : 9.0
cuDNN Version : 7.0
TensorFlow Version : 1.9

CONCLUSIONS

OpenSeq2Seq is **TensorFlow**-based toolkit that

- supports **distributed mixed precision** training out of the box
- contains implementations of **many SoTA models**

More models and features to come!

Contributions are welcome ☺

GitHub: https://github.com/NVIDIA/OpenSeq2Seq

Docs: https://nvidia.github.io/OpenSeq2Seq

20


NVIDIA