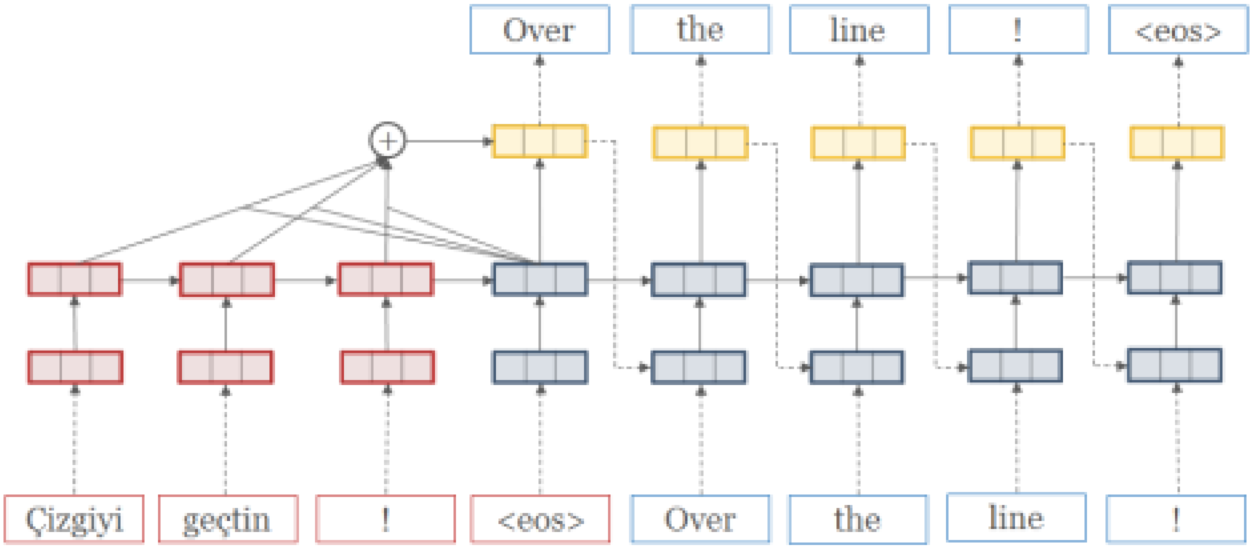
Hands-on Practical Session on Machine Translation
By: Dr. Sahinur Rahman Laskar

Instructions
On
Neural Machine Translation for English to German

Reference: OpenNMT https://github.com/OpenNMT/OpenNMT-py Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart and Alexander M. Rush. OpenNMT: Open-Source Toolkit for Neural Machine Translation. Proc. ACL (2017)

OpenNMT (https://opennmt.net/) is an open-source toolkit for neural machine translation and neural sequence learning.



Technique: RNN with LSTM model

~:Steps to be executed:~

Step 1: **Open Google-Colab** link: https://colab.research.google.com/

Step 2: **Google Colab Configuration**

Open the Google drive

Install Google Colaboratory

Mount the drive

Step 3: **Install OpenNMT tool using the link**

```
!wget https://github.com/OpenNMT/OpenNMT-py/archive/refs/tags/2.3.0.zip
!unzip 2.3.0.zip
!mv OpenNMT-py-2.3.0 OpenNMT-py
```

Step 4: Enter into the OpenNMT directory using the command

```
cd OpenNMT-py
```

Step 5: Install all the required packages of OpenNMT-py. Follow commands below one by one.

```
!pip install -e .
!pip install -r requirements.opt.txt
!pip install "numpy<2"

Note: Restart session
```

Step 6: Data Preparation

```
!wget https://s3.amazonaws.com/opennmt-trainingdata/toy-ende.tar.gz
!tar xf toy-ende.tar.gz
```

Step 7: Create Vocab File

Run Below Code:

```
%%bash
cat > /content/OpenNMT-py/toy_en_de.yaml << EOF
# toy_en_de.yaml

## Where the samples will be written
save_data: toy-ende/run/example
## Where the vocab(s) will be written
src_vocab: toy-ende/run/example.vocab.src
tgt_vocab: toy-ende/run/example.vocab.tgt
# Prevent overwriting existing files in the folder
overwrite: False

# Corpus opts:
data:
    corpus_1:
        path_src: toy-ende/src-train.txt
        path_tgt: toy-ende/tgt-train.txt
    valid:
        path_src: toy-ende/src-val.txt
        path_tgt: toy-ende/tgt-val.txt
EOF
```

```
!cat /content/OpenNMT-py/toy_en_de.yaml

!onmt_build_vocab -config /content/OpenNMT-py/toy_en_de.yaml -n_sample
10000
```

Step 8: Start Training

Run Below Code

```
%%bash
cat > /content/OpenNMT-py/toy_en_de.yaml << EOF
# toy_en_de.yaml

## Where the samples will be written
save_data: toy-ende/run/example
## Where the vocab(s) will be written
src_vocab: toy-ende/run/example.vocab.src
tgt_vocab: toy-ende/run/example.vocab.tgt
# Prevent overwriting existing files in the folder
overwrite: False

# Corpus opts:
data:
    corpus_1:
        path_src: toy-ende/src-train.txt
        path_tgt: toy-ende/tgt-train.txt
    valid:
        path_src: toy-ende/src-val.txt
        path_tgt: toy-ende/tgt-val.txt

# Vocabulary files that were just created
src_vocab: toy-ende/run/example.vocab.src
tgt_vocab: toy-ende/run/example.vocab.tgt

# Train on a single GPU
world_size: 1
gpu_ranks: [0]

# Where to save the checkpoints
```

```
save_model: toy-ende/run/model
save_checkpoint_steps: 500
train_steps: 1000
valid_steps: 500
EOF
```

```
!cat /content/OpenNMT-py/toy_en_de.yaml
!onmt_train -config /content/OpenNMT-py/toy_en_de.yaml
!sed -i 's/map_location=lambda storage, loc: storage)/map_location=lambda
storage, loc: storage, weights_only=False)/'
/content/OpenNMT-py/onmt/model_builder.py
```

Step 9: Start Translation

```
!onmt_translate -model toy-ende/run/model_step_1000.pt -src
toy-ende/src-test.txt -output toy-ende/pred_1000.txt -gpu 0
-verbose
```

Step 10: Evaluation: BLEU score

```
!chmod +x /content/OpenNMT-py/tools/multi-bleu-detok.perl
```

```
!rm bleuoutput.txt
! /content/OpenNMT-py/tools/multi-bleu-detok.perl
"/content/toy-ende/tgt-test.txt" < "toy-ende/pred_1000.txt" >>
bleuoutput.txt
!cat bleuoutput.txt
```

Now, you can use any language pair for machine translation.

Thank You!

Demonstration By: Dr. Sahinur Rahman Laskar