MTAT.06.055
Machine Translation
Practice Session 1: System Setup
Plan for today

- Logistics
- MT frameworks
- System setup
- Sequence-to-sequence models
- Train a sequence copy model
- Apply the trained model
- ...
Logistics

I am **Lisa Korotkova**, PhD student in NLP

**Communication:**  [piazza.com/ut.ee/spring2020/mtat06055](http://piazza.com/ut.ee/spring2020/mtat06055)

No office hours, message me to arrange a meeting

**Wed 16:15, Delta 2034:** exercises, troubleshooting, homework review, project support

4 homeworks + project
Good news

Plenty of NMT frameworks:

<table>
<thead>
<tr>
<th>Name</th>
<th>DL framework</th>
<th>Developed by</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenNMT</td>
<td>PyTorch / TensorFlow</td>
<td>Harvard NLP, SYSTRAN</td>
</tr>
<tr>
<td>Marian</td>
<td>(C++)</td>
<td>Microsoft Translator</td>
</tr>
<tr>
<td>Fairseq</td>
<td>PyTorch</td>
<td>Facebook AI</td>
</tr>
<tr>
<td>Nematus</td>
<td>TensorFlow</td>
<td>Edinburgh NLP</td>
</tr>
<tr>
<td>Sockeye</td>
<td>MXNet</td>
<td>Amazon</td>
</tr>
</tbody>
</table>
Good news

Plenty of NMT frameworks:

<table>
<thead>
<tr>
<th>Name</th>
<th>DL framework</th>
<th>Developed by</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenNMT</td>
<td>PyTorch / TensorFlow</td>
<td>Harvard NLP, SYSTRAN</td>
</tr>
<tr>
<td>Marian</td>
<td>(C++)</td>
<td>Microsoft Translator</td>
</tr>
<tr>
<td>Fairseq</td>
<td>PyTorch</td>
<td>Facebook AI</td>
</tr>
<tr>
<td>Nematus</td>
<td>TensorFlow</td>
<td>Edinburgh NLP</td>
</tr>
<tr>
<td><strong>Sockeye</strong></td>
<td><strong>MXNet</strong></td>
<td>Amazon</td>
</tr>
</tbody>
</table>
Installing Sockeye

1. Create conda virtual environment:
   ```bash
   conda create --name mtcourse python=3.6
   ```

2. Activate the new environment:
   ```bash
   conda activate mtcourse
   ```

3. Install NumPy:
   ```bash
   pip install numpy==1.14.0
   ```

4. Install Sockeye:
   ```bash
   pip install sockeye
   ```

5. Print Sockeye training help message:
   ```bash
   python -m sockeye.train -h
   ```
Sequence-to-sequence

Sequence-to-sequence

Neural Machine Translation
SEQUENCE TO SEQUENCE MODEL

ENCODER

DECODER

https://jalammar.github.io/images/seq2seq_4.mp4
Sequence-to-sequence

Recurrent Neural Network

Time step #1:
An RNN takes two input vectors:

Processes them

Then produces two output vectors:

Let’s train a model

See notebook

How to add your env to jupyter notebook/jupyter lab:

```
conda activate mtcourse
conda install ipykernel
ipython kernel install --user --name=<some_name>
```
Installing Sockeye (if you have a GPU)

1. Create conda virtual environment:
   
   ```bash
   conda create -n mtcourse python=3.6
   ```

2. Activate the new environment:
   
   ```bash
   source activate mtcourse
   ```

3. Download requirements file ($\{CUDA_VERSION\}$ can be 80 (8.0), 90 (9.0), 92 (9.2), or 100 (10.0)):
   
   ```bash
   wget https://raw.githubusercontent.com/awslabs/sockeye/master/requirements
   /requirements.gpu-cu$\{CUDA_VERSION\}.txt
   ```

4. Install Sockeye
   
   ```bash
   pip install sockeye --no-deps -r
   requirements.gpu-cu$\{CUDA_VERSION\}.txt
   ```

5. Remove requirements file
   
   ```bash
   rm requirements.gpu-cu$\{CUDA_VERSION\}.txt
   ```