

# Debugging Machine Translations

**Matīss Rikters**

The University of Tokyo

# MT Marathon 2017

## *Visualizing Neural Machine Translation Attention and Confidence*

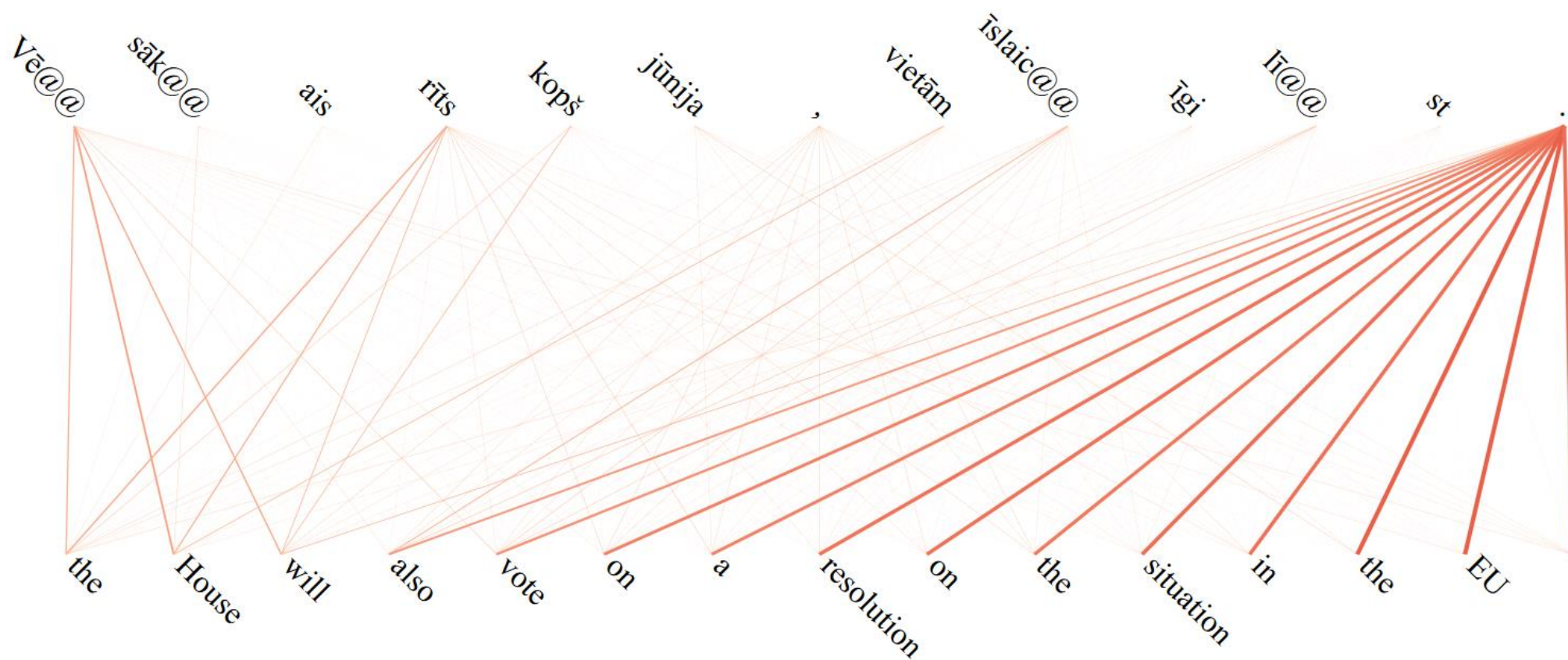
- Look at specific attention alignments in the terminal
- Browse whole test sets and find *suspicious* translations
  - Sort by length or 3 other confidence metrics
- Generate screenshots of example translations for publications, presentations, etc.

## Later additions

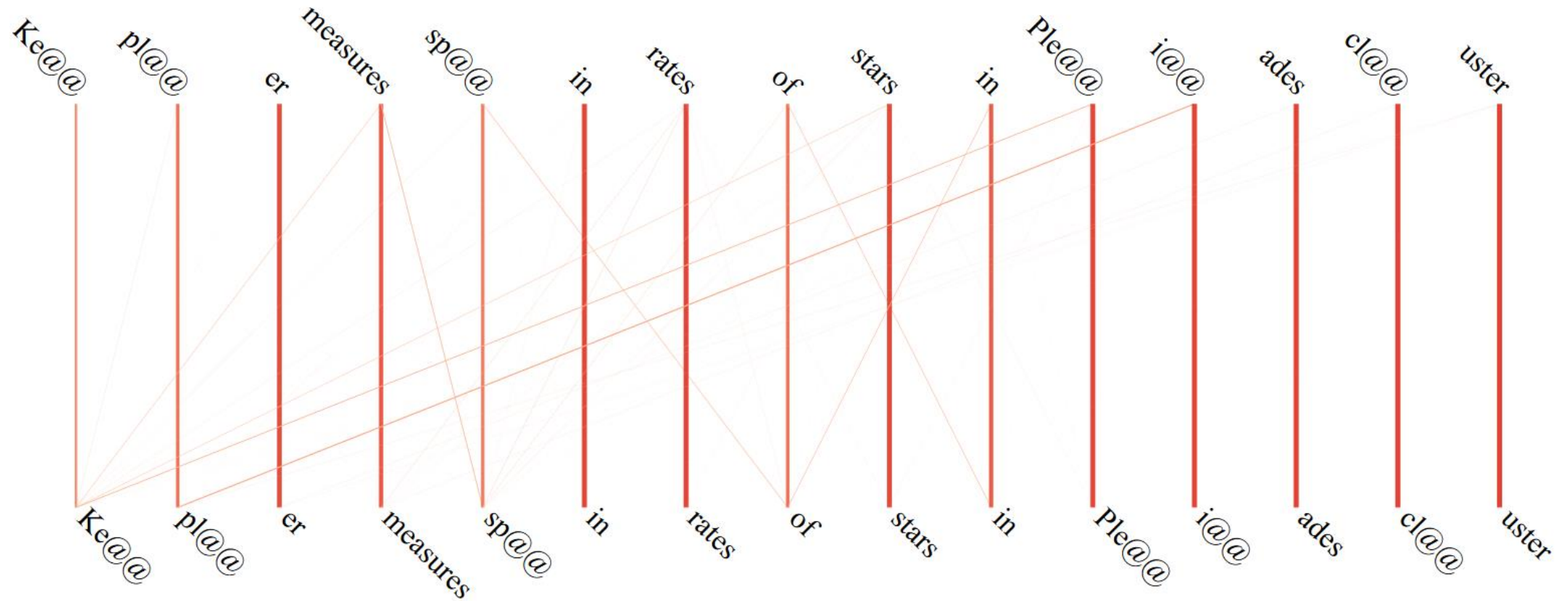
- Directly compare two translations of the same source
- Sort by BLEU or similarity to source
- Tweaked confidence score



# Unrelated Translations



# Untranslated Sentences



Check out the paper for more details

$$AP_{out} = -\frac{1}{L_{src}} \sum_i \sum_j \alpha_{ij} \cdot \log \alpha_{ij}$$

$$AP_{in} = -\frac{1}{L_{trg}} \sum_i \sum_j \alpha_{ij} \cdot \log \alpha_{ij}$$

$$CDP = \frac{1}{L_{src}} \sum_j \log \left( 1 + \left( \sum_i \alpha_{ji} \right)^2 \right)$$

$$confidence = CDP + AP_{out} + AP_{in}$$

# Requirements

- <https://github.com/M4t1ss/SoftAlignments>
  - This presentation, links to [models and data](#), scripts used in this tutorial are all in [\*\*assets/MT-Marathon-2019\*\*](#)
- Python 2 or 3
  - NLTK (for BLEU calculation)
- PHP 5.4 or newer (for web visualization)
- Translations and NMT attention alignments
  - + source text
  - + reference (for BLEU calculation)



# Model Specific Details

- RNNs mostly work
- Does anyone actually use CNNs for MT?
- Transformers have too many attention matrices
  - Use guided alignment in Marian
  - Use averaged attention in Sockeye (<https://github.com/aws-labs/sockeye/pull/504/files>)

# Guided Alignment in Marian

- Prepare data up to the part of splitting in subword units
- Run *fast\_align* on the subword training data
  - An example is in `assets/MT-Marathon-2019/scripts/fast_align.sh`
- Pass the resulting *grow-diag-final-and* file to Marian for training via the *--guided-alignment* parameter
  - An example is in `assets/MT-Marathon-2019/scripts/train.sh`
- Translate as usual
  - To get translations with attention alignments, pass the *--alignment soft* parameter to marian-decoder
  - An example is in `assets/MT-Marathon-2019/scripts/translate.sh` – use this to translate the prepared data
  - Marian outputs a slightly different alignment format than Nematus, Amun, OpenNMT. Use the included `format-output.py` to convert it



# NMT Attention Alignment Visualizations

<https://github.com/M4t1ss/SoftAlignments>

- Run examples from the readme using the tiny data sets that are in the repository
- Load the data you just translated using one of the models
  - Find some whacky translation examples
  - Are there any sentences completely untranslated?
- Translate the same data with both models and compare both outputs

# Find Some Buggy Translations

<https://github.com/M4t1ss/SoftAlignments>

- High BLEU, low other scores
- Low BLEU, high other scores
- Not so long sentences with some or all low scores

# You can use the online version

- <http://attention.lielakeda.lv>
- Examples from this lab have names starting with MTM-
- Check out other datasets that are there