Neural machine translation Pydata2017 Amsterdam

Rob Romijnders

RomijndersRob@gmail.com
robromijnders.github.io

April 9, 2017

Overview

GMail reply

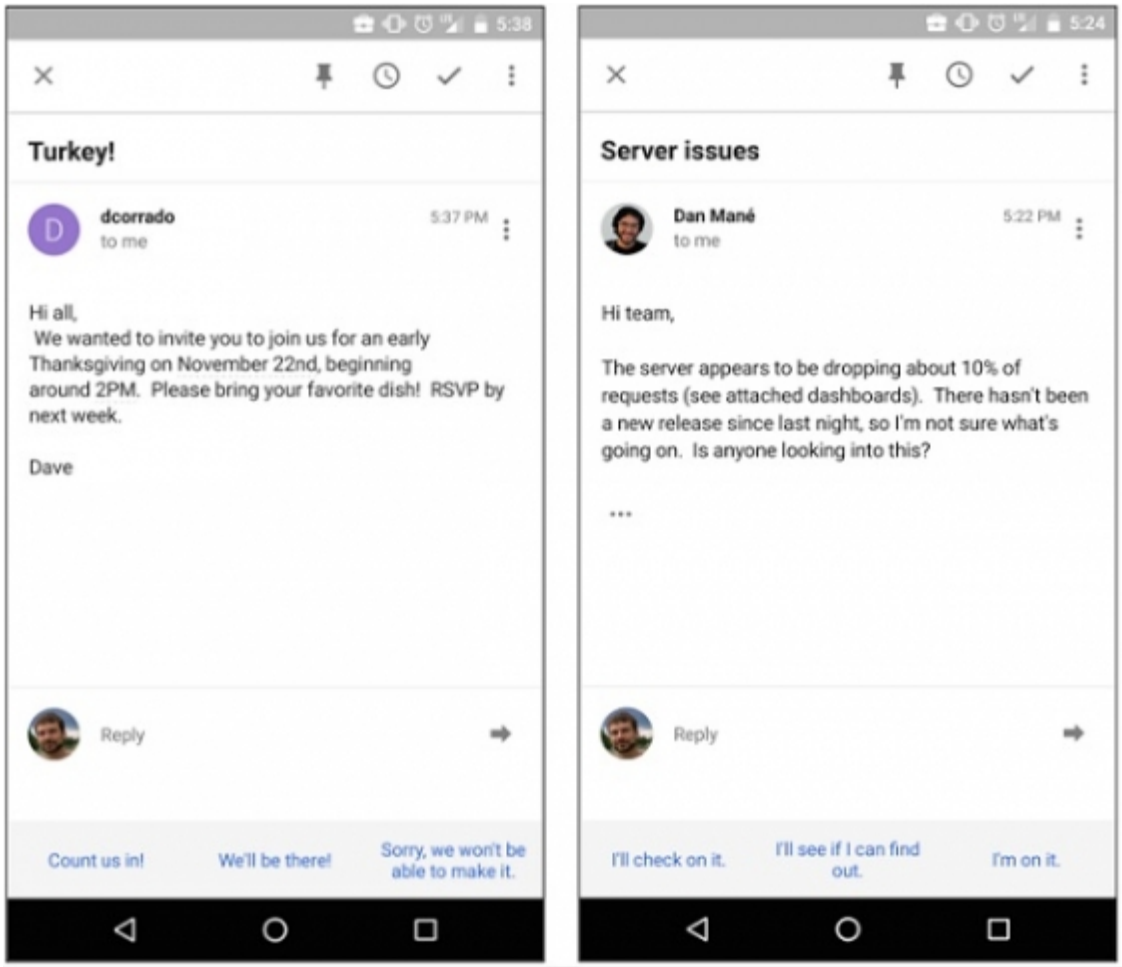


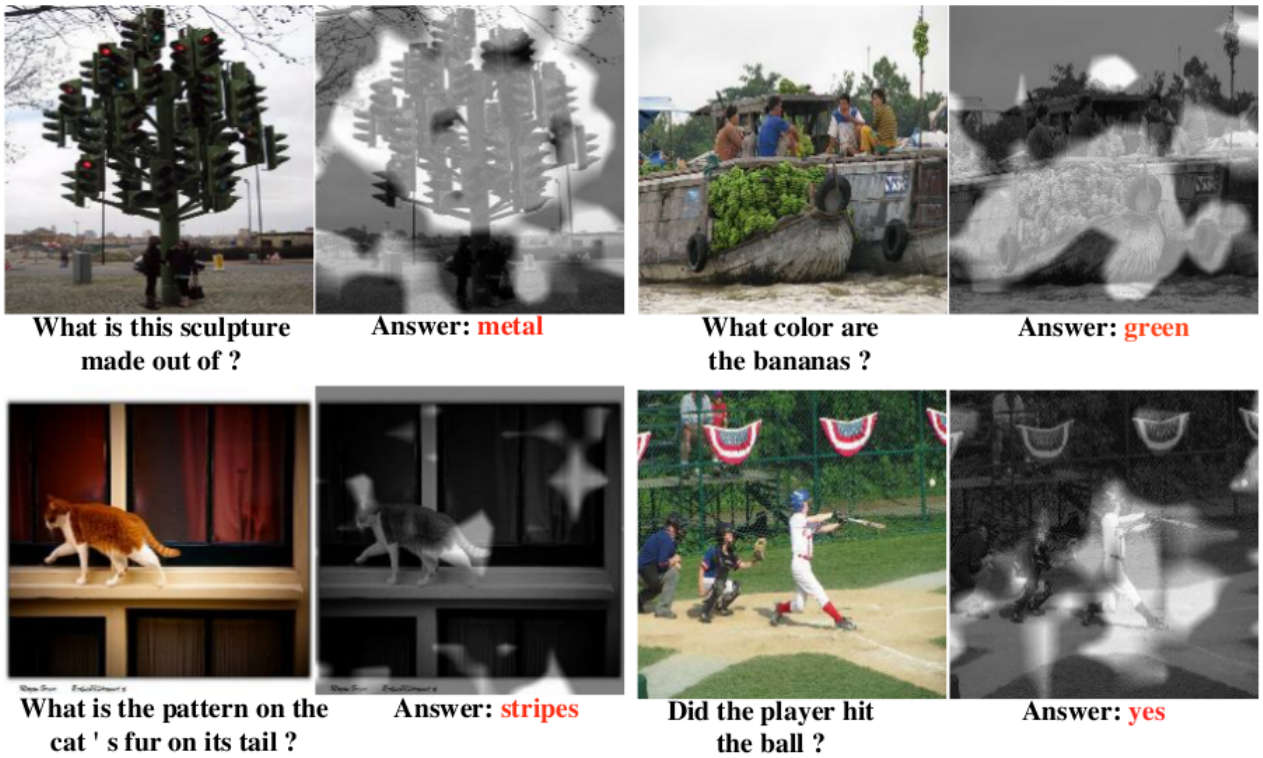
Photo: Greg Corrado, Google Research Blog

Apple siri

Photo: cultofmac.com

Question Answering

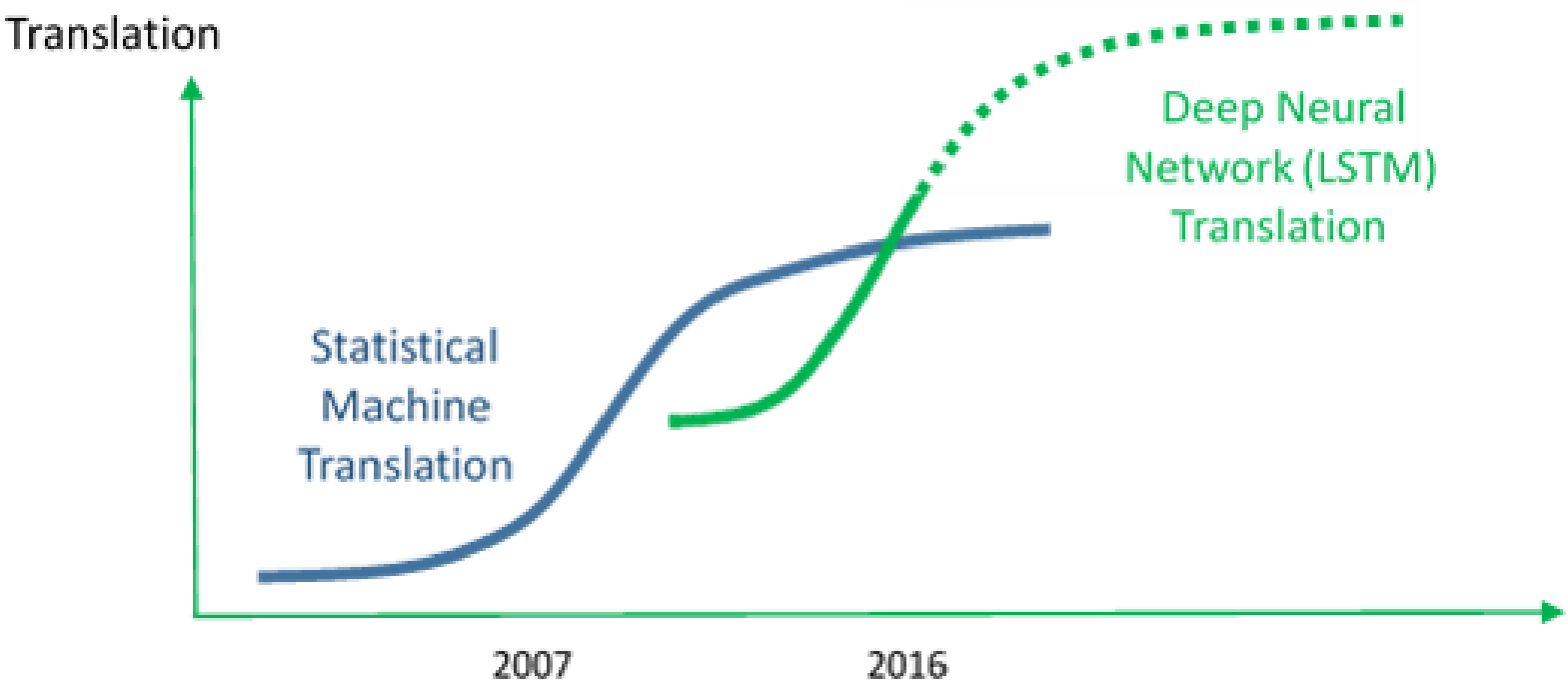
Attention Visualization



credits: NIPS slides, Richard Socher

Question answering

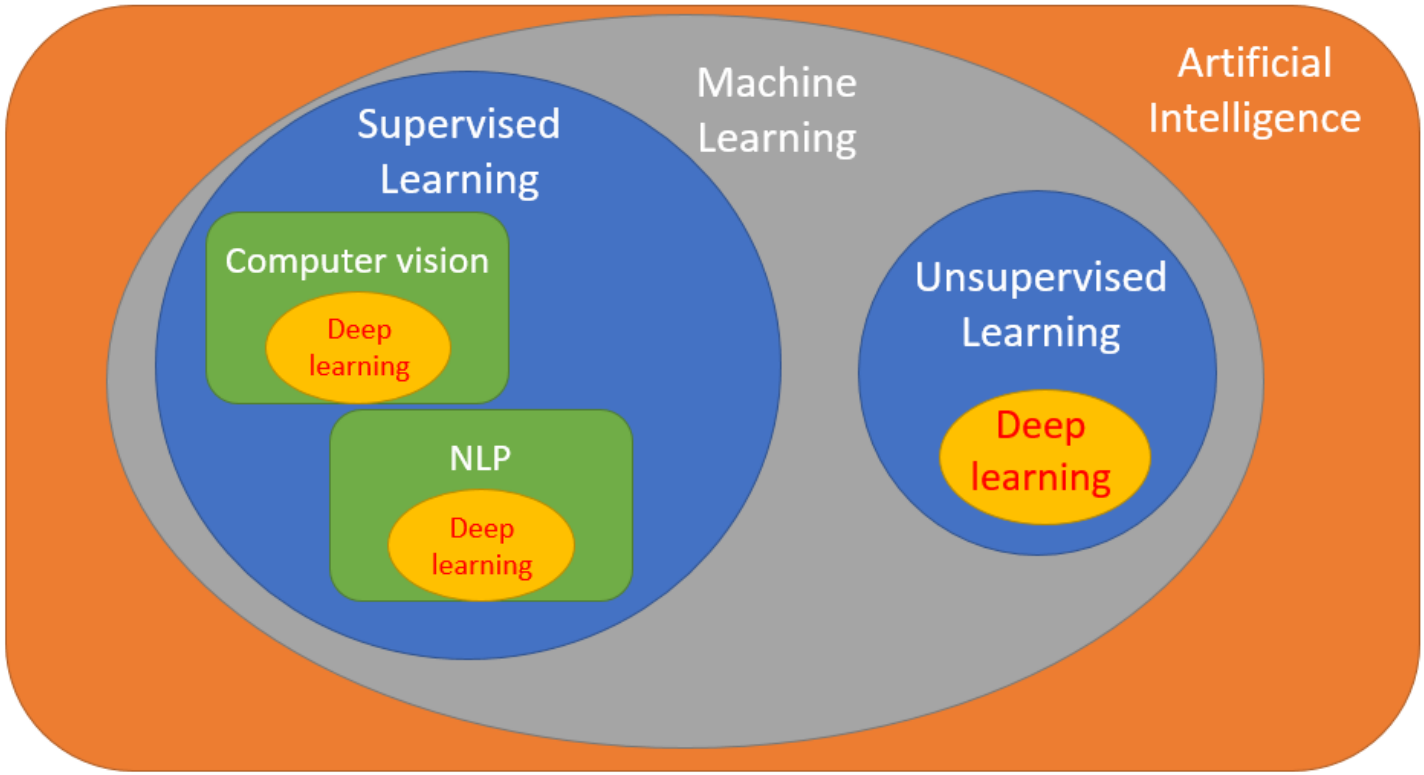
- Info: Mary Walked to the bathroom
- Info: Sandra went to the garden
- Info: Daniel went back to the garden
- Info: Sandra took the milk there
- Question: *Where is the milk?*
- Answer: **Garden**



credits: Microsoft research blog

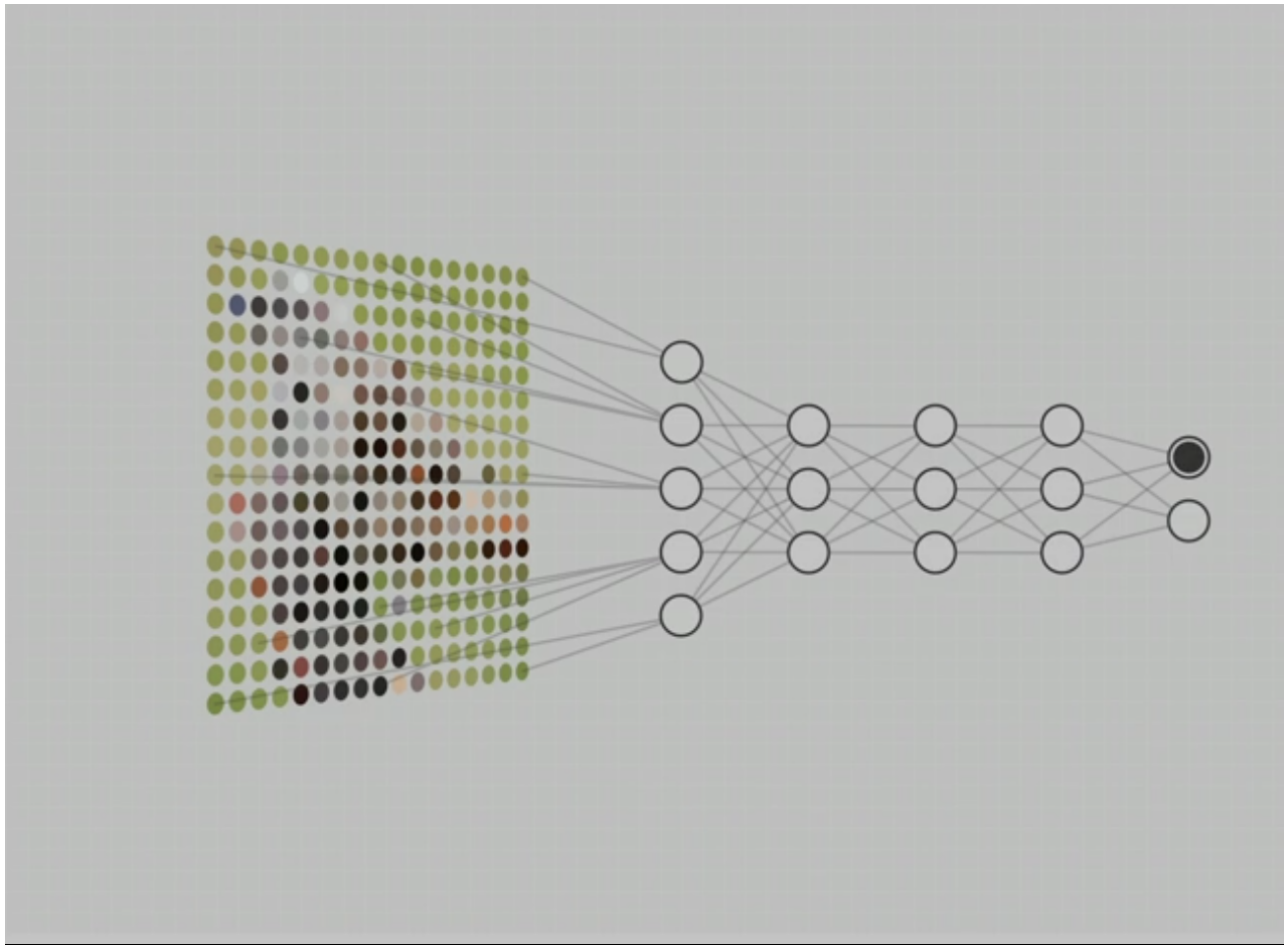
Input sentence:	*Translation (PBMT):*	*Translation (GNMT):*	*Translation (human):*
李克強此行將啟動中加總理年度對話機制，與加拿大總理杜魯多舉行兩國總理首次年度對話。	Li Keqiang premier added this line to start the annual dialogue mechanism with the Canadian Prime Minister Trudeau two prime ministers held its first annual session.	Li Keqiang will start the annual dialogue mechanism with Prime Minister Trudeau of Canada and hold the first annual dialogue between the two premiers.	Li Keqiang will initiate the annual dialogue mechanism between premiers of China and Canada during this visit, and hold the first annual dialogue with Premier Trudeau of Canada.

credits: Google research blog


Artificial Intelligence
Machine Learning
Supervised Learning
Computer vision
Deep learning
NLP
Deep learning
Unsupervised Learning
Deep learning

How do RNN's work?

Credits for images to Andrej Karpathy



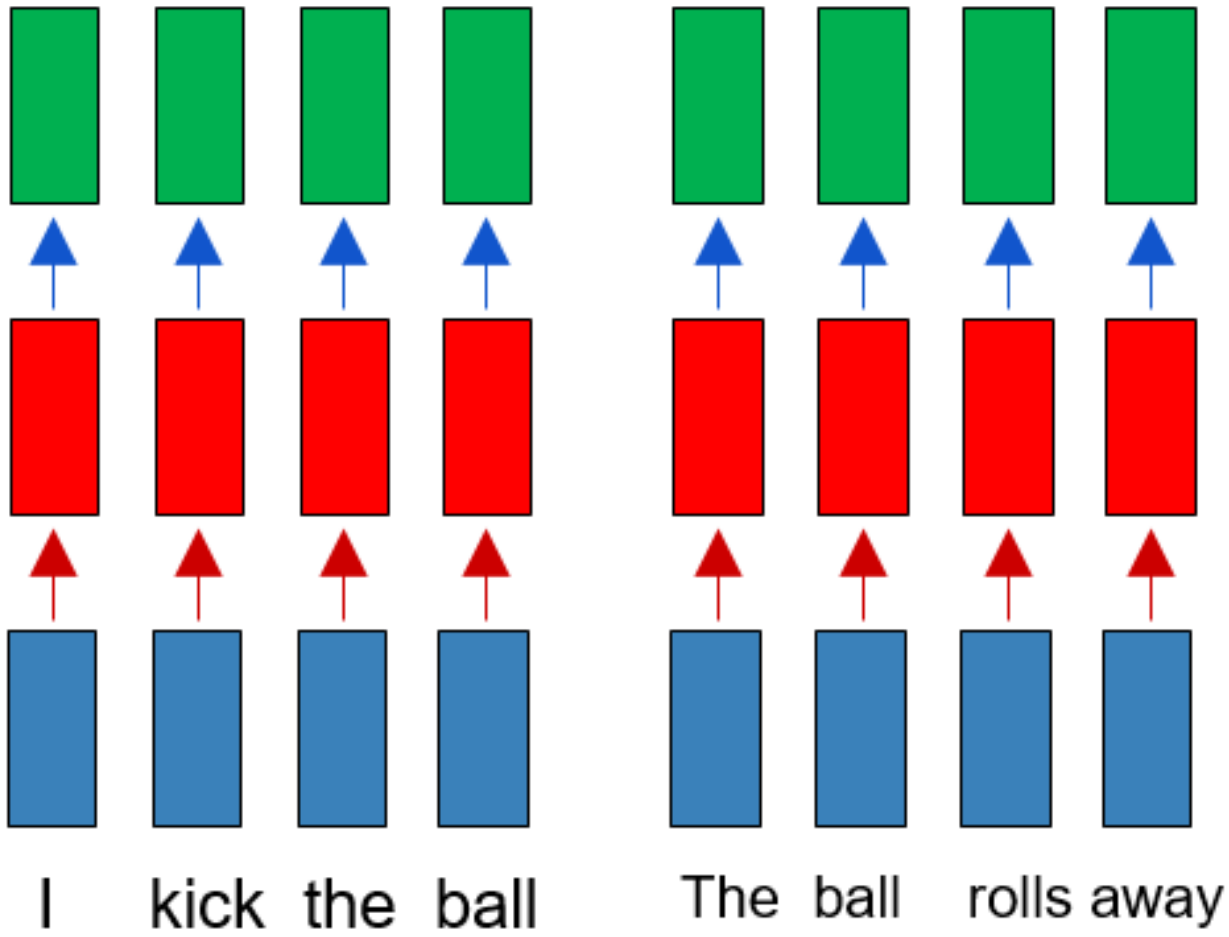
credits: Blaise Aguera y Arcas

Python code

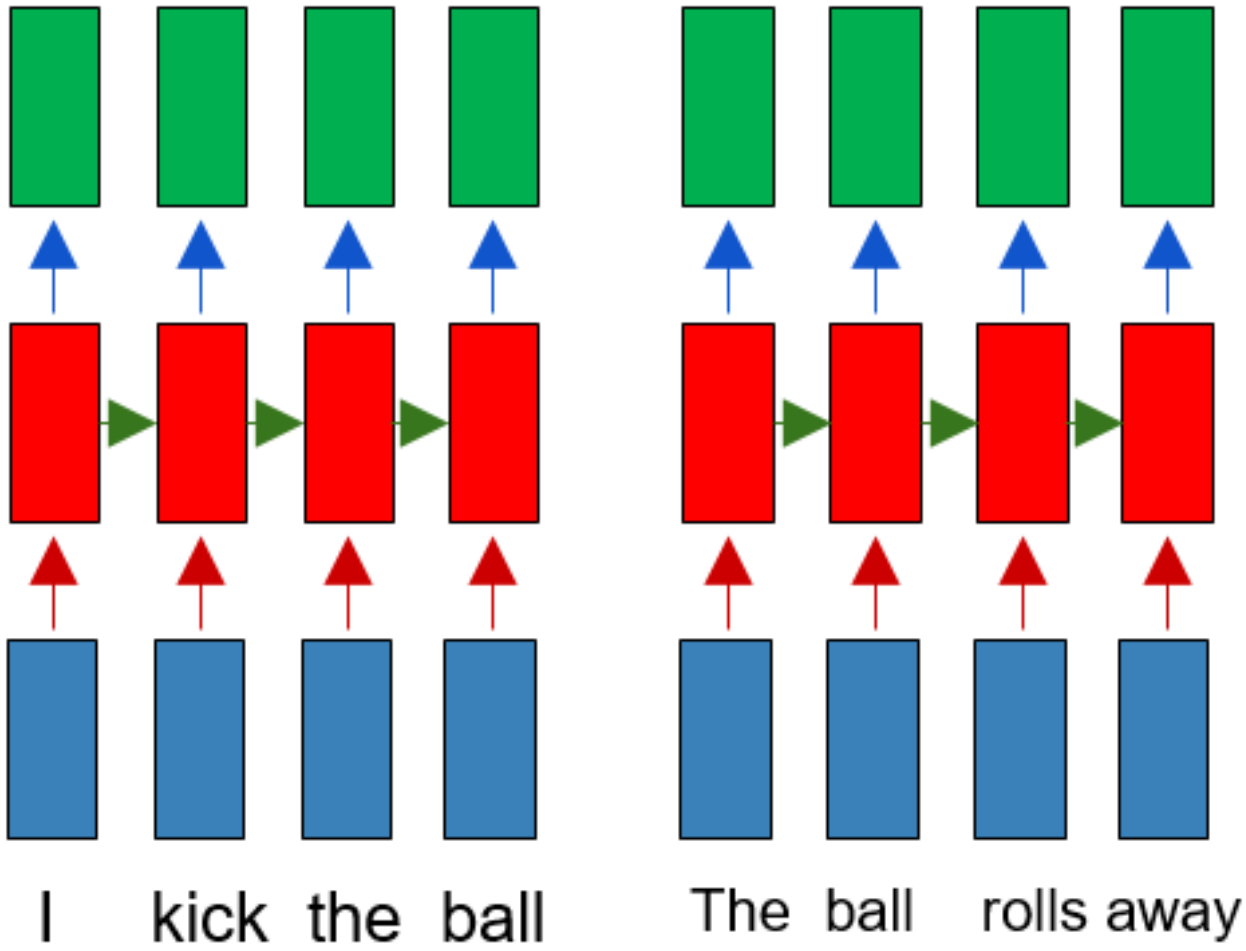

```
#FF
def step(x):
  h = np.tanh(np.dot(W1, x) + b1)
  y = np.dot(W2, h) + b2
  return y

y = step(x)
```

Word tagging



Connect over time



RNN cell

$$h = \phi(W_{xh}x) \qquad y = \phi(W_{hy}h)$$

$$h_t = \phi(W_{xh}x_t + W_{hh}h_{t-1}) \qquad y_t = \phi(W_{hy}h_t)$$

Python code


```
#FF
def step(x):
  h = np.tanh(np.dot(W1, x) + b1)
  y = np.dot(W2, h) + b2
  return y

y = step(x)
```

Python code

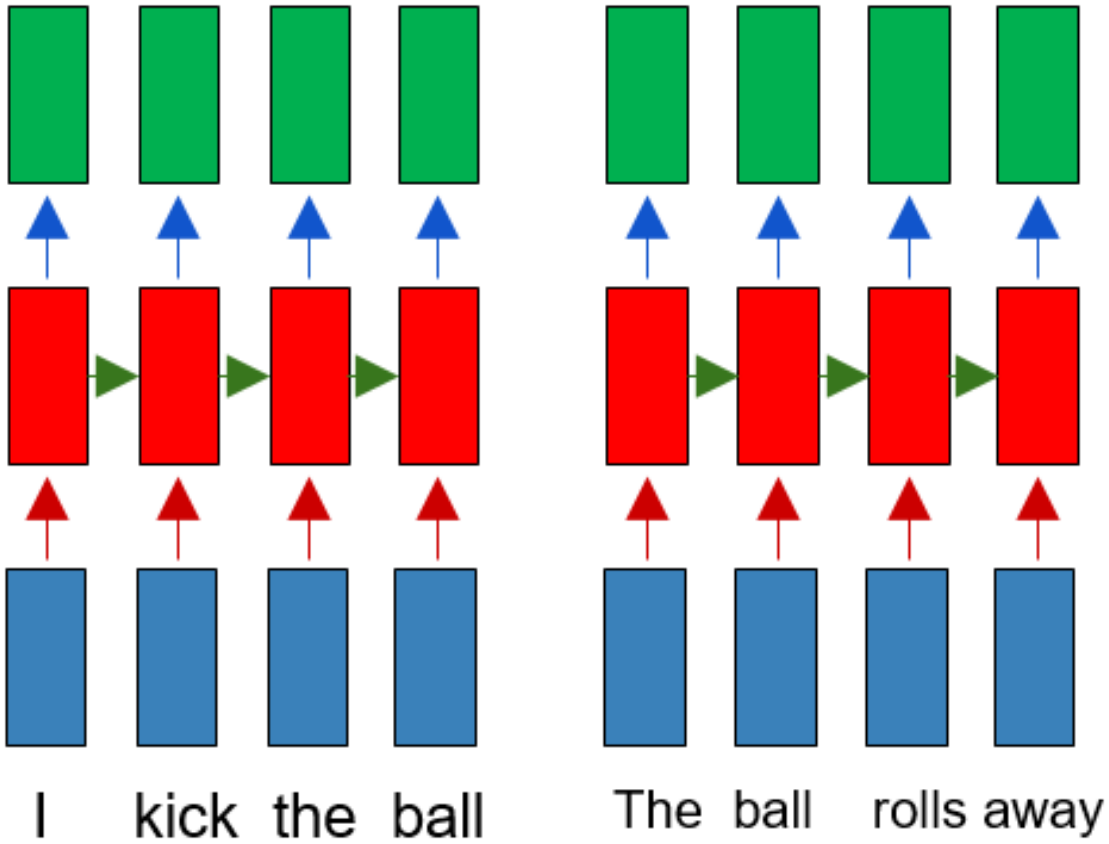

```
# RNN
rnn = RNN()
y = rnn.step(x)

class RNN:
  def step(self, x):
    self.h = np.tanh(np.dot(W1, self.h)
          + np.dot(W2,x))
    y = np.dot(W3, self.h)
    return y
```

Python code


```
#FF
def step(x):
  h = np.tanh(np.dot(W1, x) + b1)
  y = np.dot(W2, h) + b2
  return y
#RNN
class RNN:
  def step(self, x):
    self.h = np.tanh(np.dot(W1, self.h)
          + np.dot(W2,x))
    y = np.dot(W3, self.h)
    return y
```

Word tagging

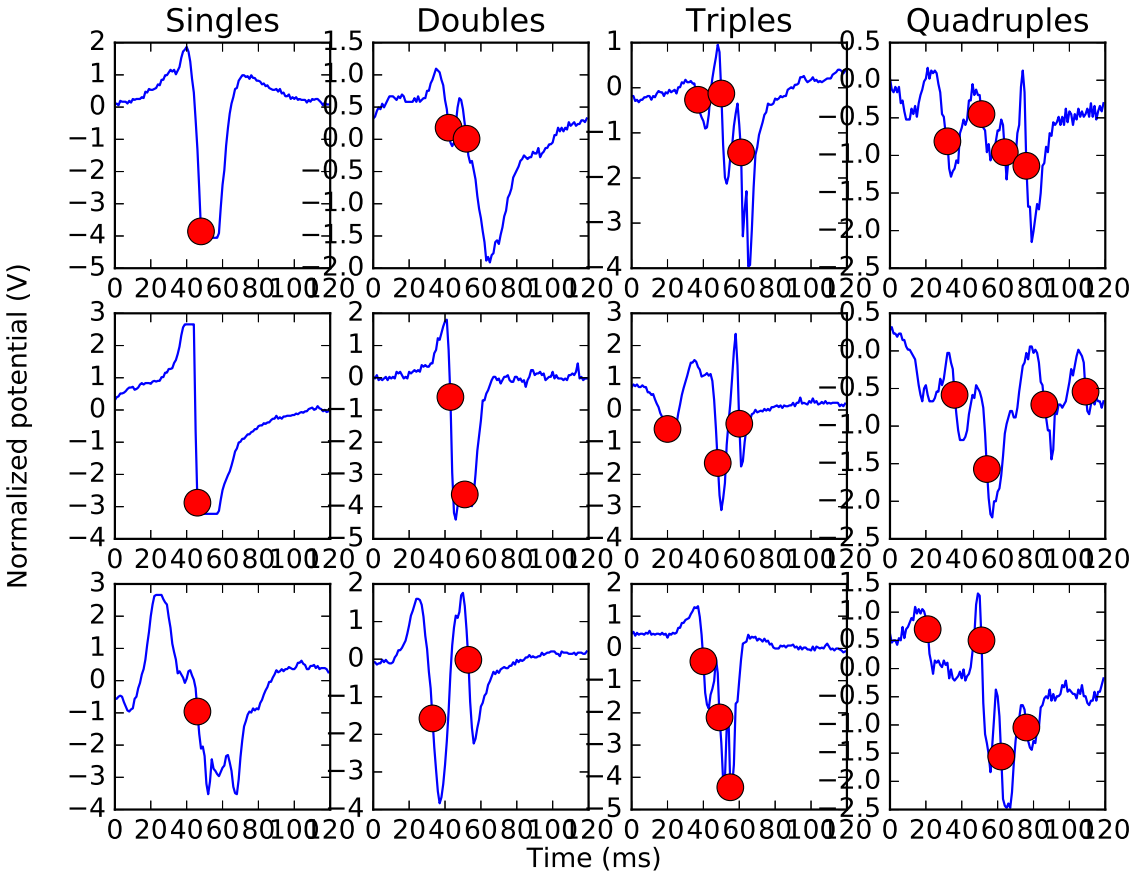


RNN in 112 lines

Examples

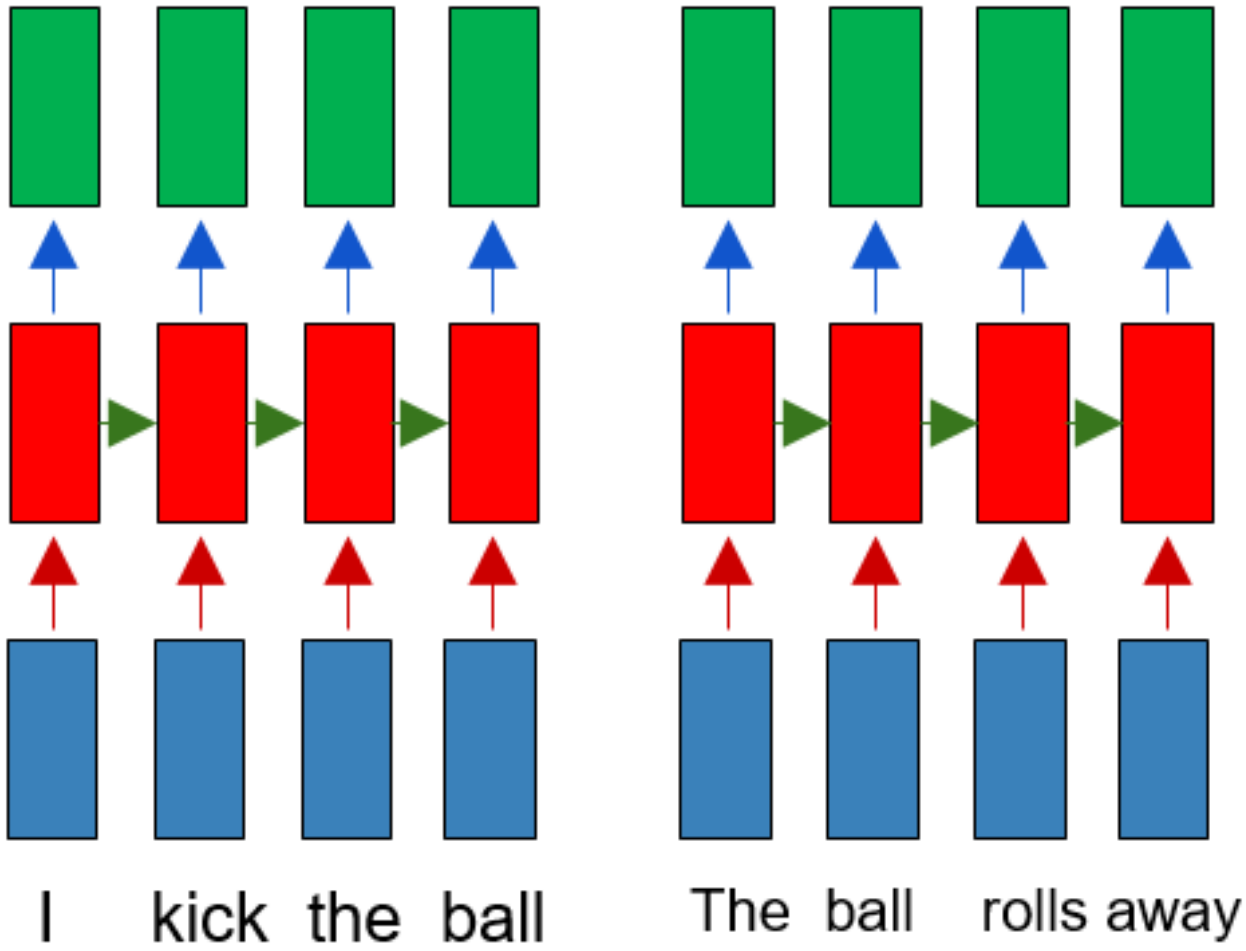
- Weather forecasting
- Detection of walk/run/stairs in smartphones
- Analyse stock markets and other financial markers
- Speech transcription

Tagging of clinical events



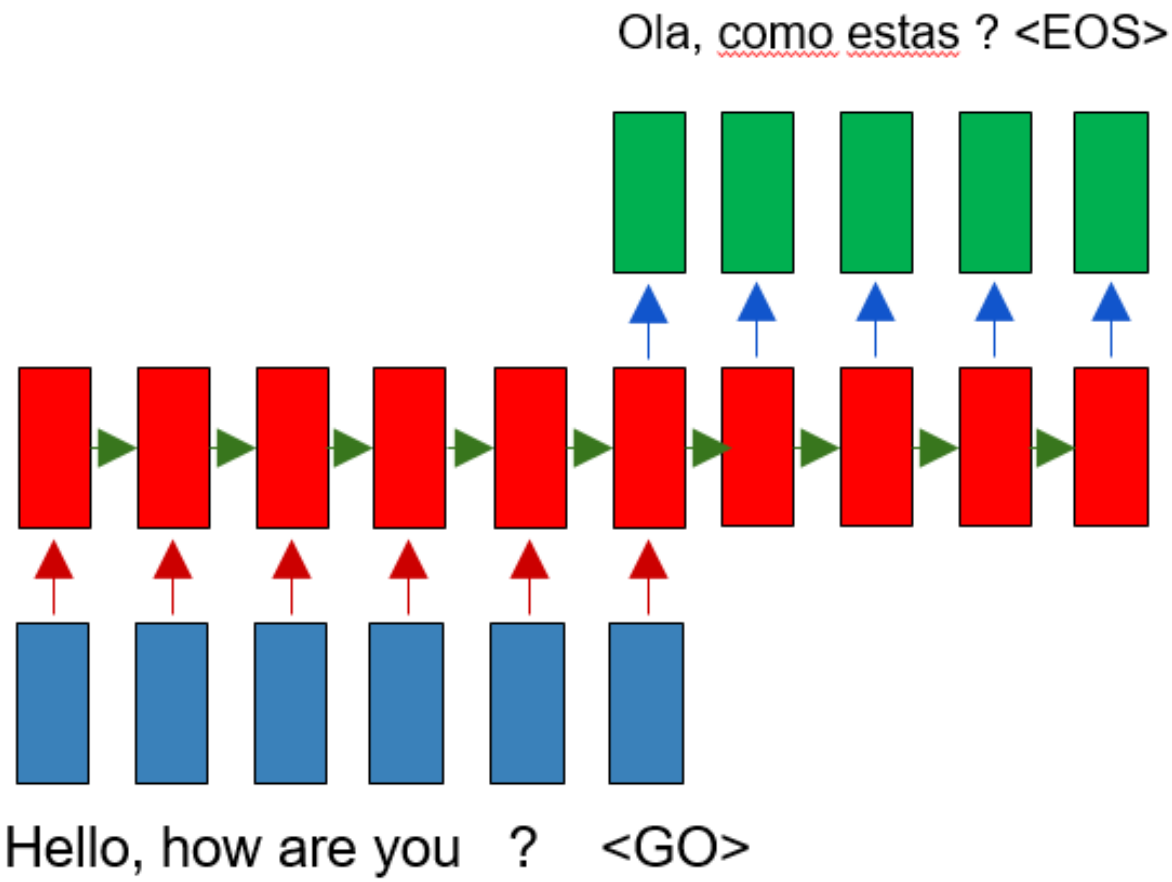
Classification of fractionated electrograms in epicardial mappings using a recurrent neural network, R Romijnders et al.

Word tagging

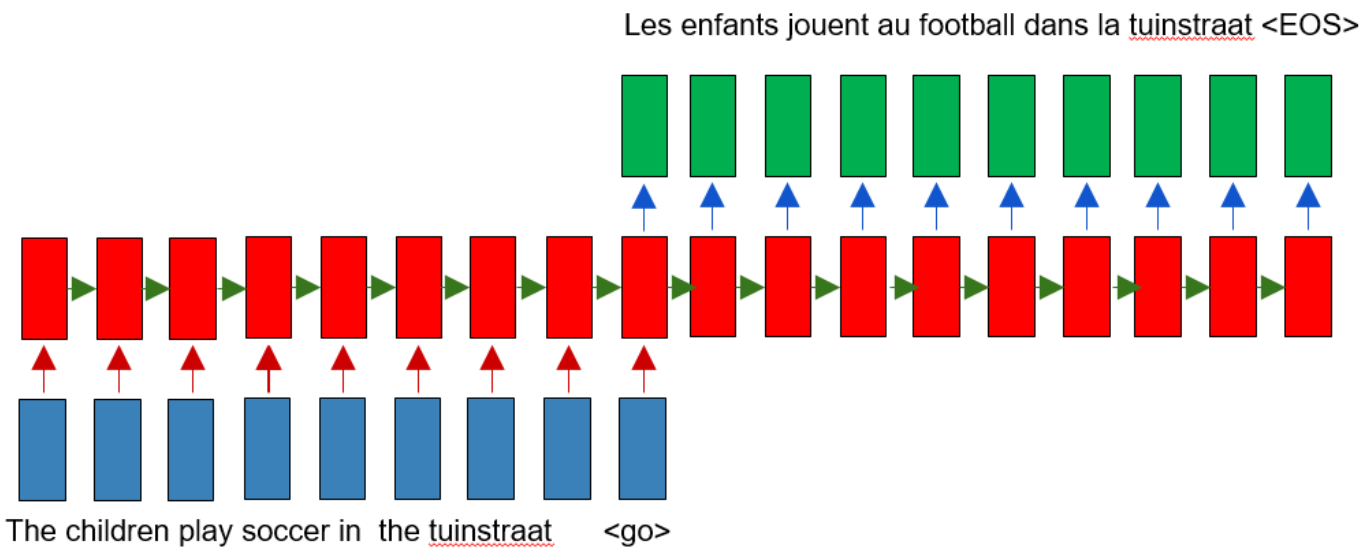


Seq2Seq

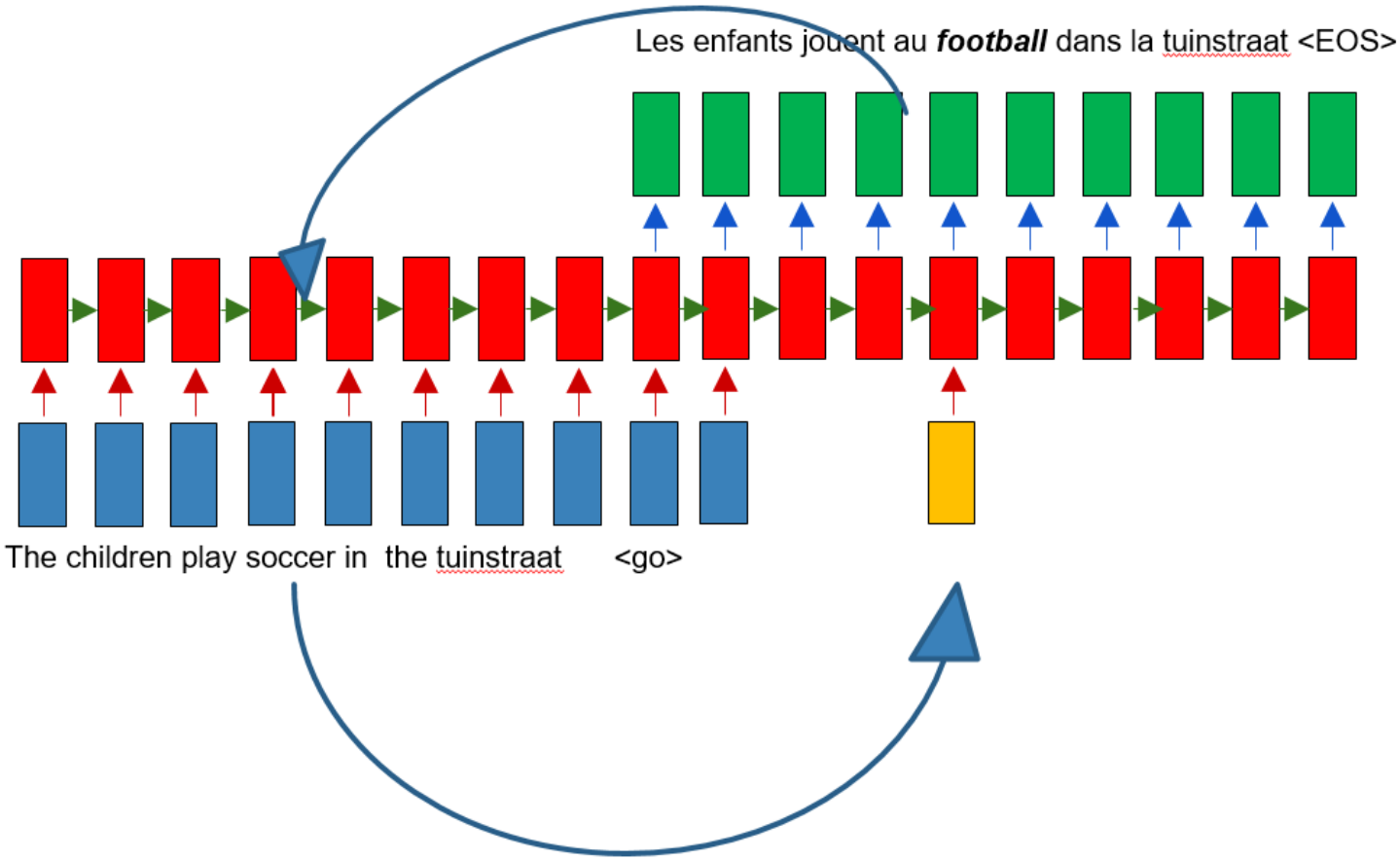
Sequence 2 sequence



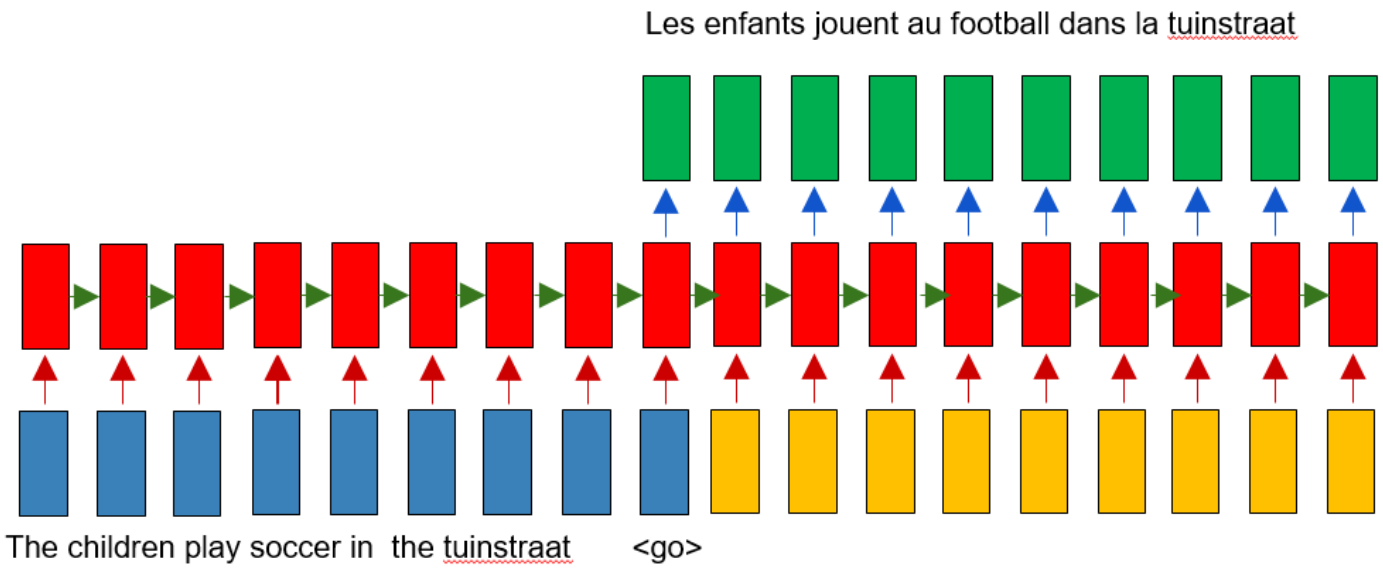
Long sequence 2 long sequence



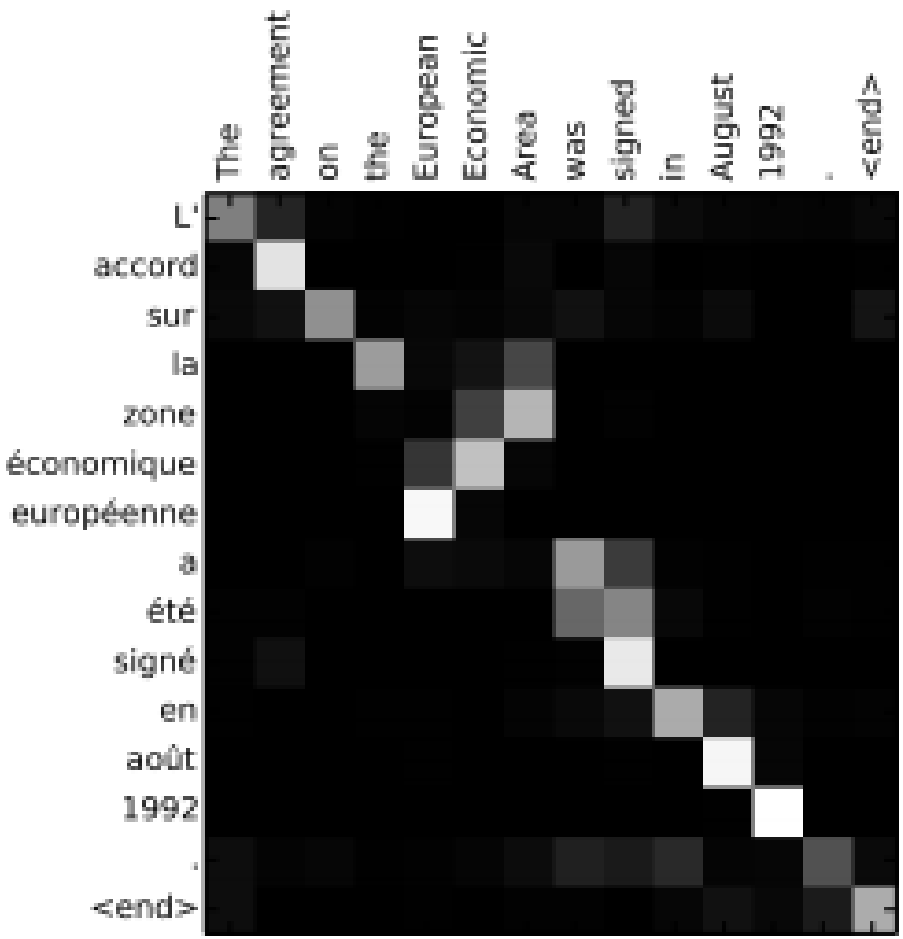
Long sequence 2 long sequence



Long sequence 2 long sequence



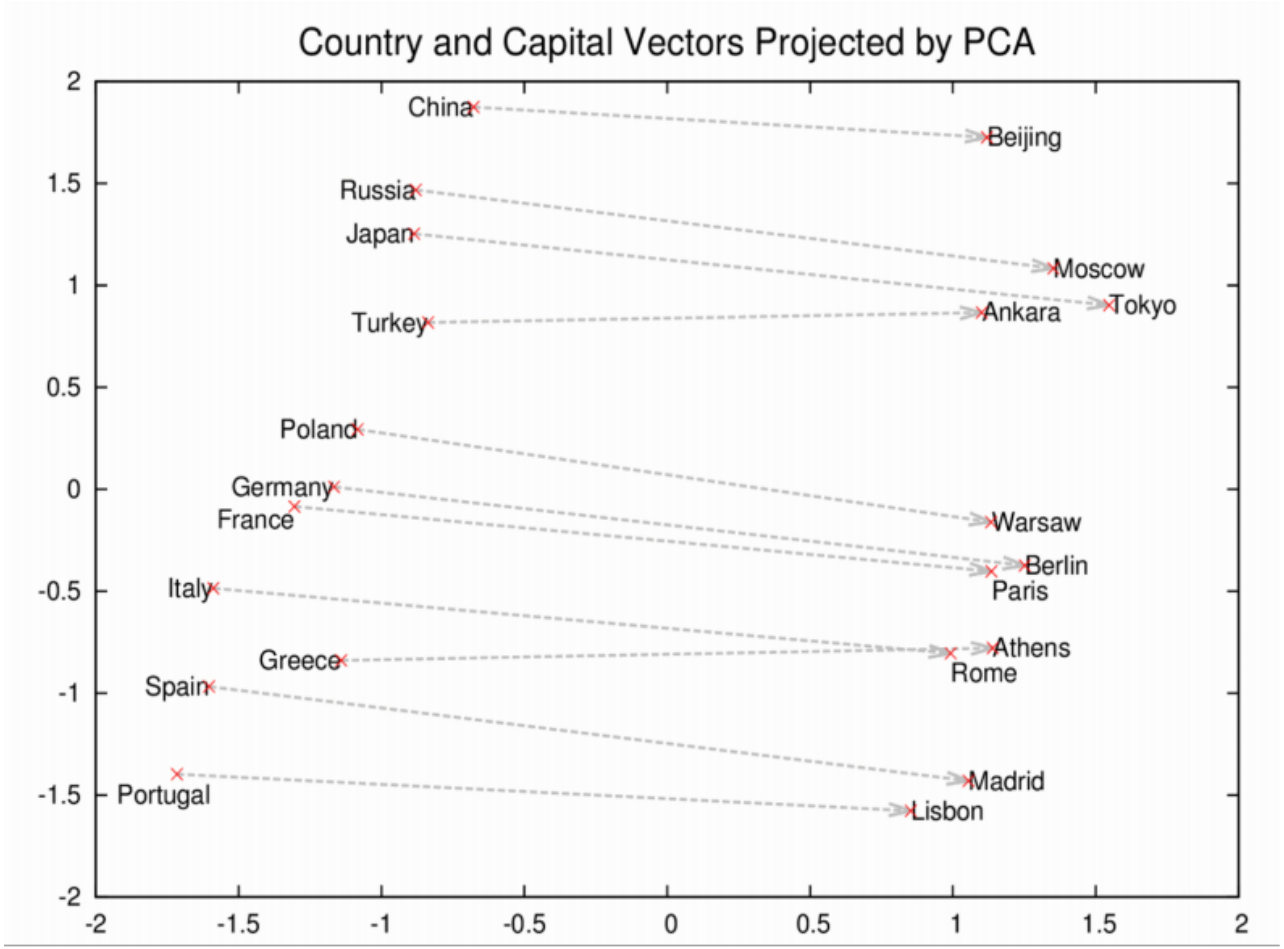
Visualize attention



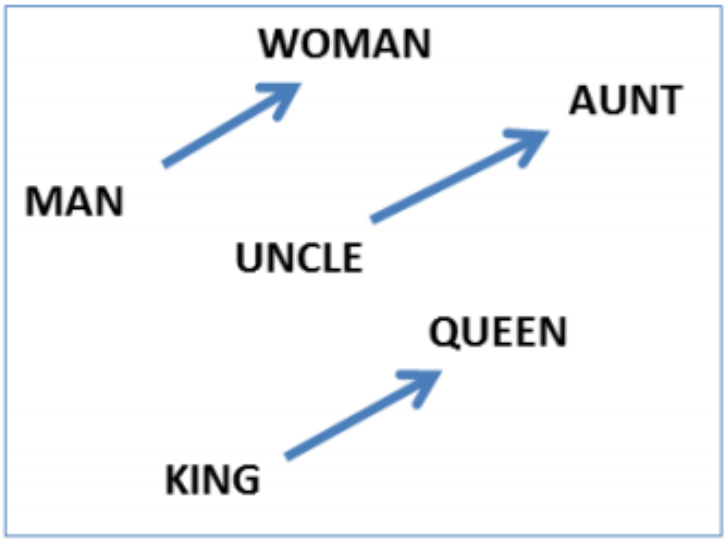
-

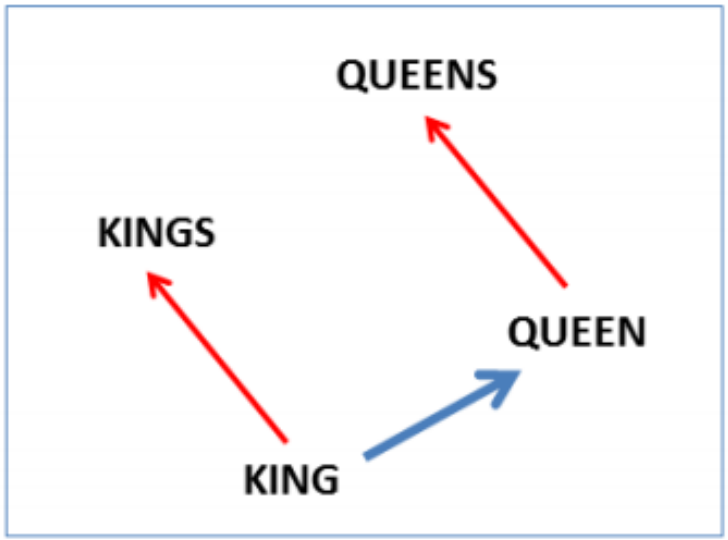
How to train

- Word vectors
- Training
  - Data parallelism
  - Model parallelism
- Fast translation



credits: deeplearning4j.org/word2vec

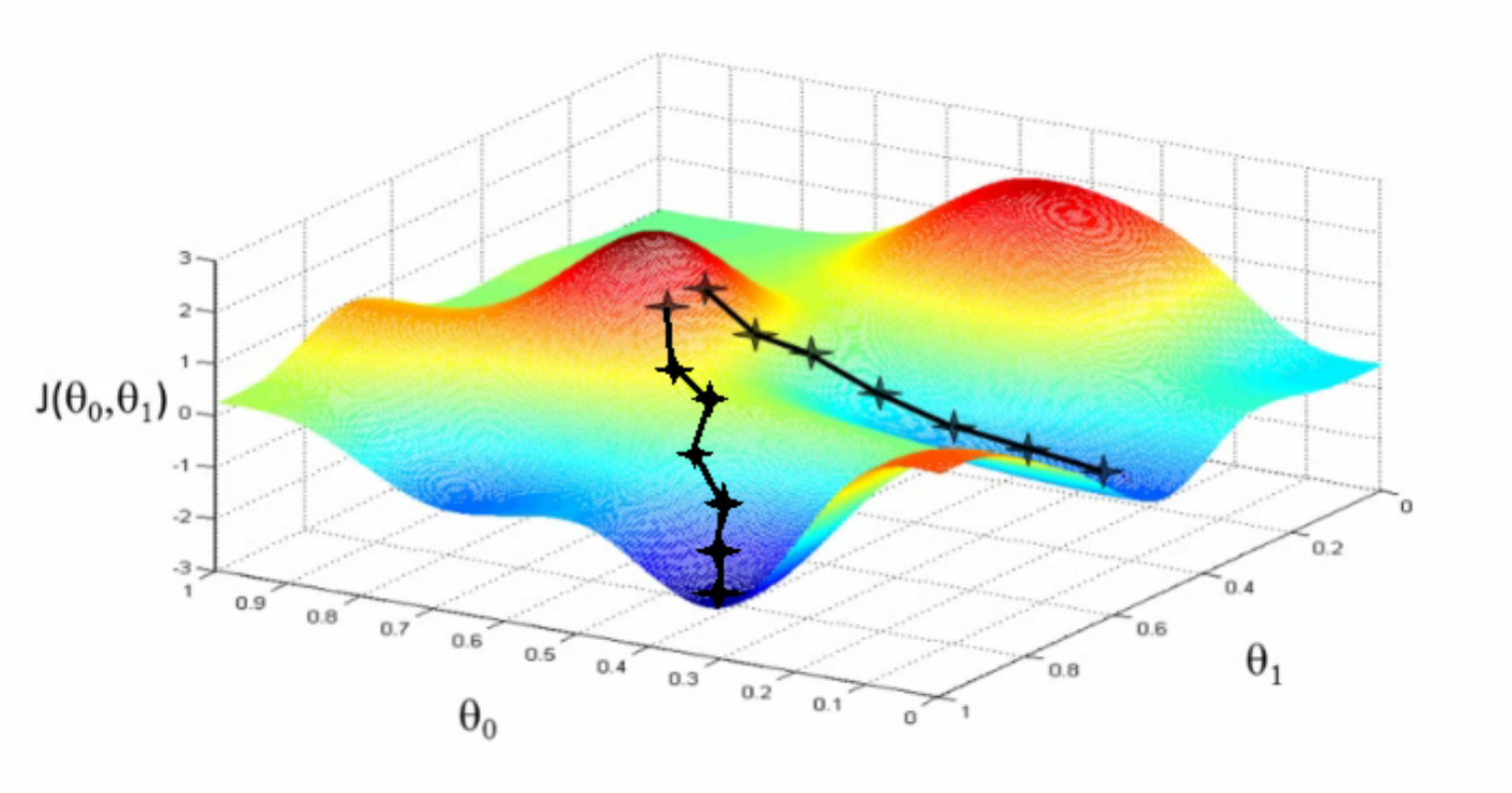




(Mikolov et al., NAACL HLT, 2013)

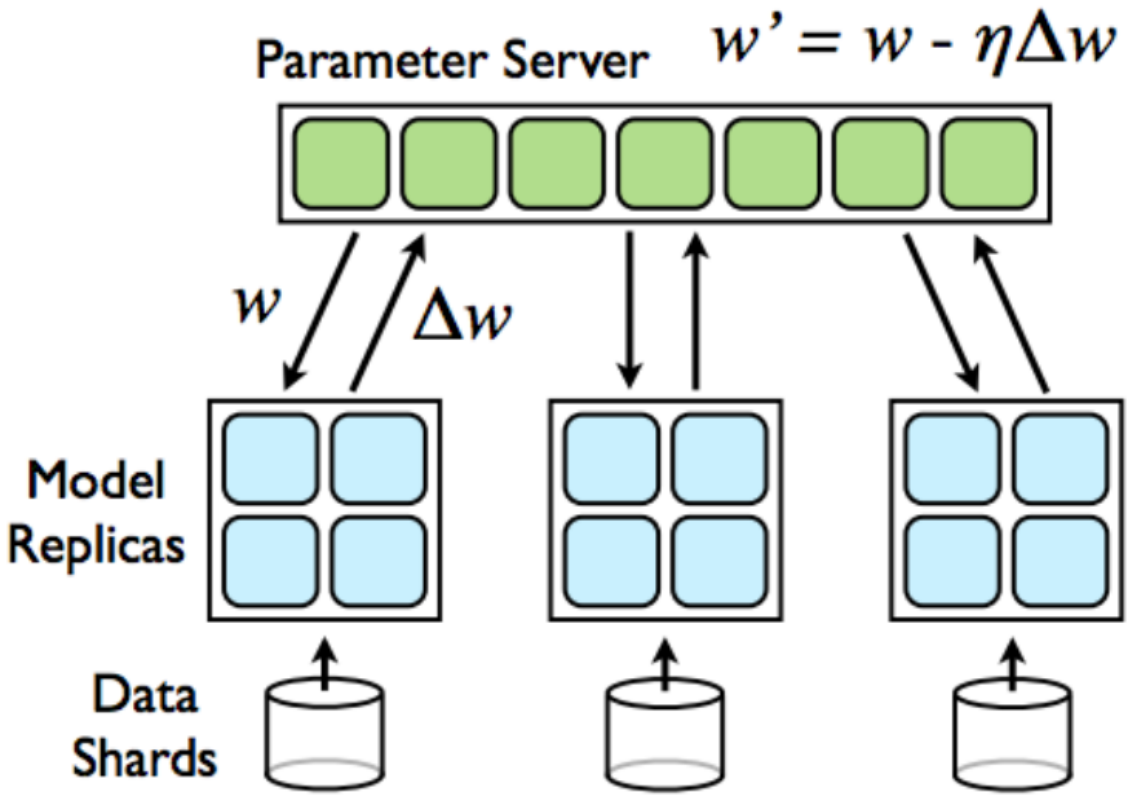
Python implementation

Gradient Descent



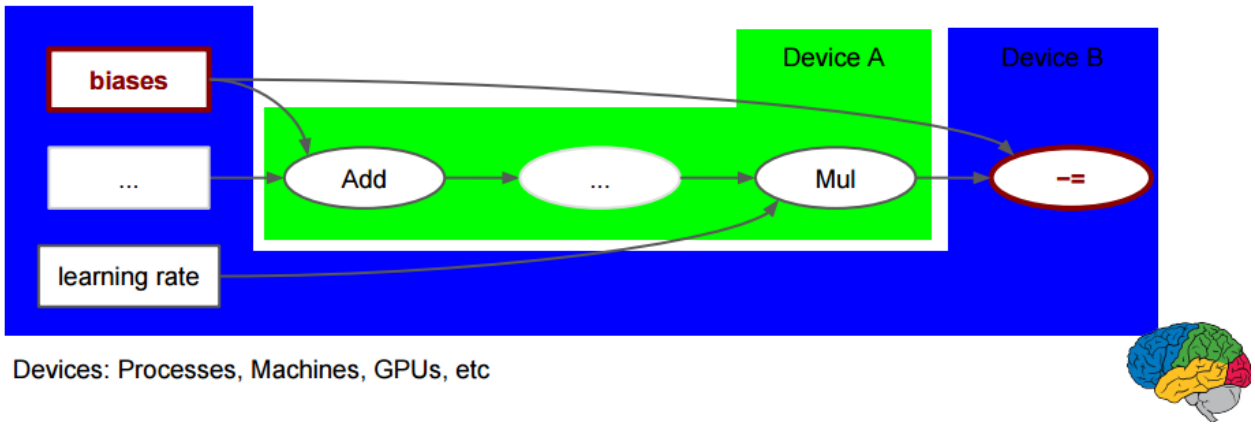
Credits: ML blog Vasilis Vryniotis

Data parallelism



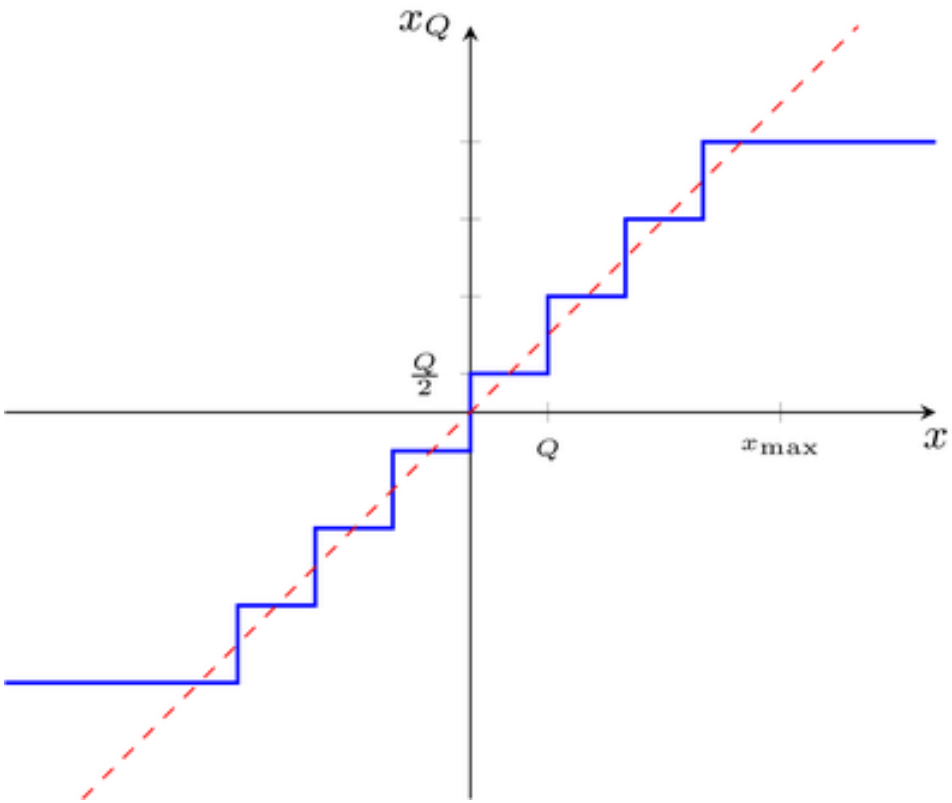
Credits: arimo.com/machine-learning/deep-learning/2016/arimo-distributed-tensorflow-on-spark/

Model parallelism



Credits: slideshare Jeff Dean

Quantization



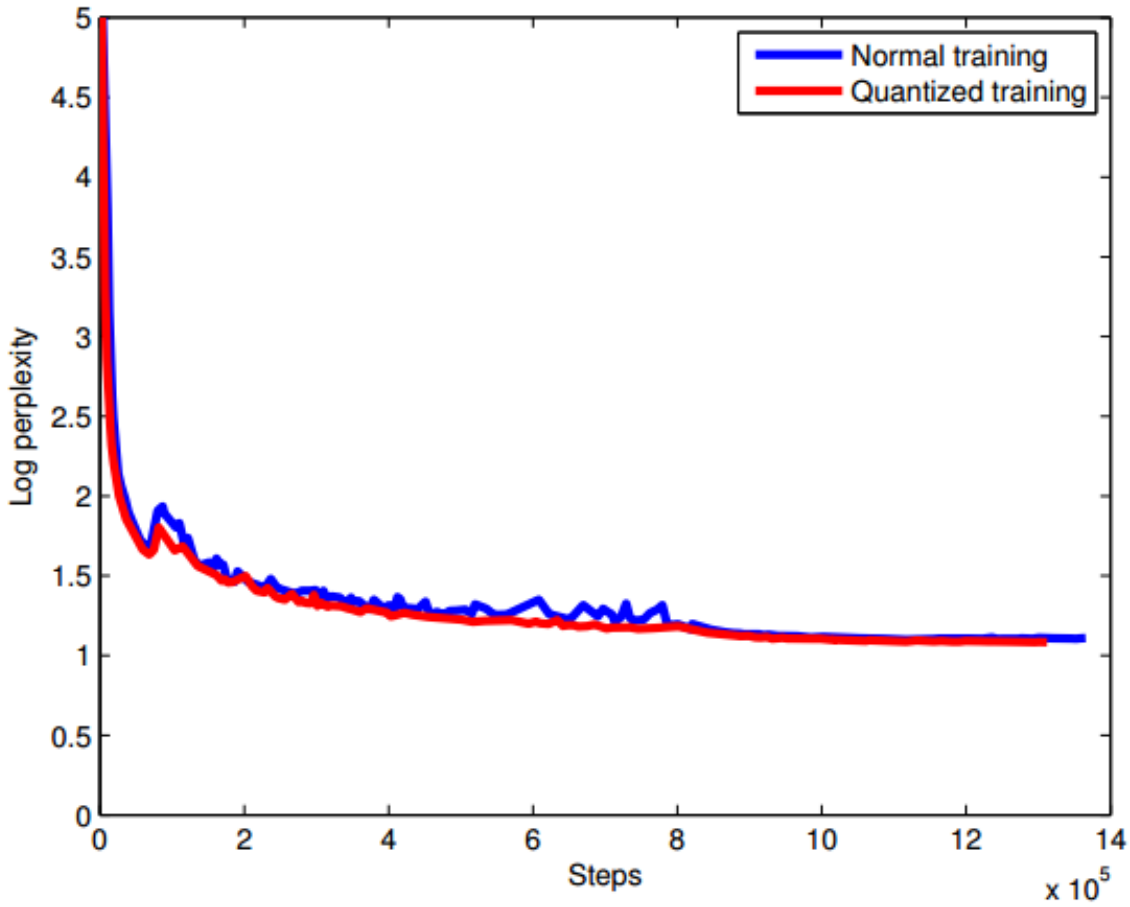
Credits: Sascha Spors

Speed up

	BLEU	Log Perplexity	Decoding time (s)
CPU	31.20	1.4553	1322
GPU	31.20	1.4553	3028
TPU	31.21	1.4626	384

Credits: Arxiv GNMT paper

Quantization



Credits: Arxiv GNMT paper

Conclusion

Further reading

1. GNMT paper and the blog
2. The Unreasonable Effectiveness of Recurrent Neural Networks, Andrej Karpathy
3. Supervised sequence labelling with RNN, Alex Graves
4. robromijnders.github.io
5. Courses
    1. Oxford Deep NLP
    2. Stanford DL for NLP