Neural Machine Translation

Arul Selvam Periyasamy

June 13, 2017

Rheinische Friedrich-Wilhelms-Universität Bonn
Seminar: Natural Language Processing

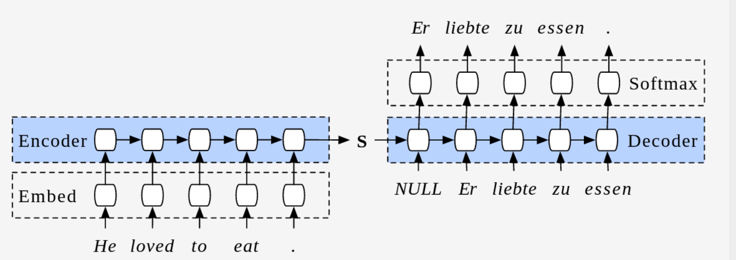
Machine Translation

In a probabilistic perspective, machine translation can be formulated the problem of finding a target sentence y that maximizes the conditional probability of y from a given source sentence x.

$$arg\, max_y \ p(x|y)$$

In Neural Machine Translation, a parameterized model (a neural network) is trained to maximize the conditional probability of the sentence pairs given parallel training corpus.

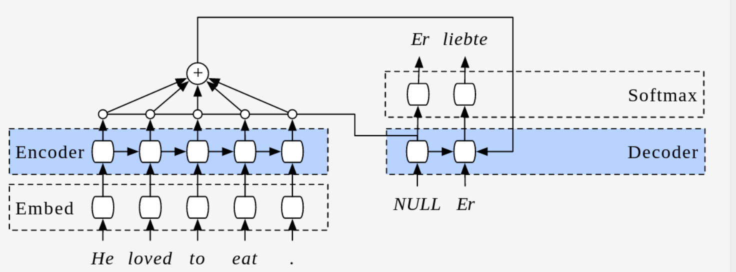
NMT - A historic perspective



Encoder-Decoder model for Machine Translation

- Fixed size encodings.
- Each language typically required an Encoder and Decoder.

Jointly learning to Align and translate



Encoder-Decoder model with context

Jointly learning to Align and translate

For a input sentence, $X = (x_1, \cdots, x_{T_x})$. The NMT[1] system consists of,

- Encoder and Decoder are multi-layer recurrent neural networks (RNNs).
- Encoder RNN, at each input step t, generates hidden state, $h_t = f(x_t, h_{t-1})$.
- Context vector encodes the input sequence as, $c = q(\{h_1, \cdots, h_{T_x}\})$.

[1] **NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE. Dzmitry Bahdanau**, **KyungHyun Cho**, **Yoshua Bengio**, **ICLR**, **2015**.

Jointly learning to Align and translate

- The decoder is trained to predict the next work y_t given the context vector c and all previously predicted words $\{y_1, \cdots, y_{t-1}\}$

$$p(y) = \prod_{t=1}^{T} p(y_t | \{y_1, \cdots, y_{t-1}\}, c)$$

- With RNN, each conditional probability is modeled as,

$$p(y_t | \{y_1, \cdots, y_{t-1}\}, c) = g(y_{t-1}, s_t, c)$$

where s_t is the hidden state of the RNN.

Context Vector

The context vector for a input sentence i, is computed as a weighted sum of hidden states of the encoder (also known as **annotations**)

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$$

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{T_x} exp(e_{ik})}$$

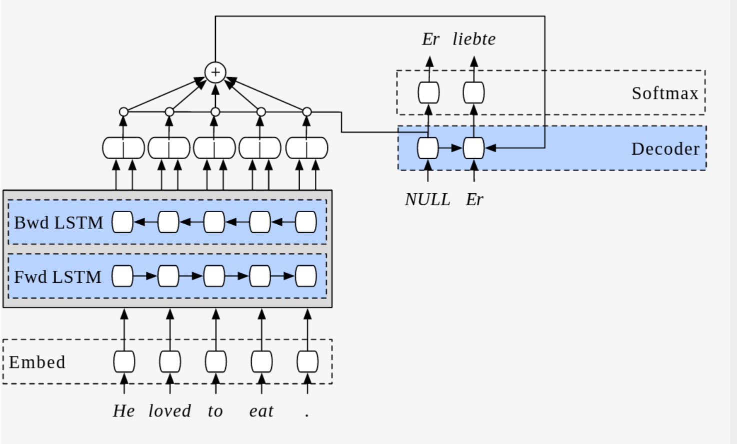
where,

$$e_{ij} = a(s_{i-1}, h_j)$$

is the alignment model that scores how well the inputs around the j and the output at the position i match.

A feedforward neural network is used as the alignment model and is **jointly trained** with all the NMT system as a whole.

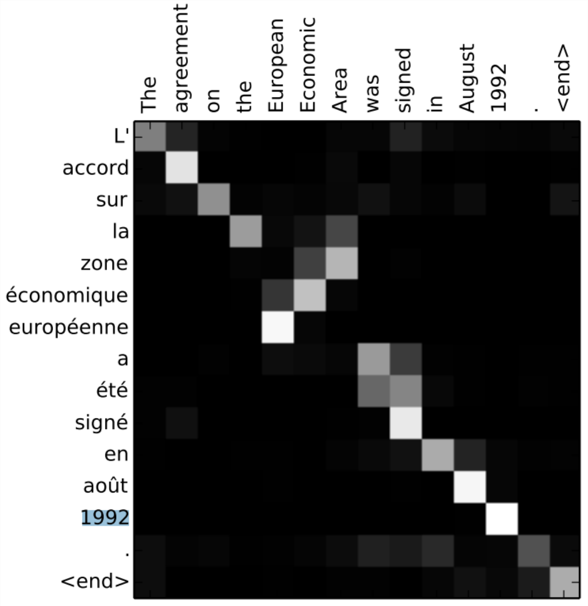
Bi-directional Encoder



Bi-directional Encoder

- Recurrent connection in both directions.
- Two independent states, updated independently.

Visualization of the context



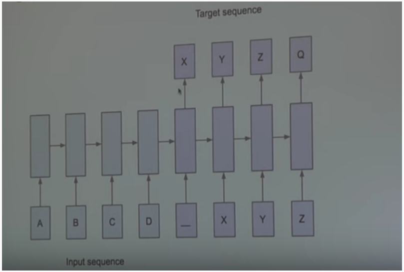
Seq2Seq Learning

Machine Translation is can treated as a special case of a more generic sequence to sequence modeling.

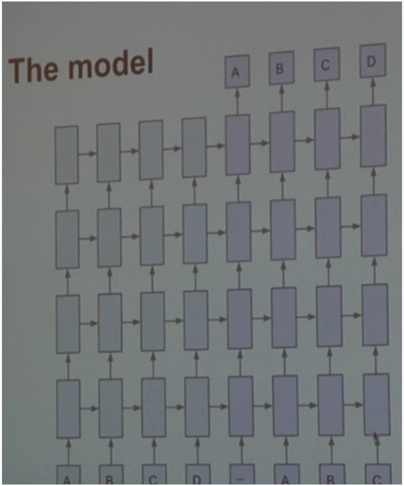
1. Idea is simple: Throw more power at the network.
2. Deep LSTM layers.
3. No special handling for Machine translation.
4. Trained with SGD.

$$p(y_1, \cdots, y_T | x_1, \cdots, x_T) = \prod_{t=1}^{T} p(y_t | v, y_1, \cdots, y_{t-1})$$

Seq2Seq Learning

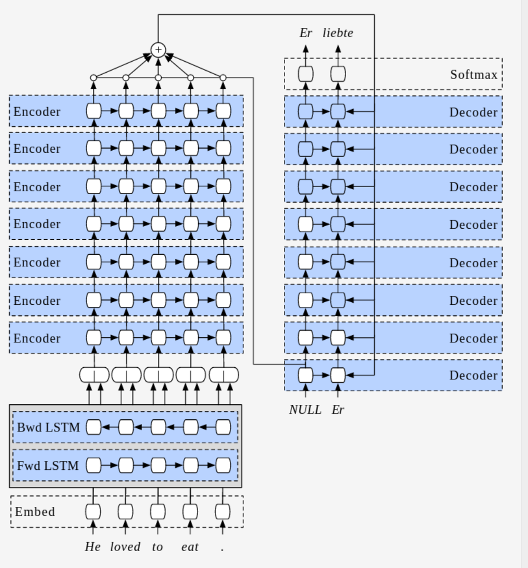


(a) Seq2Seq model



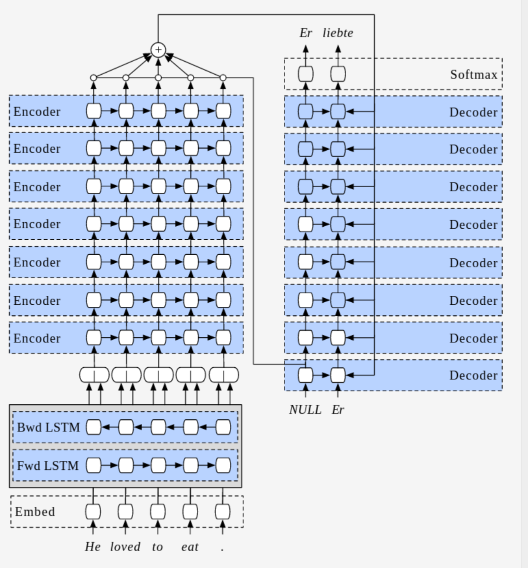
(b) More powerful model

Google NMT



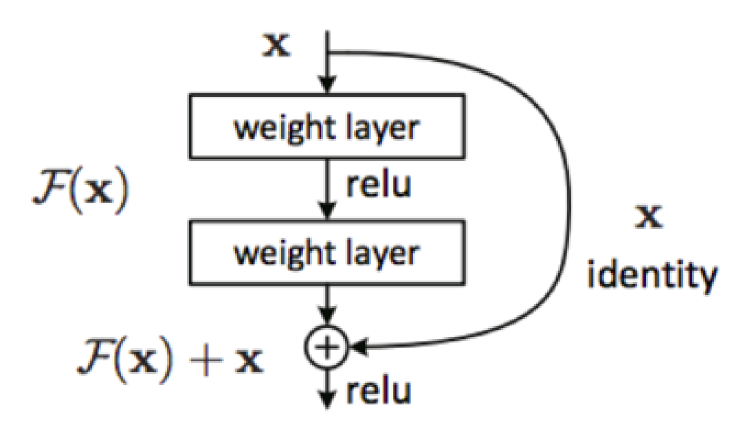
Simple Encoder-Decoder but more deeper as in Seq2Seq, and Context

Google NMT



Simple Encoder-Decoder but more deeper as in Seq2Seq and Context

Residual learning

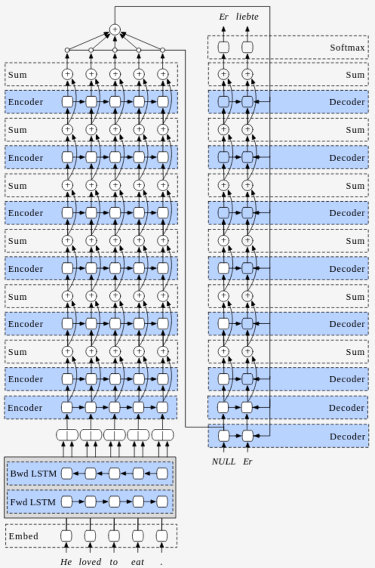


Residual networks

Residual connections enables training of very deep networks.[2]

[2] Deep Residual Learning for Image Recognition. Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. CVPR, 2016.

Google NMT with Residual connection



GNMT with residual connections.

Conclusion

Neural Machine Translation systems,

- Are State of the Art in Machine translation.
- Greatly benefited from the Neural Network research by other communities.
- In production by companies like Google, Microsoft, Facebook, etc.