SRP 2022:
Lecture 6-8 Summary

by Xinyu

Overview

- Language Models
- Recurrent Neural Network (RNN)
- Long short-term memory (LSTM) and Gated Recurrent Units (GRU)
- Machine Translation
- Seq2seq
- Attention

Language Model

- compute the probability of occurrence of a number of words in a particular sequence

$$P(w_1, \ldots, w_m) = \prod_{i=1}^{i=m} P(w_i | w_1, \ldots, w_{i-1}) \approx \prod_{i=1}^{i=m} P(w_i | w_{i-n}, \ldots, w_{i-1})$$

- n-gram model: statistical learning
    - sparsity problem – smoothing
    - storage problems
- window-based Neural LM

Language Model

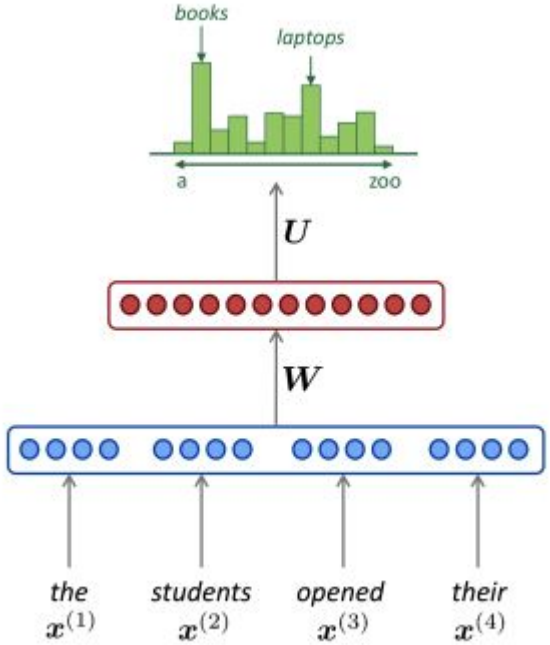
- compute the probability of occurrence of a number of words in a particular sequence

- n-gram model: statistical learning

    - sparsity problem – smoothing

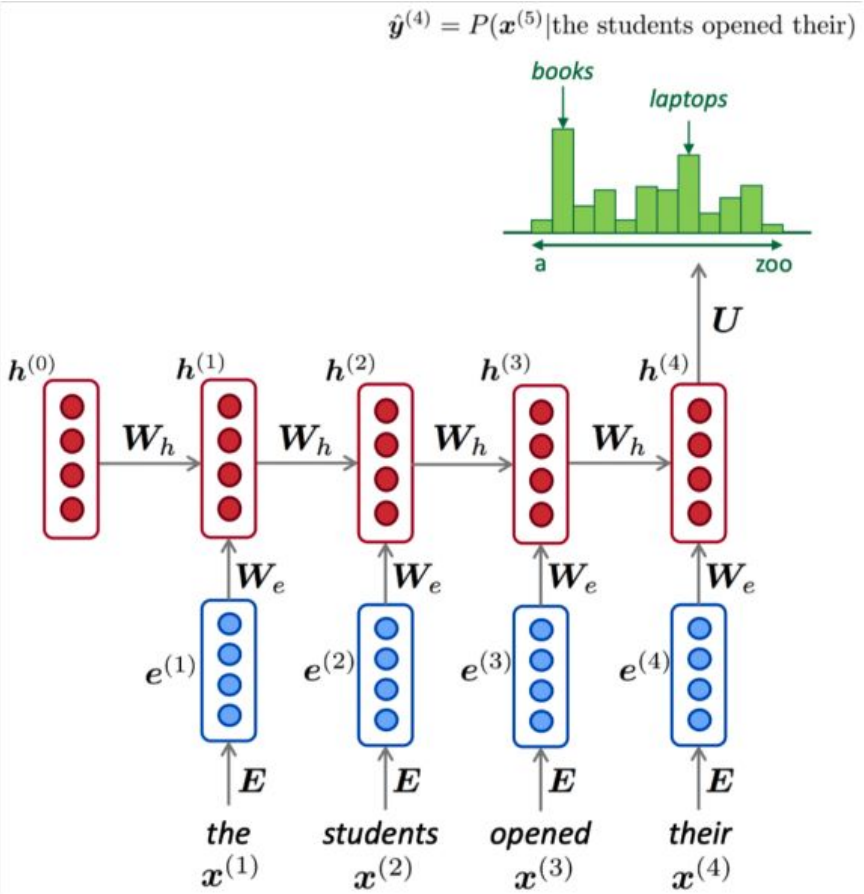
    - storage problems

- window-based Neural LM



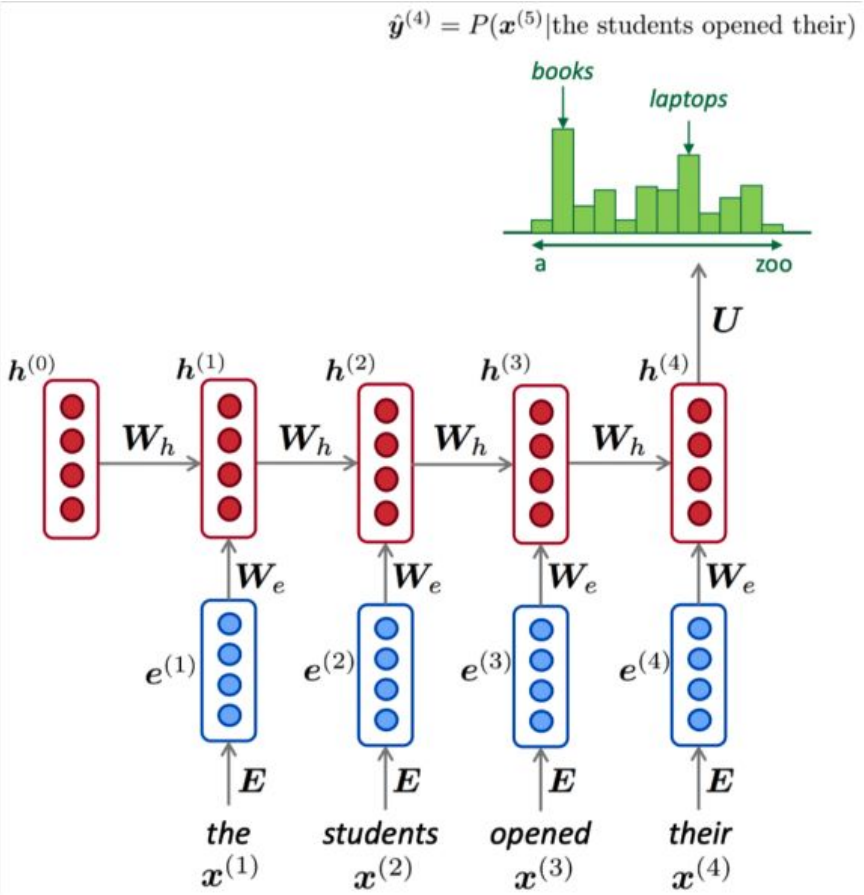
Recurrent Neural Network

- same weight matrix
- training - average CE loss
- metric: perplexity - exponential of the CE



Recurrent Neural Network

- pros
    - variable input length
    - fixed model size
    - keep information from many steps back
- cons
    - non-parallelized
    - vanishing / exploding gradient

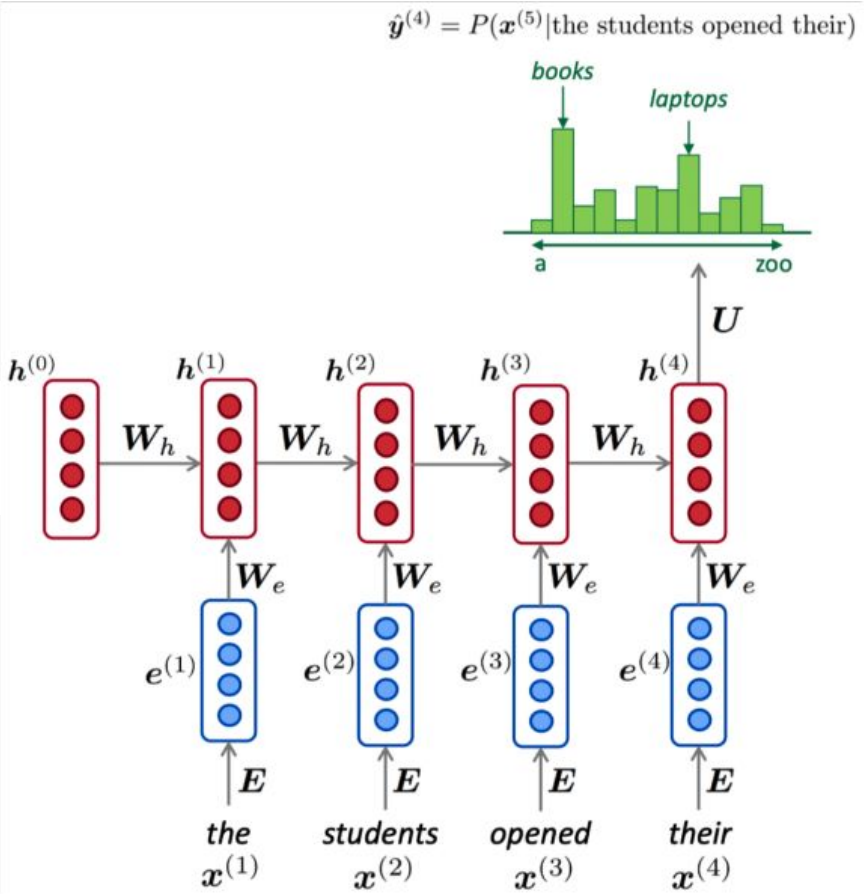


Recurrent Neural Network

Exploding / vanishing gradient

$$\boldsymbol{h}^{(t)} = \sigma\left(\boldsymbol{W}_h \boldsymbol{h}^{(t-1)} + \boldsymbol{W}_x \boldsymbol{x}^{(t)} + \boldsymbol{b}_1\right)$$

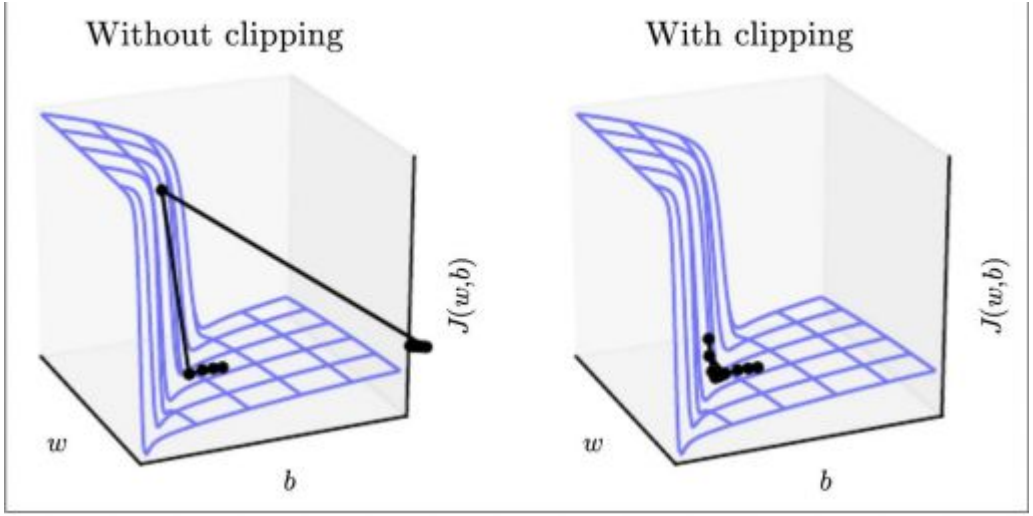
$$\frac{\partial \boldsymbol{h}^{(t)}}{\partial \boldsymbol{h}^{(t-1)}} = \operatorname{diag}\left(\sigma'\left(\boldsymbol{W}_h \boldsymbol{h}^{(t-1)} + \boldsymbol{W}_x \boldsymbol{x}^{(t)} + \boldsymbol{b}_1\right)\right) \boldsymbol{W}_h$$



Recurrent Neural Network

Exploding gradient

Gradient clip



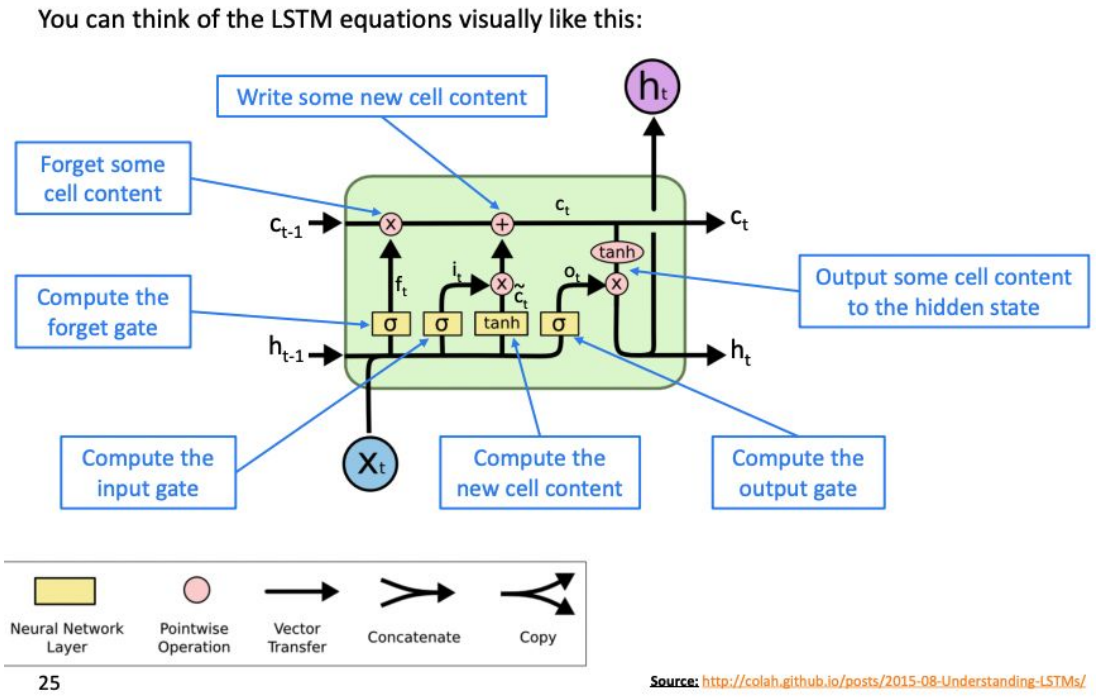
LSTM and GRU

- additional cell state for long-term information

- forget / input / output gate

- simplify above to have GRU

- without additional cell state



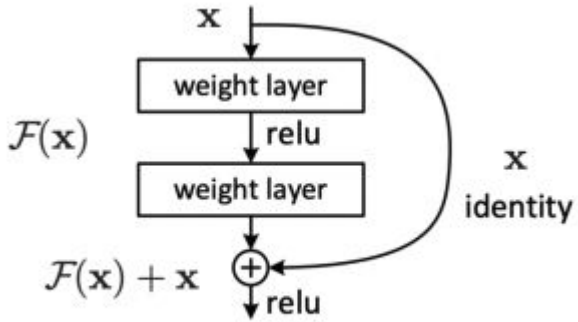


Figure 2. Residual learning: a building block.

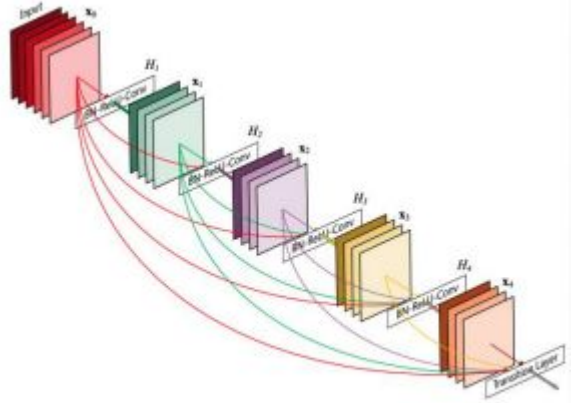
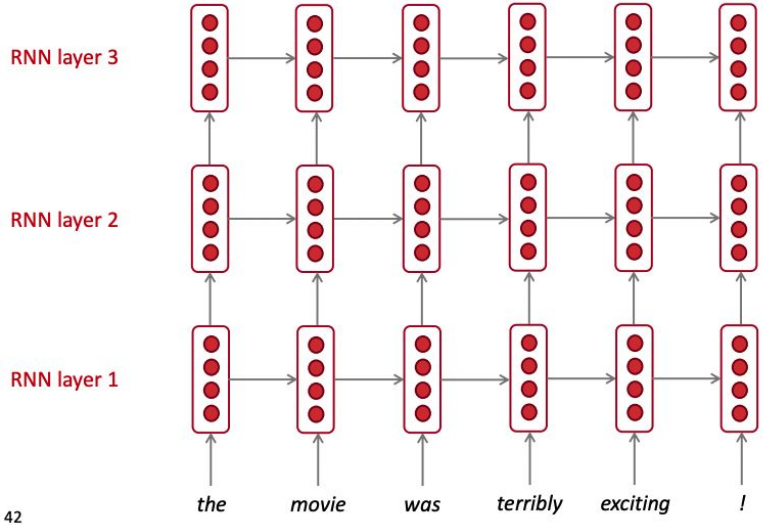
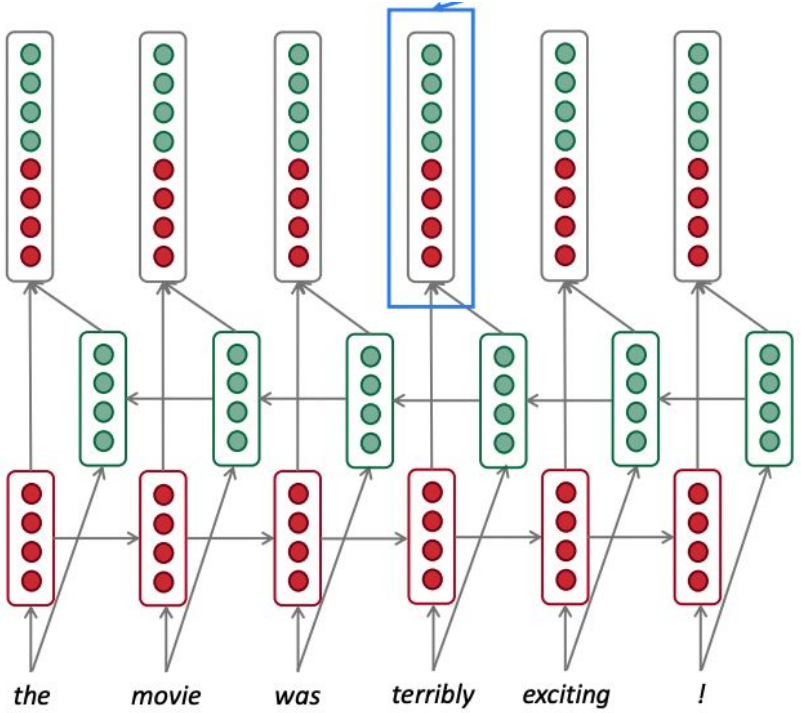


Figure 1: A 5-layer dense block with a growth rate of $k = 4$. Each layer takes all preceding feature-maps as input.

RNN mutants

- Bidirectional RNNs
- multi-layer RNN





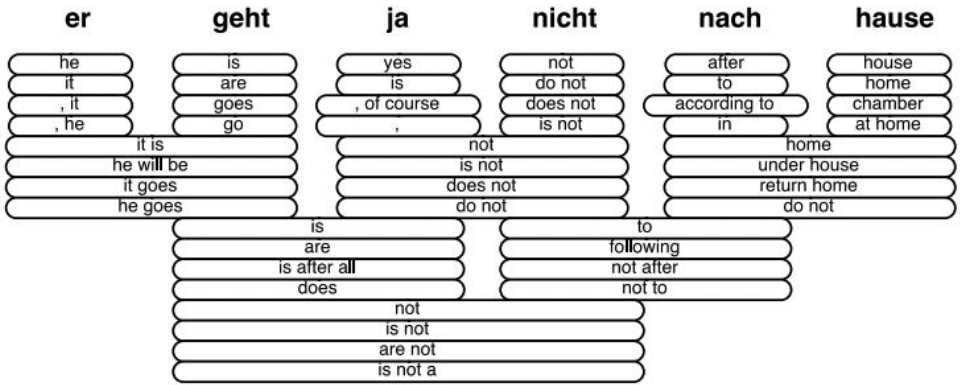
Machine Translation

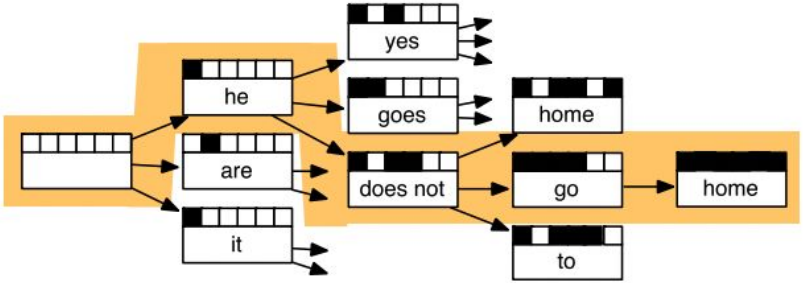
- map text from source language to target language

- statistical(SMT) probabilistic model

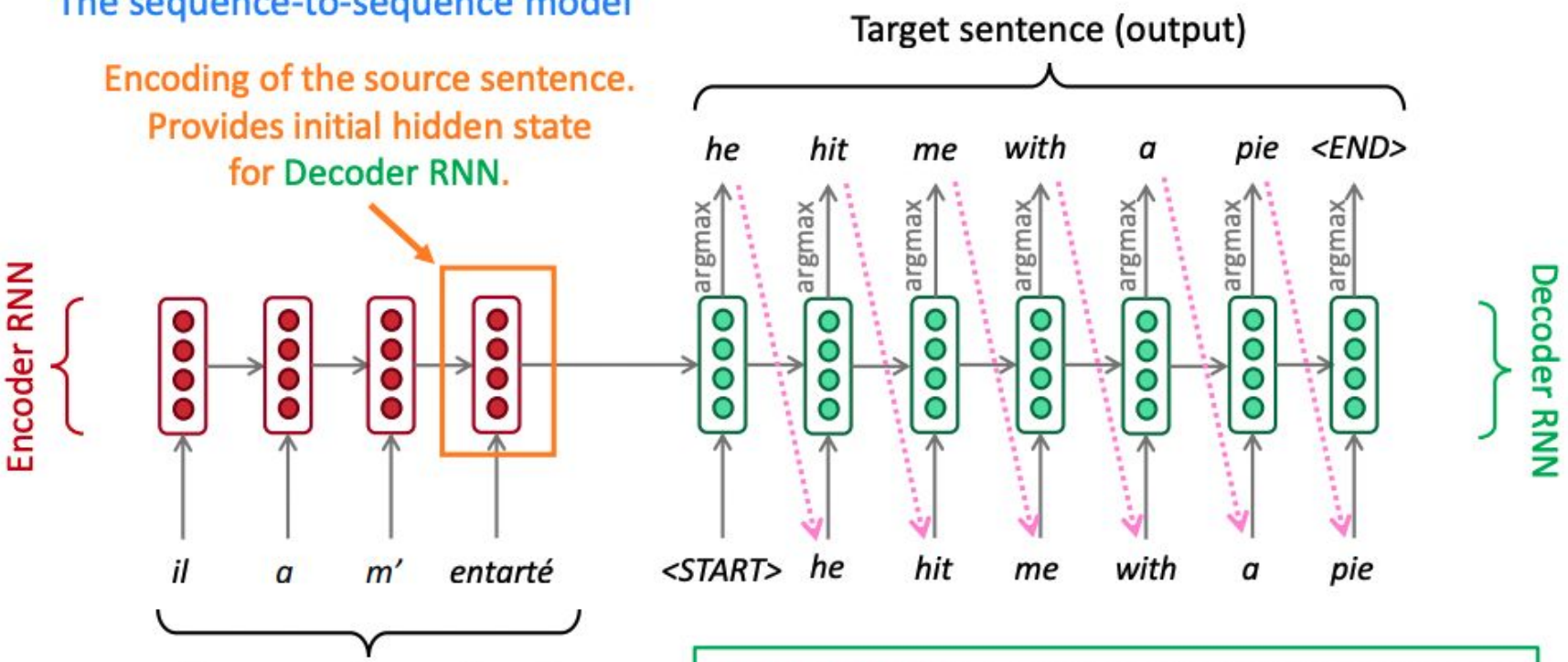
    - many subcomponents

    - extremely complex system





The sequence-to-sequence model



Encoder RNN produces an encoding of the source sentence.

Decoder RNN is a Language Model that generates target sentence, *conditioned on encoding*.

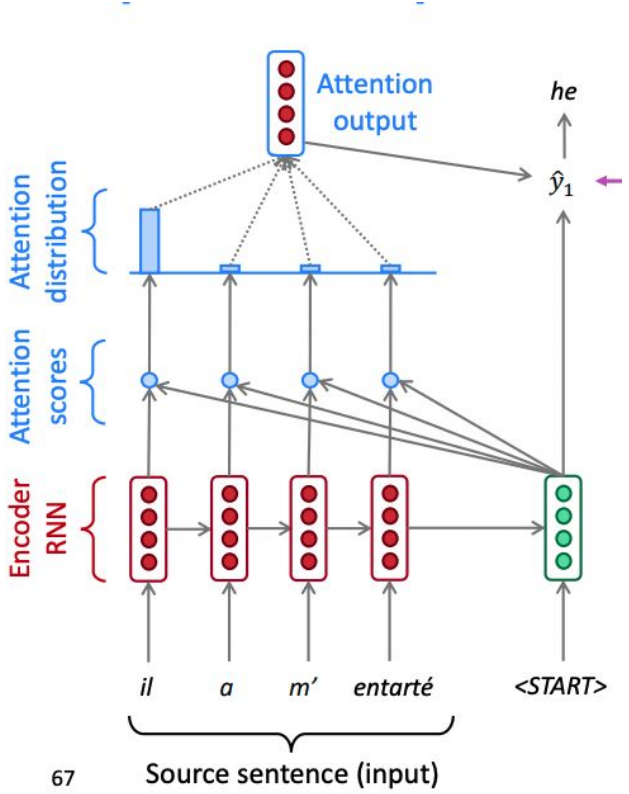
Note: This diagram shows **test time** behavior: decoder output is fed in ⋯⋯→ as next step's input

Sequence-to-sequence

- training
- search algorithm - beam search - end criterion
- pros and cons
- metrics: BLEU (Bilingual Evaluation Understudy)

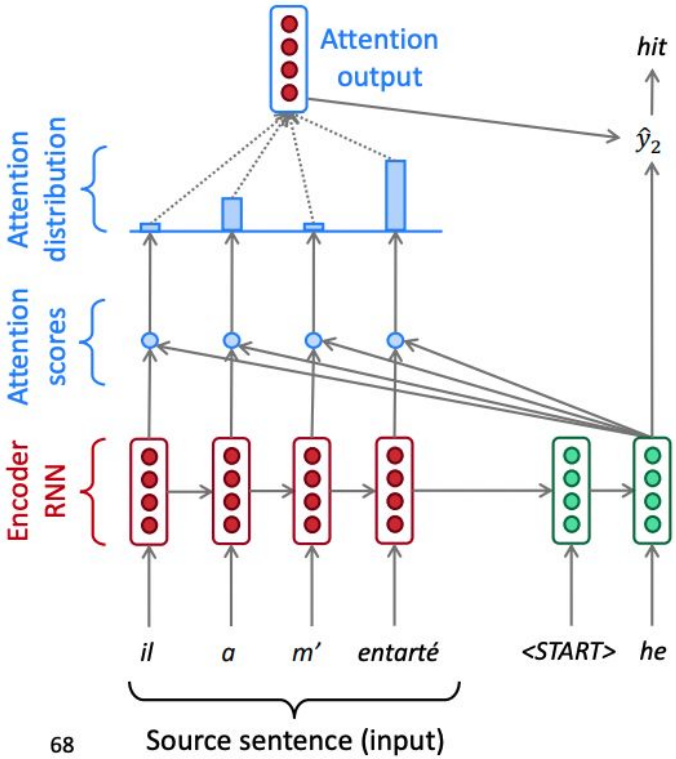
Attention

- direct connection to the encoder



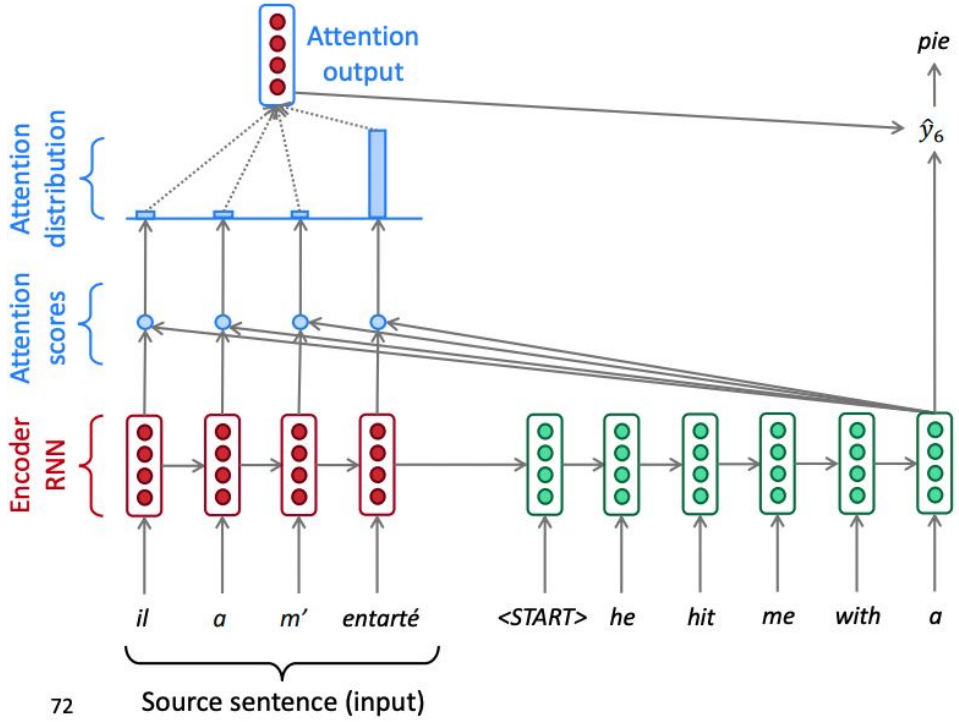
Attention

- direct connection to the encoder



Attention

- direct connection to the encoder



Thank you.