Recurrent Neural Networks

Liad Magen

Long-Distance dependencies in Language

The problem of a fixed-window Neural Language Model:

The trophy would not fit in the brown suitcase because *it* is too **big**.

Trophy

The trophy would not fit in the brown suitcase because *it* is too **small**.

Suitcase

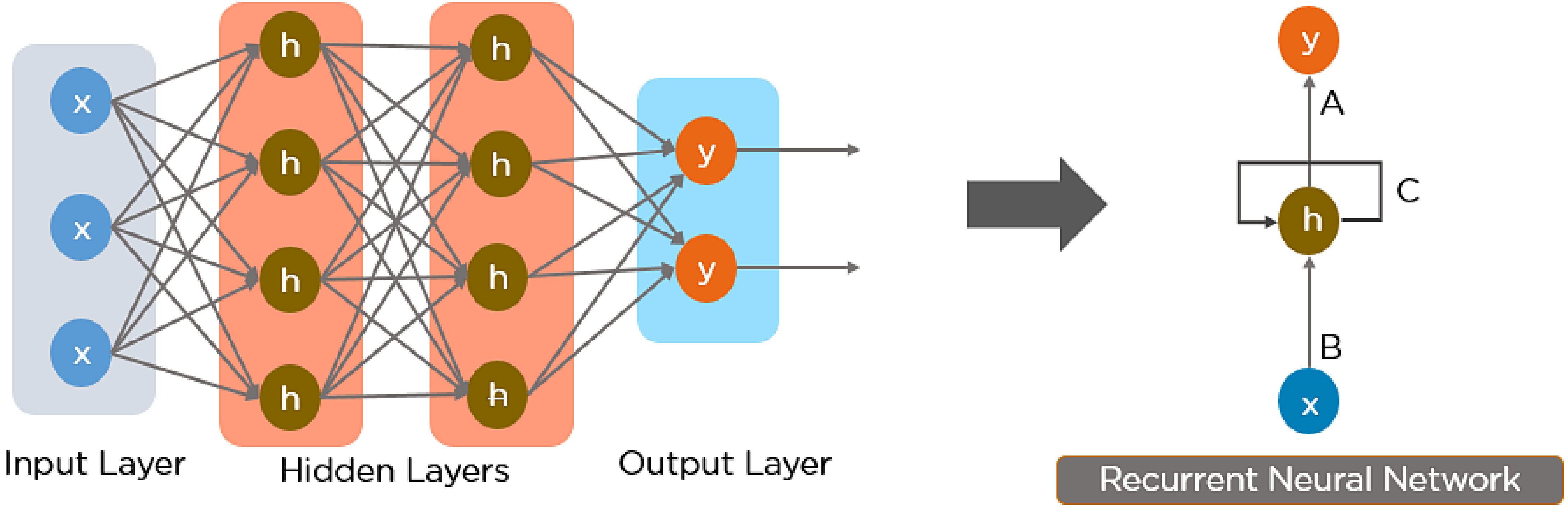
Commonsense Reasoning – Winograd Schema Challenge

Recurrent Neural Network (RNN) (Elman 1990)

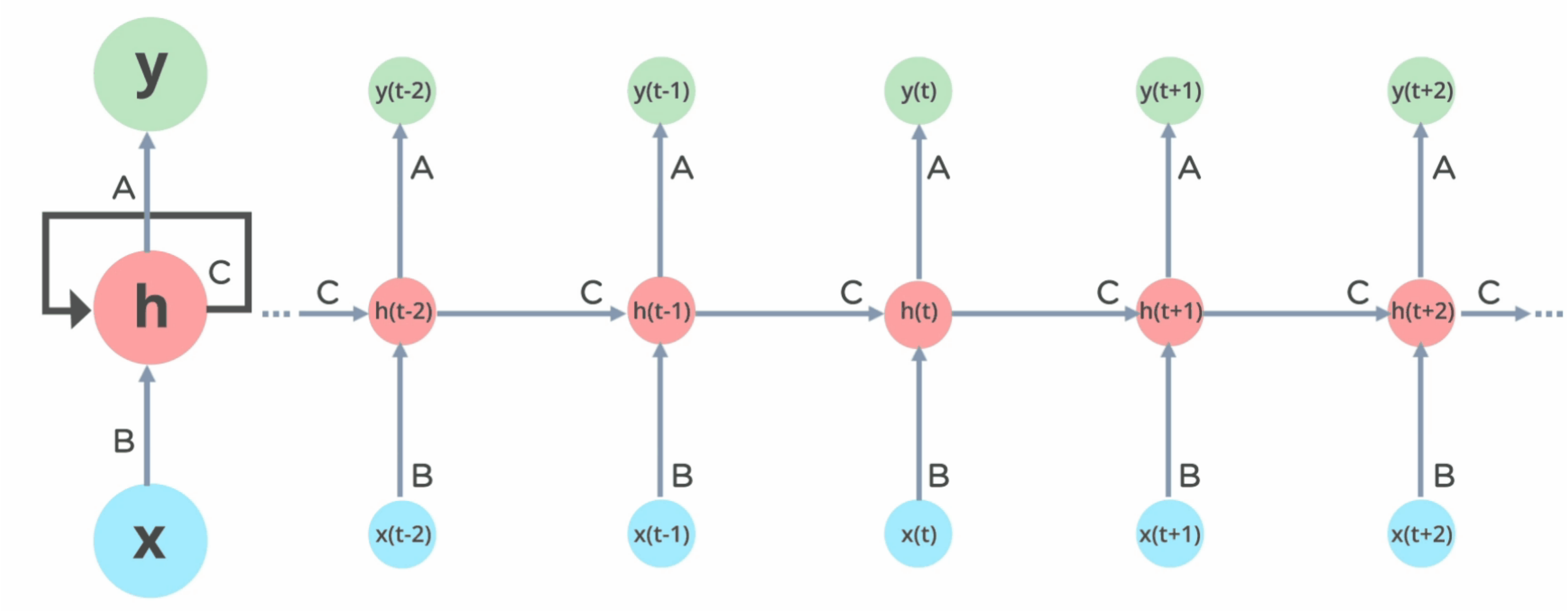
- An unrolled RNN:
  - Very deep Feed-Forward Network
  - Shared parameters across the layers
  - Can get a *new input* at each layer

From NN to RNN

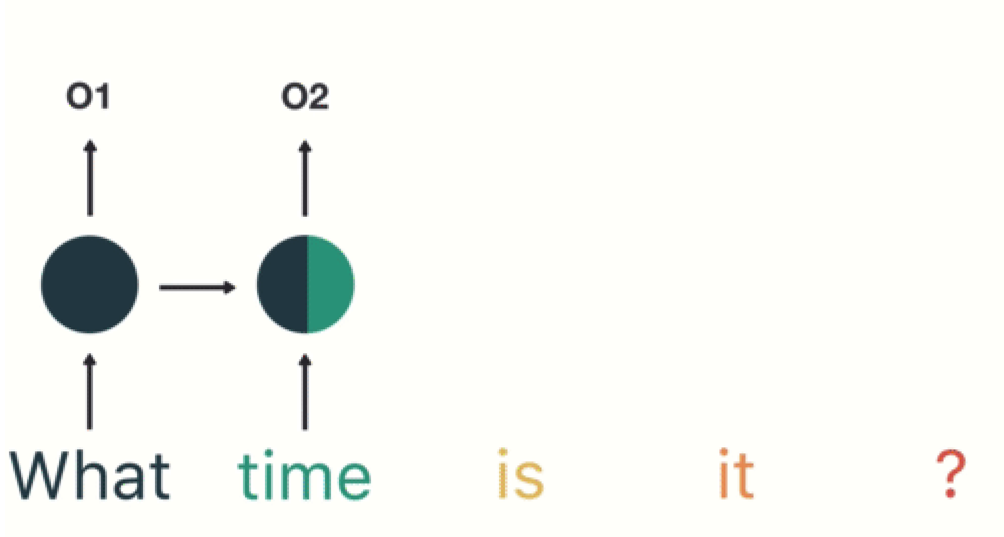
RNN feeds the output of a previous layer together with the next input to predict the output of the layer.



RNN – information cycles through the hidden layer:

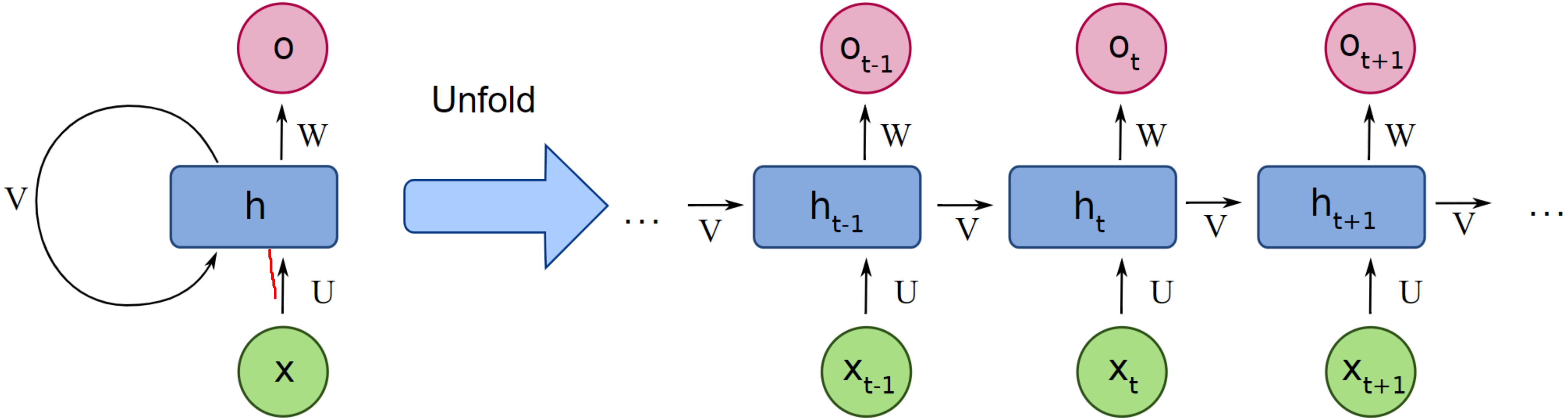


Example – What time is it ?



Recurrent Neural Networks

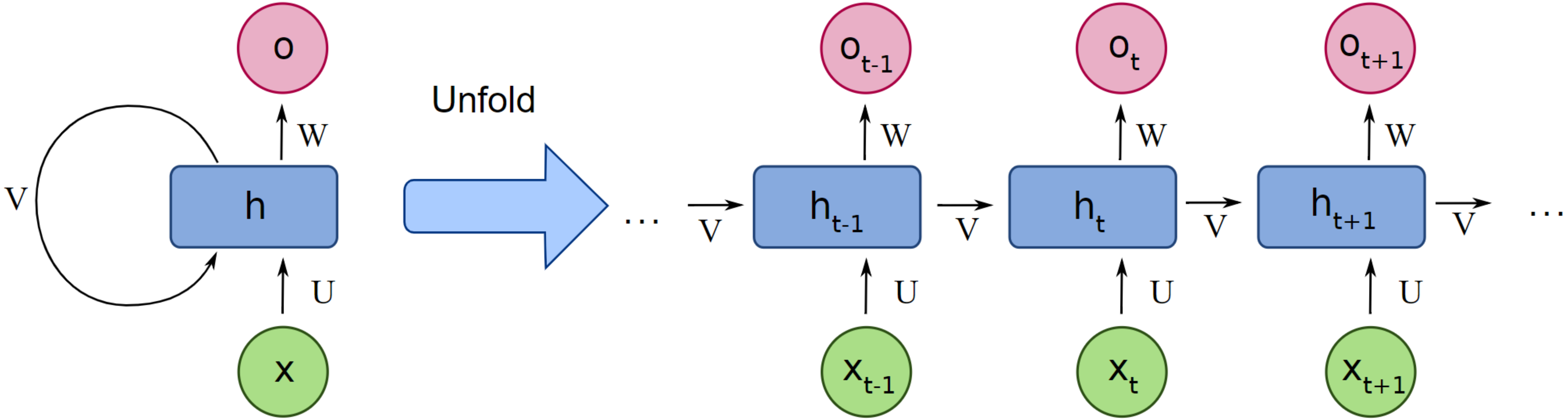
- Very strong models of sequential data
- **Trainable** function (n vectors) --> single vector
- Input: a set of word-vectors: v1, v2, v3…
- Apply the same weight – W - repeatedly
- What is the output?



RNN Intermediate Output Vectors

What are the output vectors (O_t) good for?

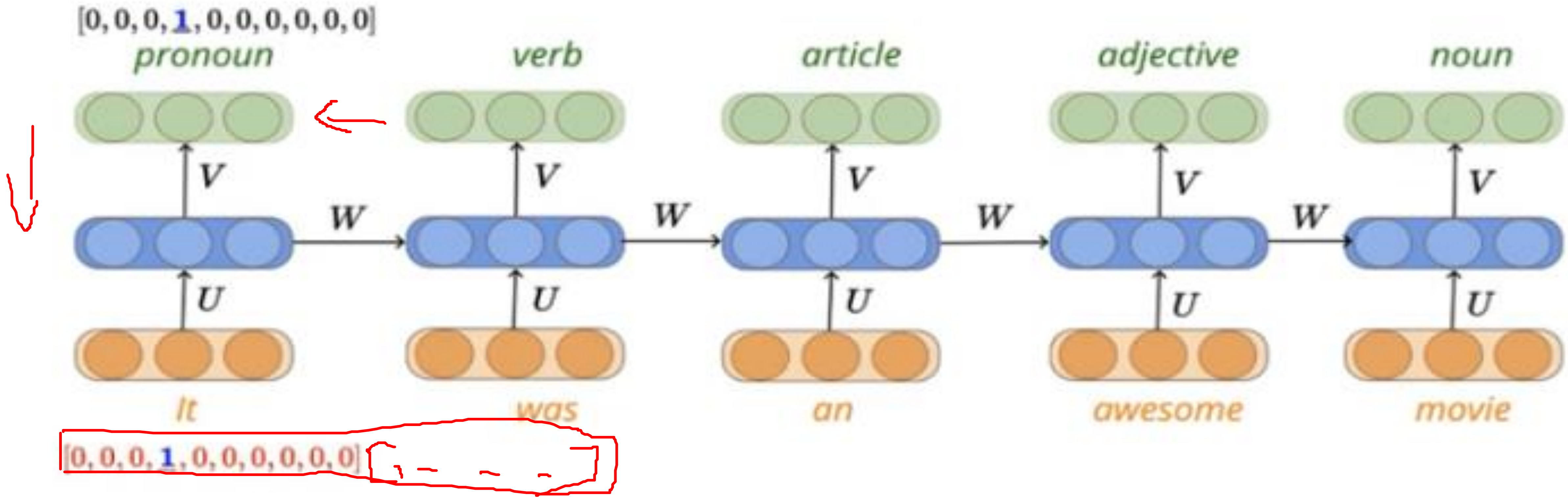
- By default? For Nothing.
  But we can train them!
  - Define function form
  - Define loss



RNN Intermediate Output Vectors

What are the output vectors (O_t) good for Examples:

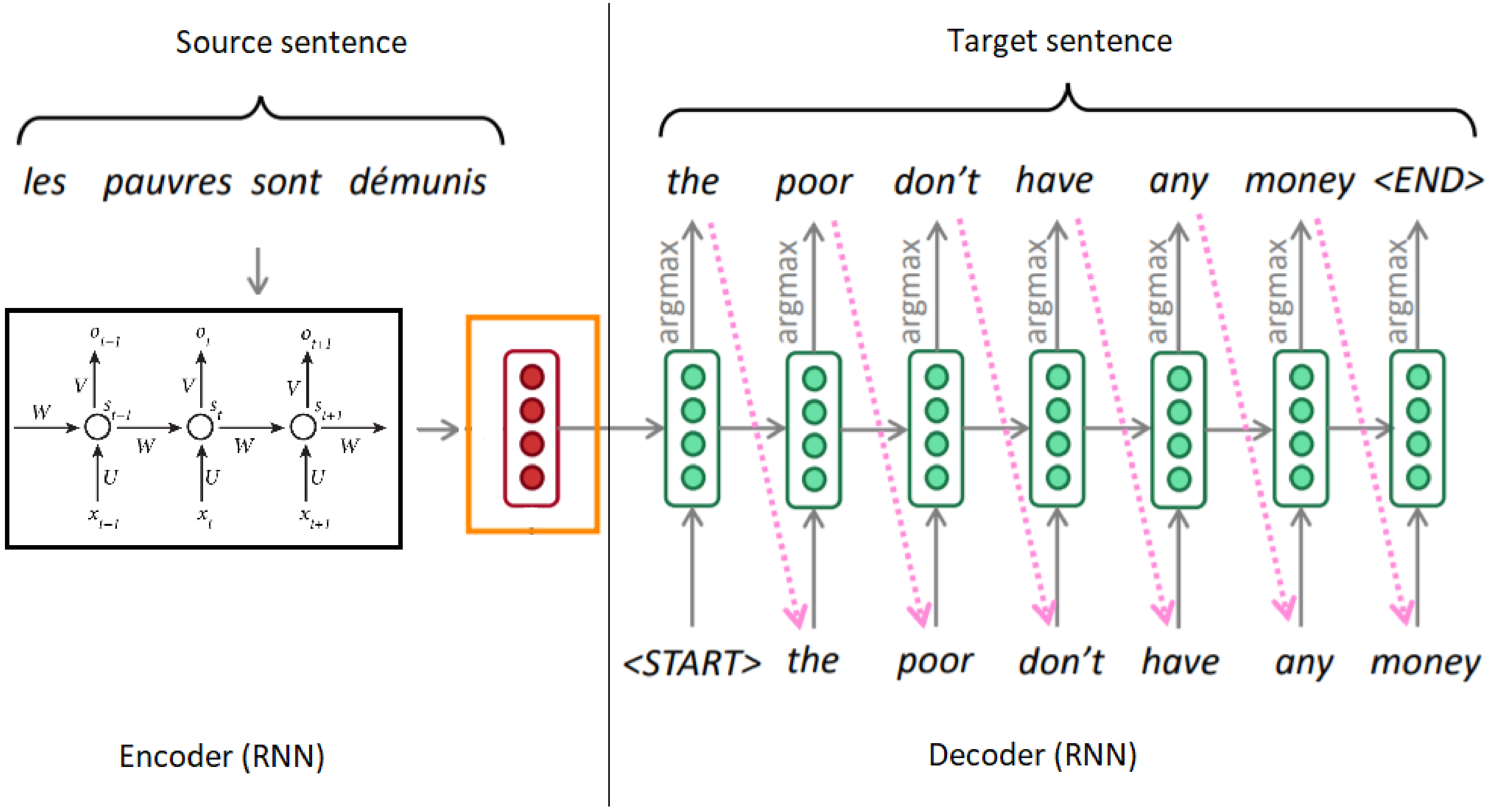
- Next word in the sequence
- Recommednation: Next item to buy/watch
- Tagging: POS, NER, SRL

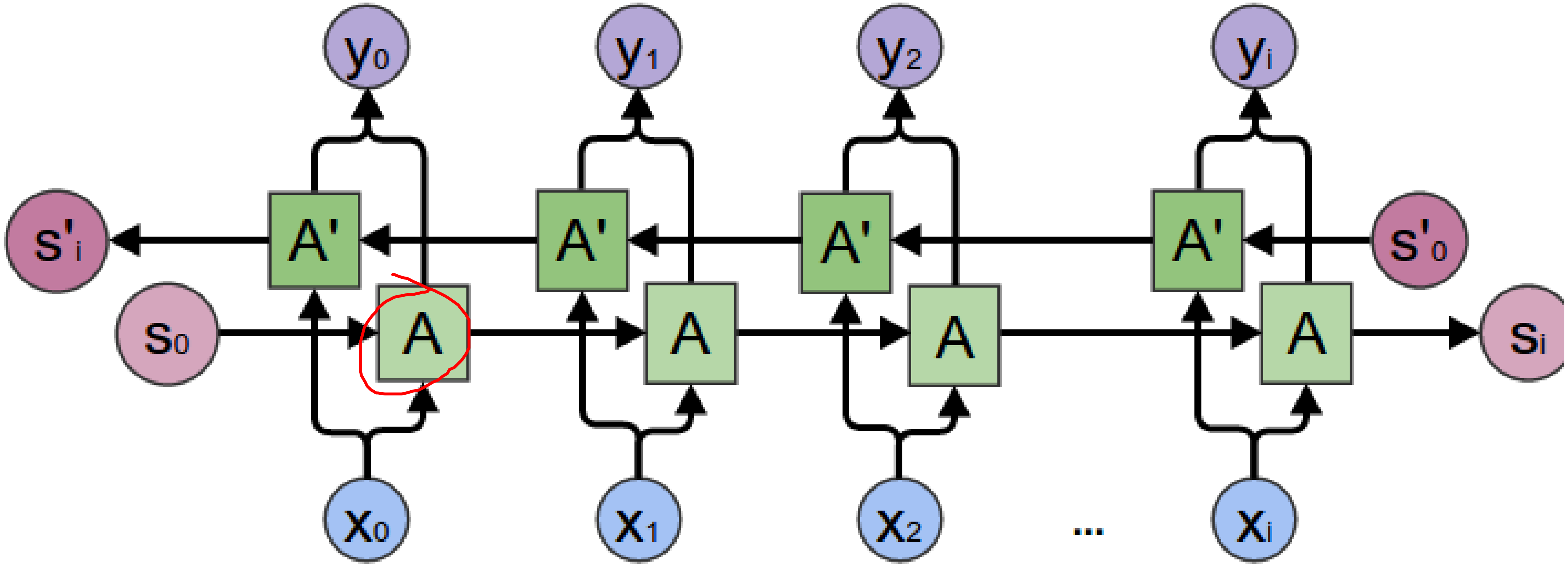


What can RNN do?

- Represent a sentence as a vector
    - Read a whole sentence, make a prediction
- Represent a context in a sentence
    - Read context up until a time-point.
- Generate content

Content Generation





Bidirectional RNNs

One RNN runs left to right.

Another runs right to left.

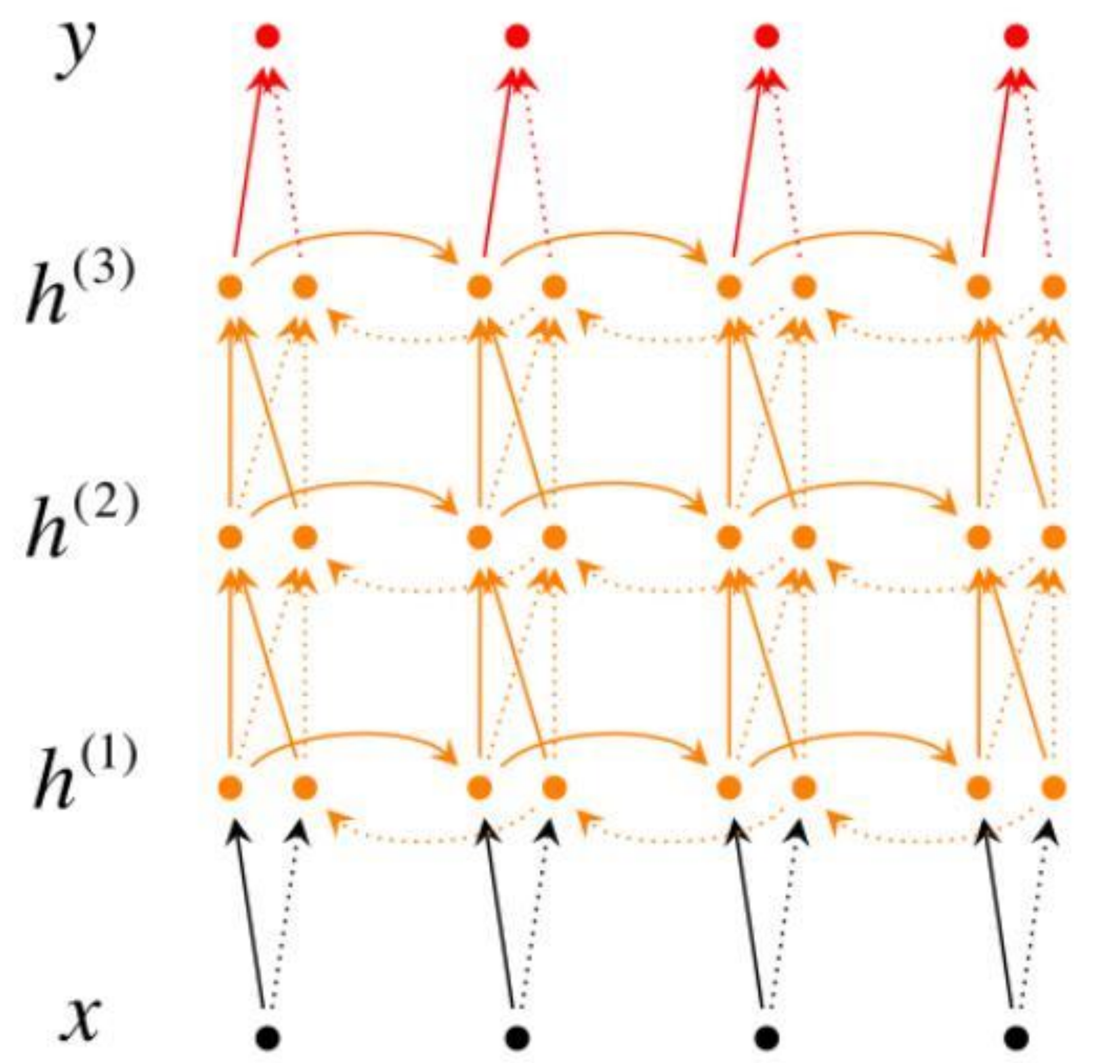
Encode both future and history of a word.

(An infinite window, around the word)

Deep Bi-RNNs

- The hidden layer can be stacked.

- Provides an 'infinite' deep window around a focus word
- Learn to extract what's important
- Easy to train
- Very effective for sequence tagging



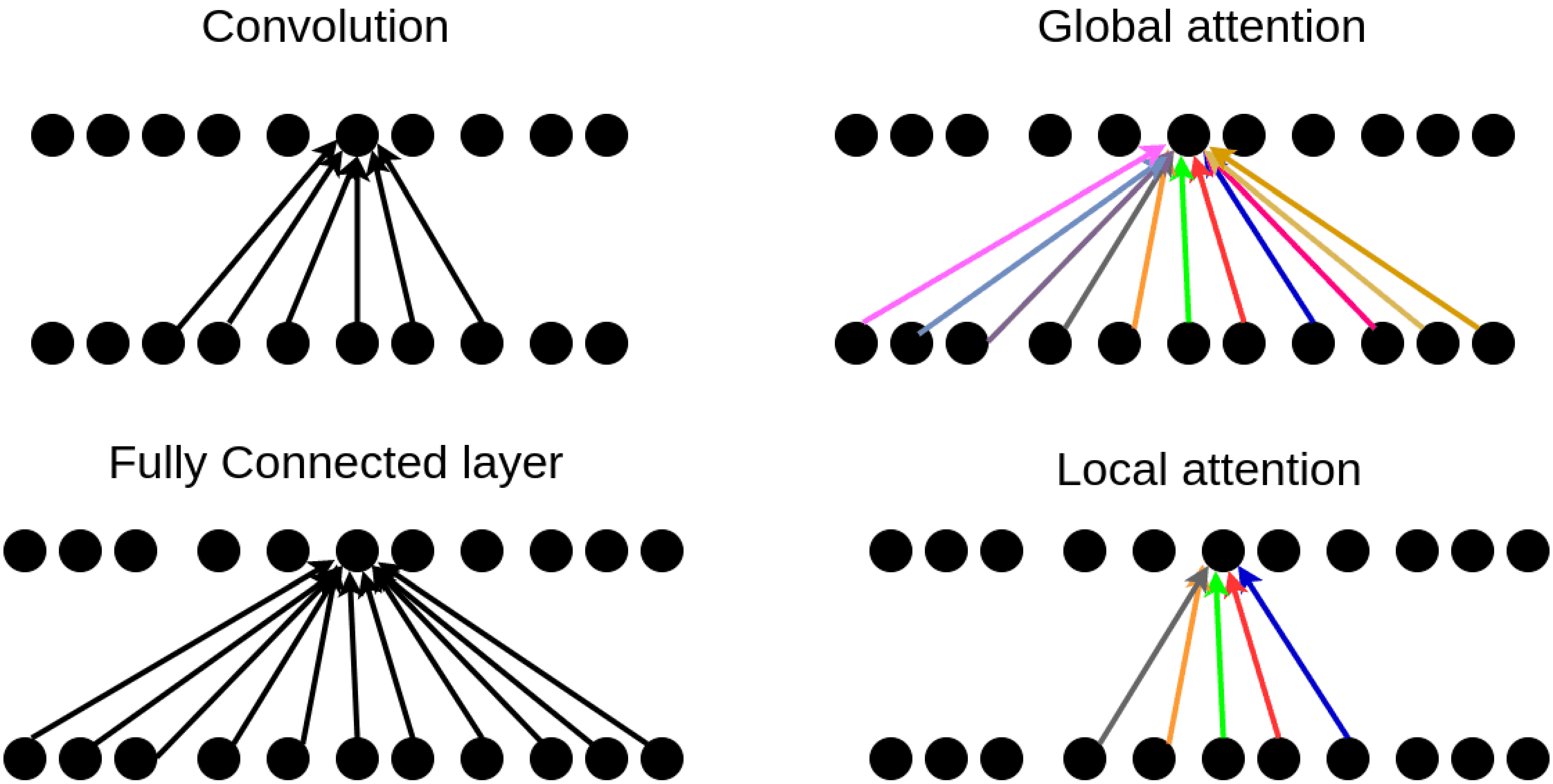
Attention Mechanism (2015)

- “You can’t cram the meaning of a whole $@#ing sentence into a single $@#ing vector”.

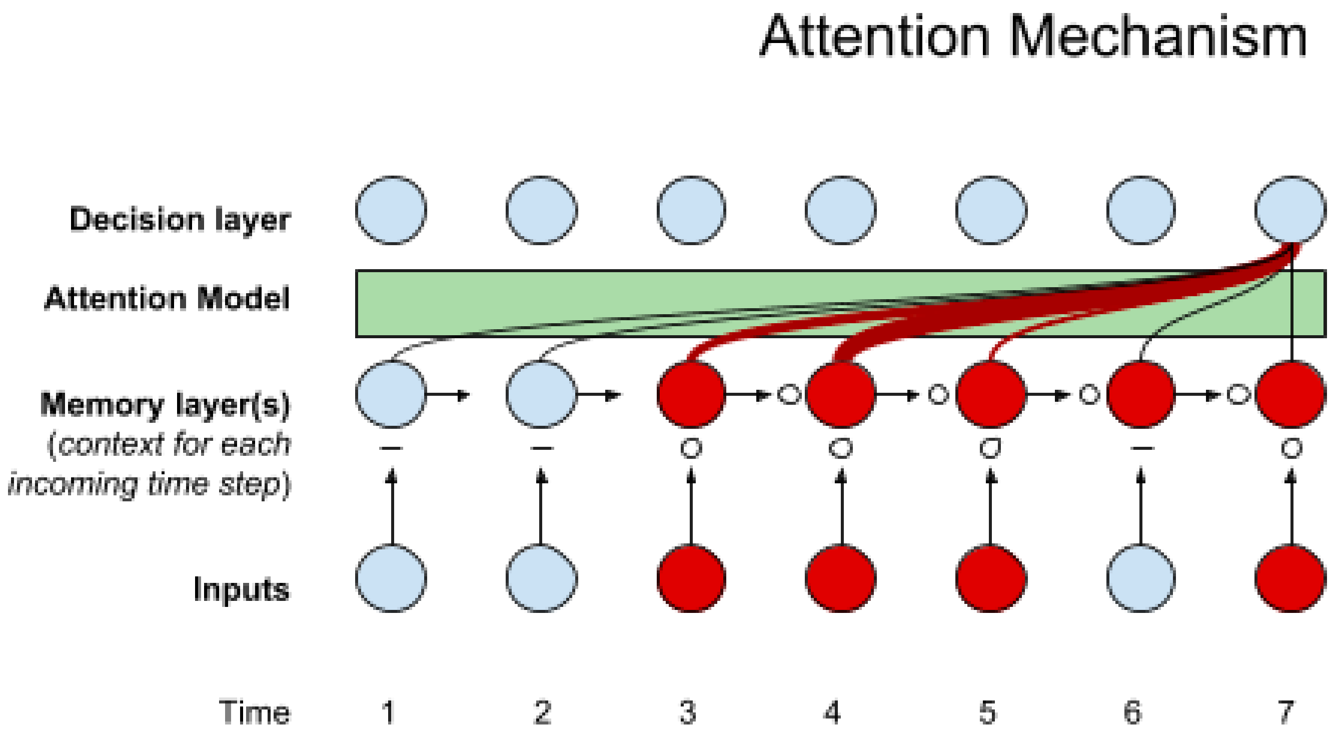
  - Ray Mooney

- Solution:
  - Each word is encoded into a vector
  - When decoding, perform a weighted linear combination of these vectors
  - Use the combination in picking the next word

Attention weights


Convolution
Global attention
Fully Connected layer
Local attention

Attention



Attention Mechanism

Ideas to try it out:

Train on Game of thrones to write the last book...

… Or a new Harry Potter:

Test yourself

- What is a Language Model?

A system that predicts the next word

- How is the Linguistic field of human sounds called?

Phonetics / Phonology

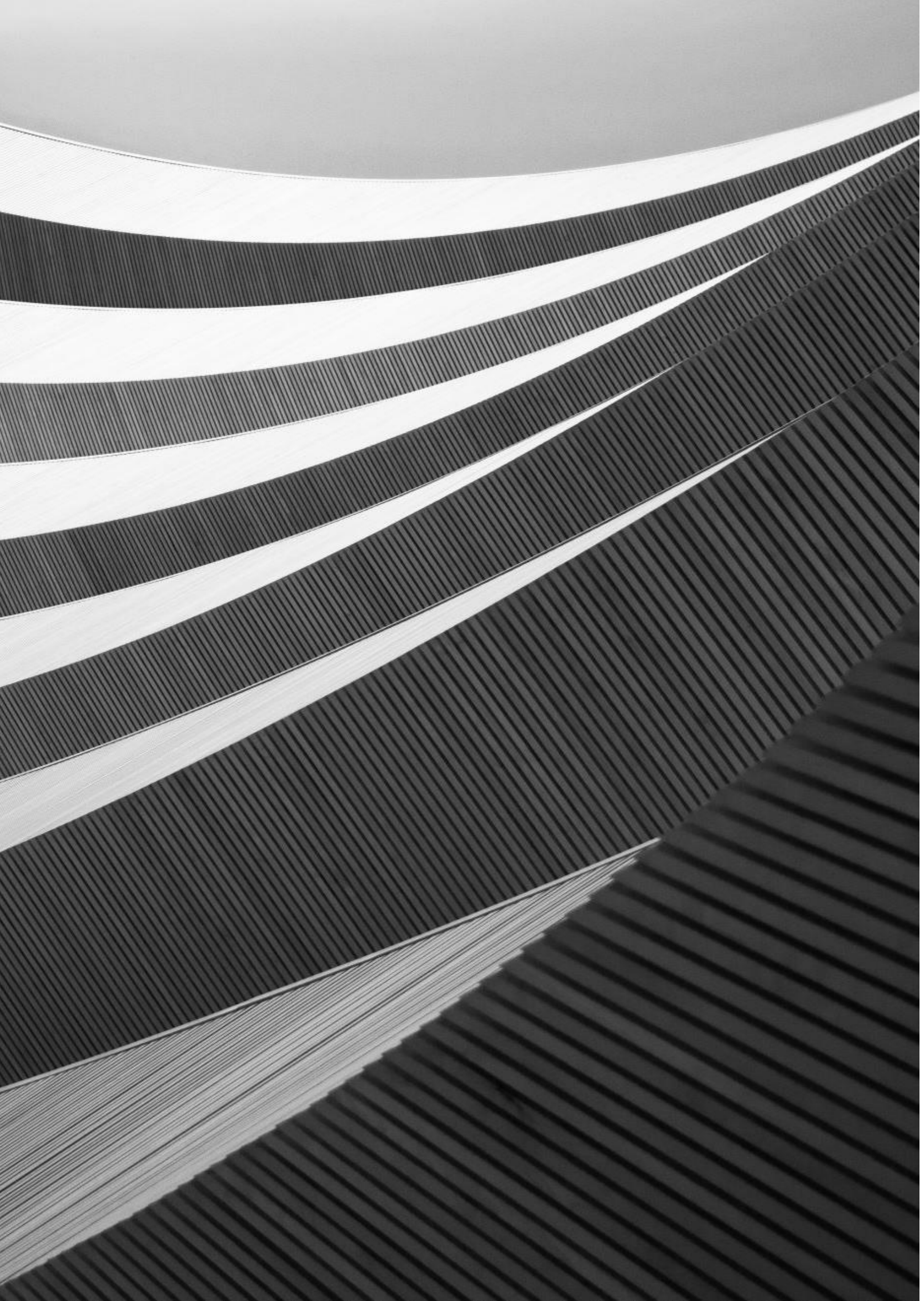
- How is the field of words and their parts called?

Morphology

BI-RNN's Recap

- Represent the history up to a point in the sequence, and the future from a point in a sequence.
- Feed into an MLP (or linear classifier) to classify the point based on history and future.
- The network learns which features are important in the history and future for the given prediction task.
- "Infinite window"

Multi-task Learning

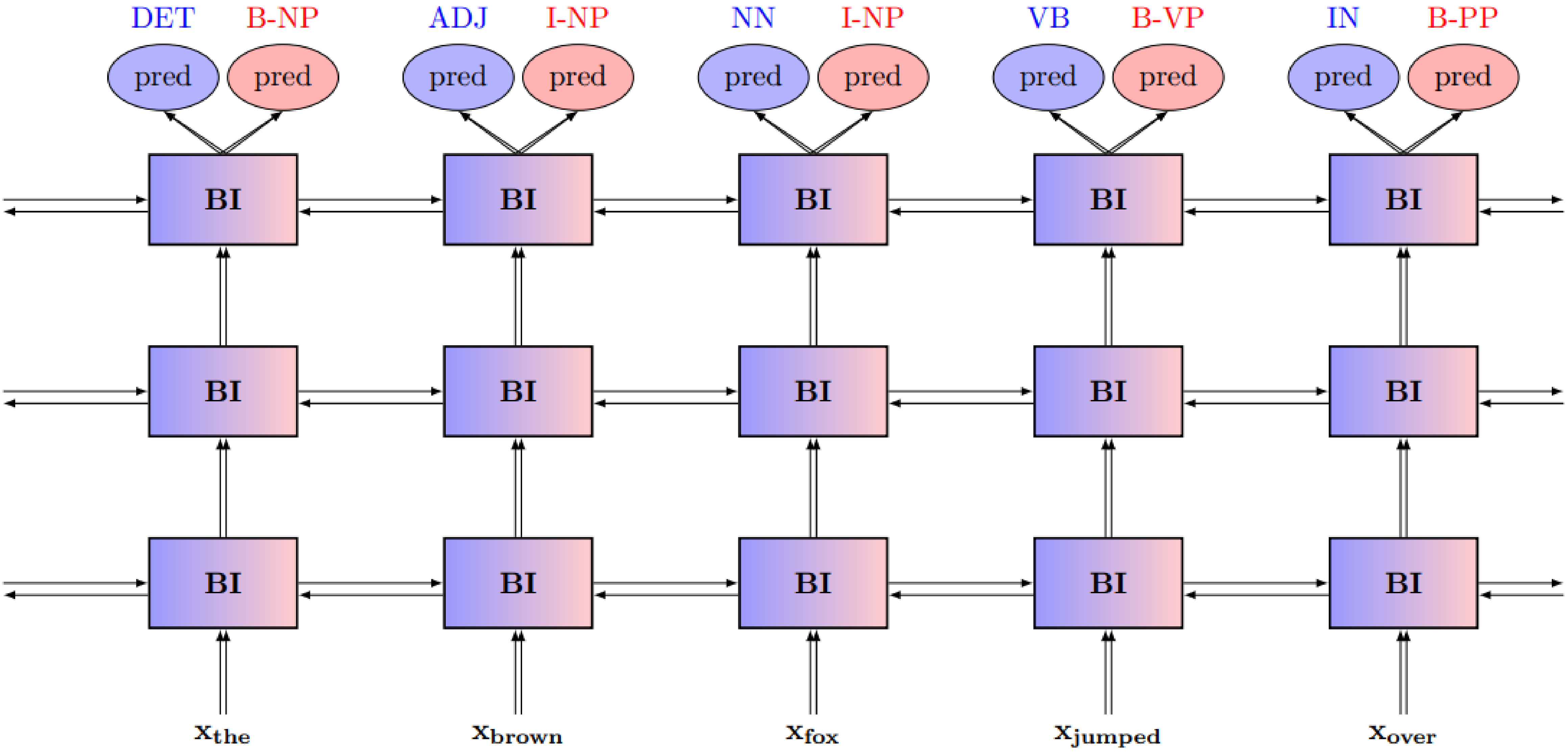


Multi-task Learning

Different sequence prediction tasks have shared structures.

Hints for predicting A may help to predict B.

Instead of training a network to do one thing, train it to do several things.

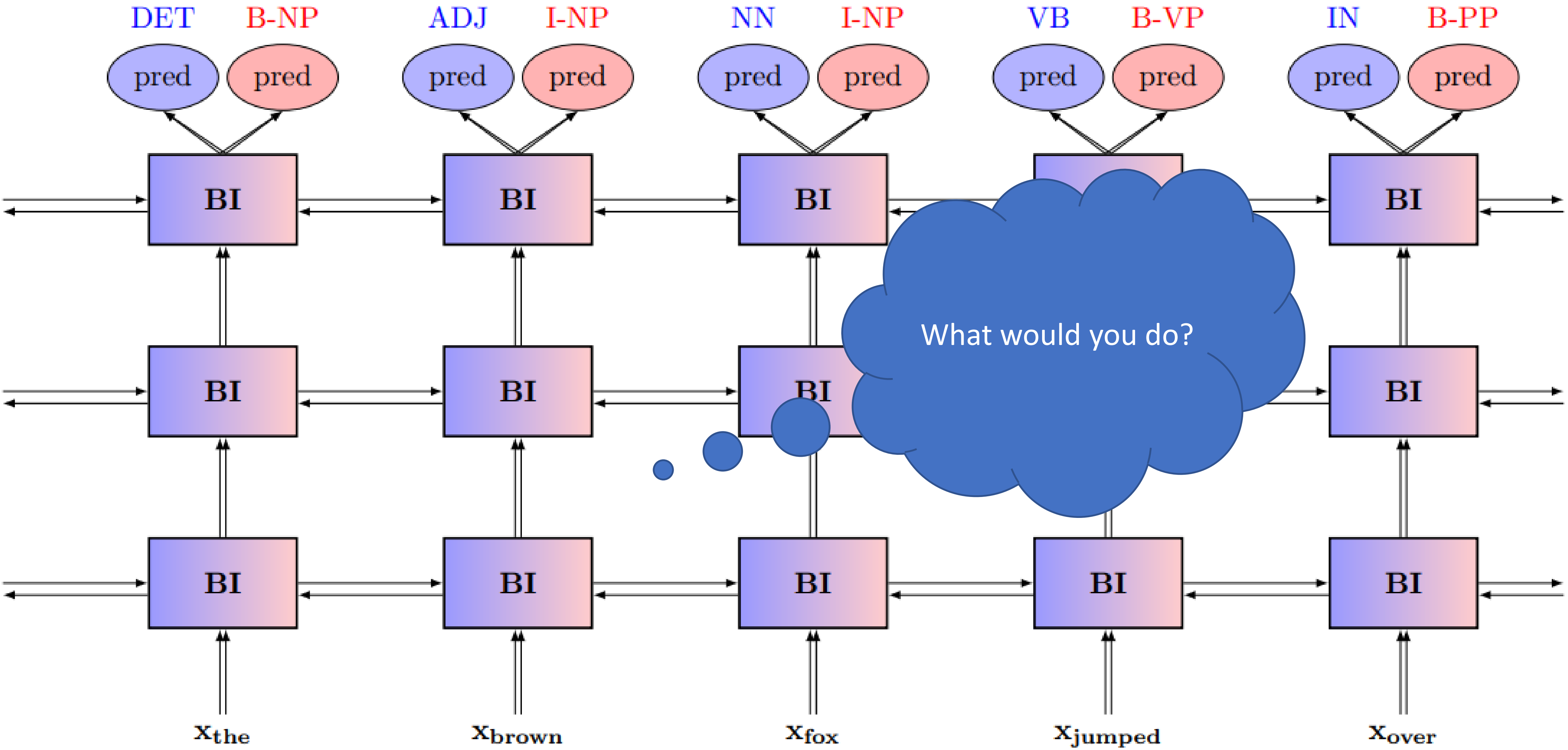

DET
B-NP
ADJ
I-NP
NN
I-NP
VB
B-VP
IN
B-PP
pred
BI
$\mathbf{x}_{\mathbf{the}}$
$\mathbf{x}_{\mathbf{brown}}$
$\mathbf{x}_{\mathbf{fox}}$
$\mathbf{x}_{\mathbf{jumped}}$
$\mathbf{x}_{\mathbf{over}}$

But...

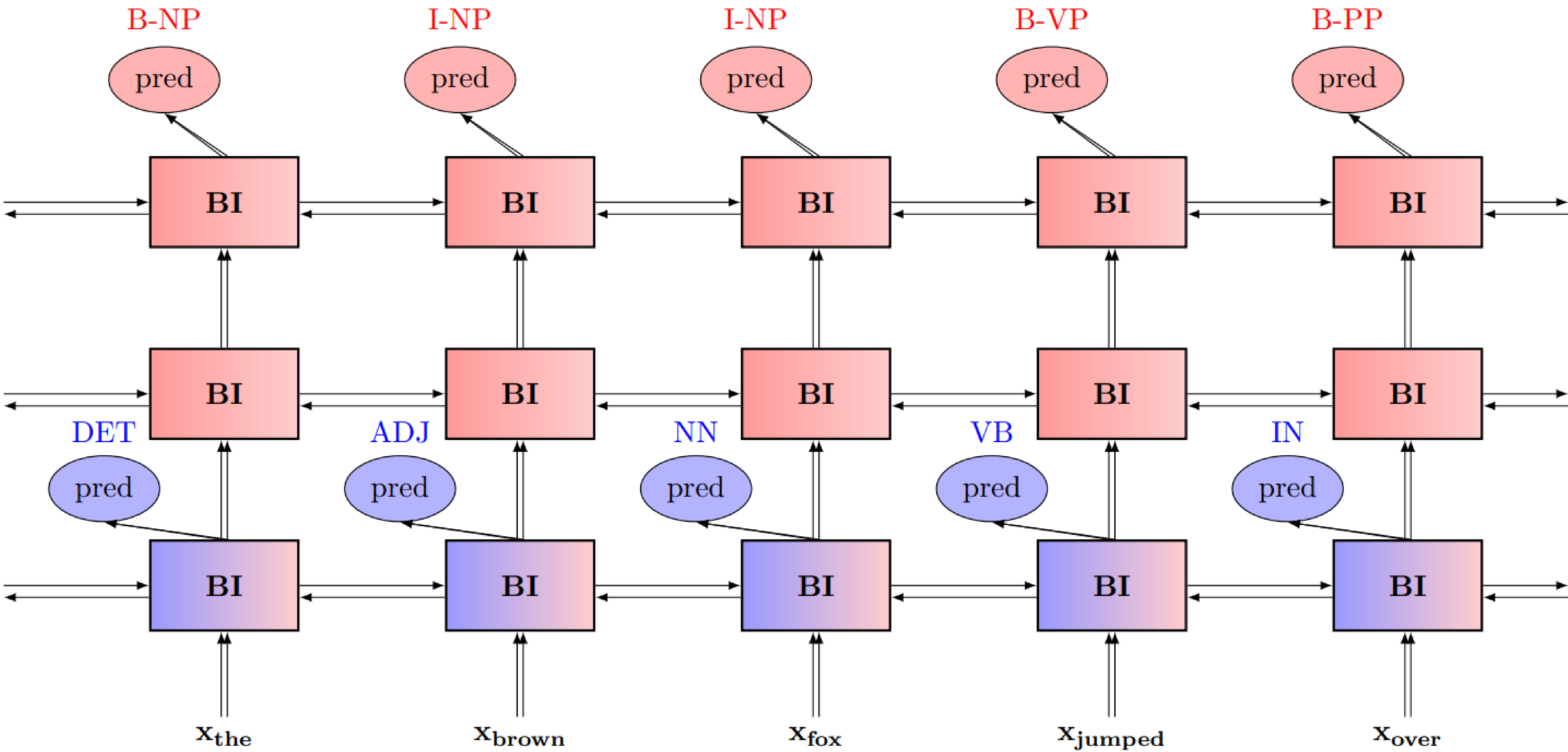
Not easy to get it to work

For many task pairs – there's no improvement, at all

If the network not wide/deep enough, MTL hurts both tasks...


DET
B-NP
ADJ
I-NP
NN
I-NP
VB
B-VP
IN
B-PP
pred
BI
$\mathbf{x}_{\mathbf{the}}$
$\mathbf{x}_{\mathbf{brown}}$
$\mathbf{x}_{\mathbf{fox}}$
$\mathbf{x}_{\mathbf{jumped}}$
$\mathbf{x}_{\mathbf{over}}$

What would you do?


B-NP
I-NP
I-NP
B-VP
B-PP
pred
pred
pred
pred
pred
BI
BI
BI
BI
BI
BI
BI
BI
BI
BI
DET
ADJ
NN
VB
IN
pred
pred
pred
pred
pred
BI
BI
BI
BI
BI
$\mathbf{x}_{\mathbf{the}}$
$\mathbf{x}_{\mathbf{brown}}$
$\mathbf{x}_{\mathbf{fox}}$
$\mathbf{x}_{\mathbf{jumped}}$
$\mathbf{x}_{\mathbf{over}}$

RNN - Summary

- Many tasks can be solved with the RNN family
    - And many ARE being solved
- Be creative with the architecture