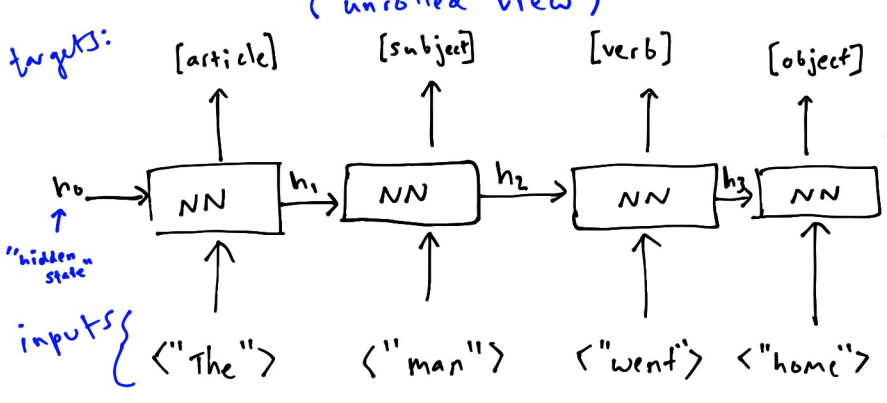
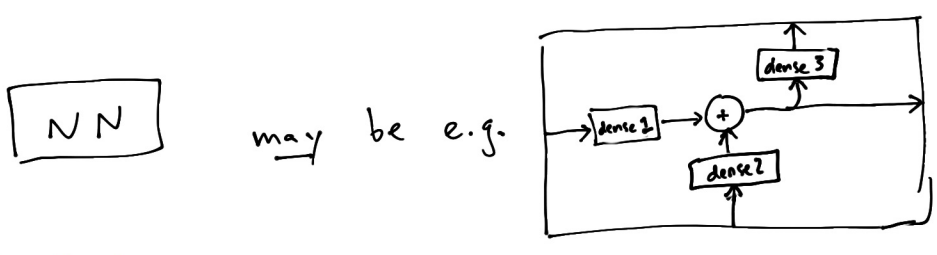
Recurrent Neural Networks

Concept: For sequential tasks, can we reuse the same neural network?

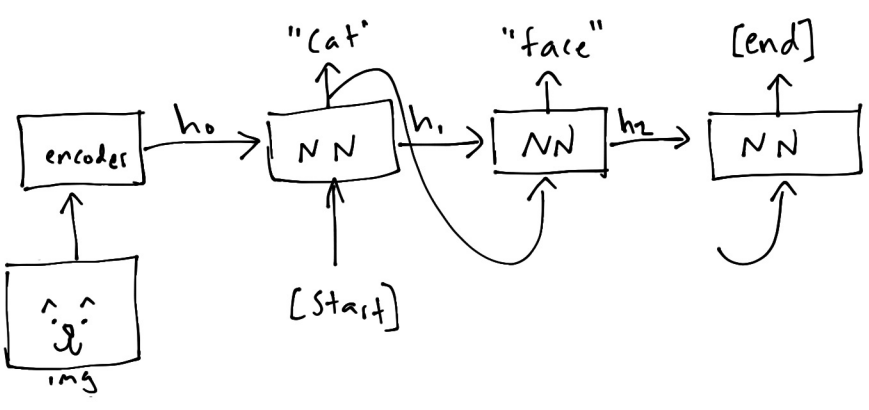
Example: Part-of-Speech labeling

("unrolled view")





Example 2: Caption Generation



RNNs can model

1) Many-to-many relationships e.g. part-of-speech labeling
2) Many-to-one relationships e.g. sentiment classification
3) one-to-many relationships e.g. caption generation

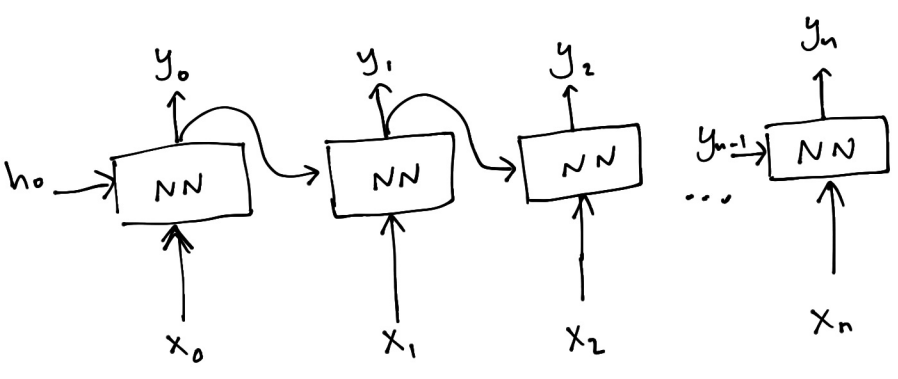
RNN Motivating Example: The XOR Problem

Given a binary sequence, predict whether sum is even or odd

Example inputs: $\vec{x}_1 = [0\ 0\ 1\ 0\ 1\ 0\ 0\ 0]$
$\vec{x}_2 = [0\ 1\ 0\ 1\ 1\ 0\ 0\ 0]$

Correct classification: $y_{1,true} = 1$ (even)
$y_{2,true} = 0$ (odd)

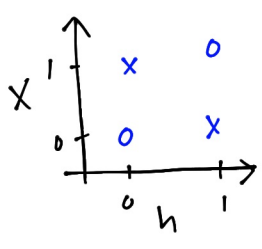
Consider framing as RNN, where network outputs current "evenness" after each input:



What function should each RNN component be learning?

h	x	y
0	0	1
0	1	0
1	0	0
1	1	1

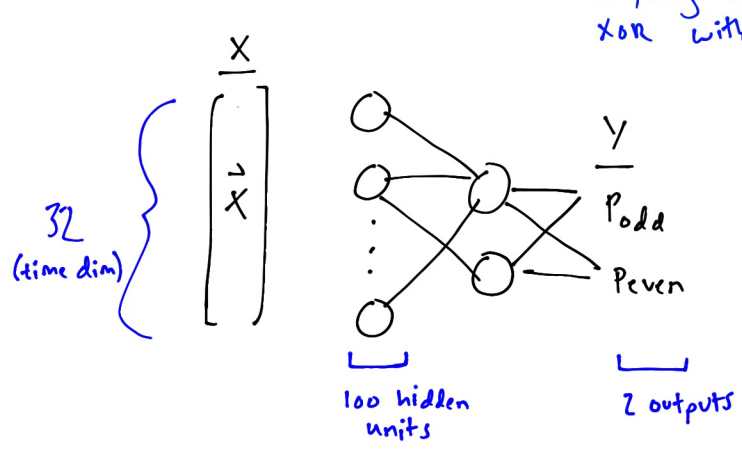
→



→ recurrent units can solve problem with XOR func.

[Notebook of Failure]

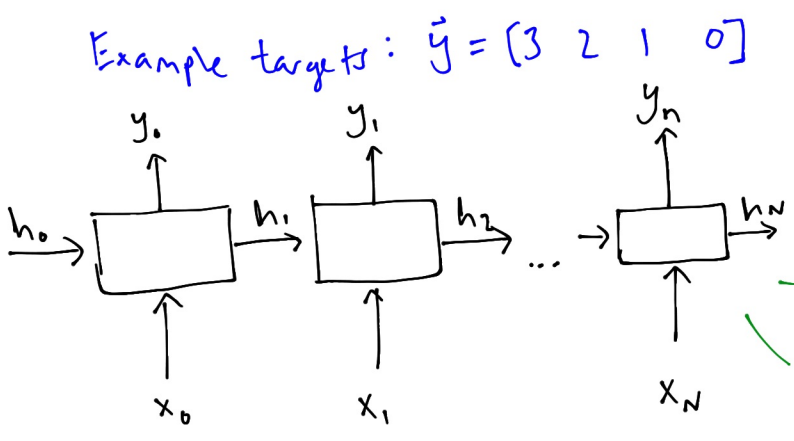
trying to learn XOR with dense NN



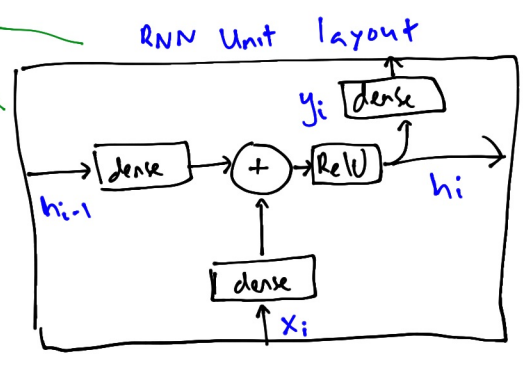
[NotebookofFailureRedux]

Try to learn sequence reversal on RNN

Example targets: $\vec{y} = [3\ 2\ 1\ 0]$

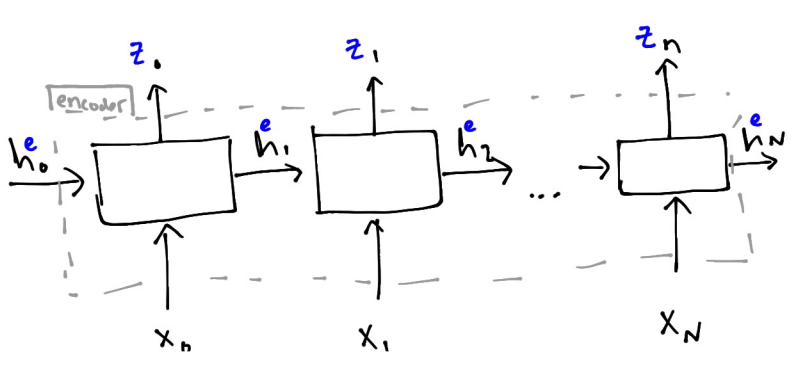


Example input: $\vec{x} = [0\ 1\ 2\ 3]$

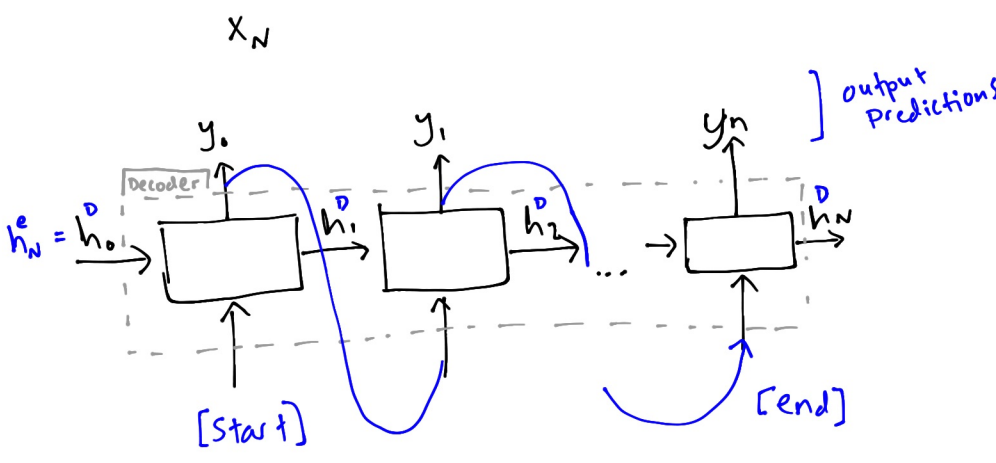


[Seq2Seq]

test (and succeed) with encoder/decoder framework



(note: encoder outputs z unused)



[Read & Ask about Attn, we'll cover it conceptually tmw]