

# LARGE LANGUAGE MODELS

David Kauchak  
CS 159 – Spring 2023

1

## Admin

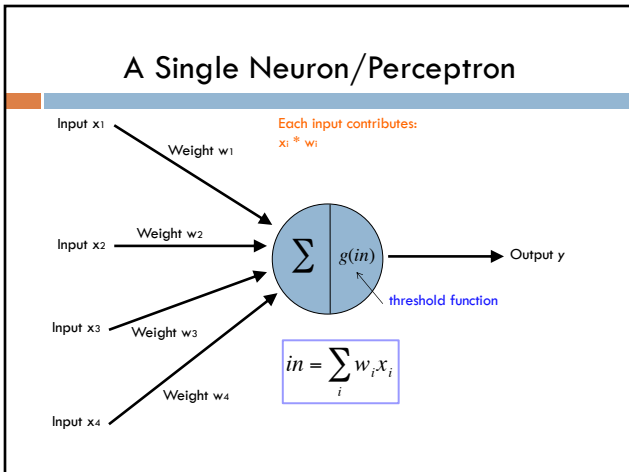
Final project proposals due today

Start working on the projects!

- ▣ Log hours that you work

Mentor hours this week?

2



3

### Activation functions

hard threshold:

$$g(in) = \begin{cases} 1 & \text{if } in \geq T \\ 0 & \text{otherwise} \end{cases}$$

sigmoid

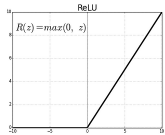
$$g(x) = \frac{1}{1 + e^{-ax}}$$

tanh x

4

## Many other activation functions

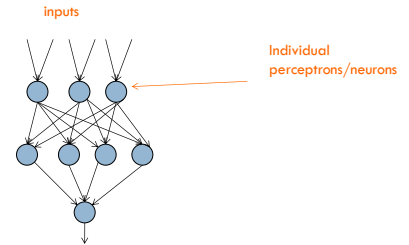
Rectified Linear Unit



Softmax (for probabilities)

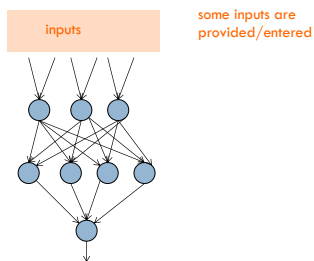
5

## Neural network



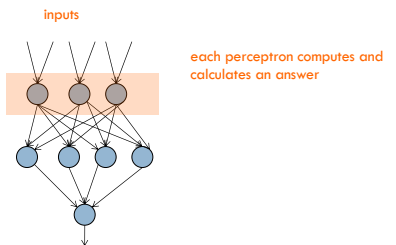
6

## Neural network

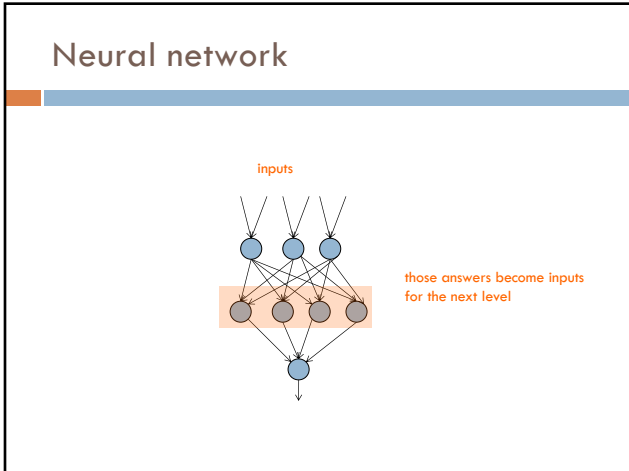


7

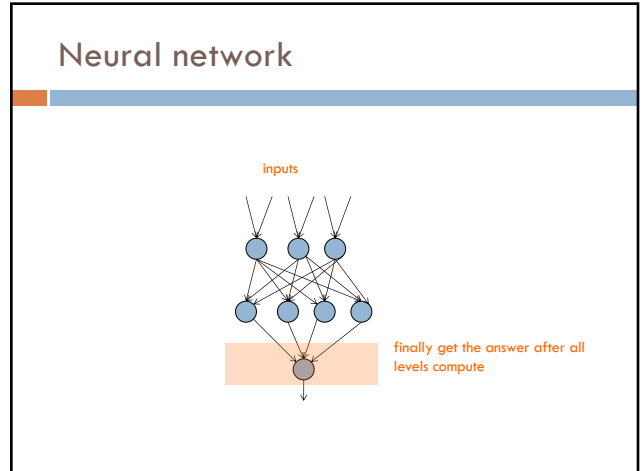
## Neural network



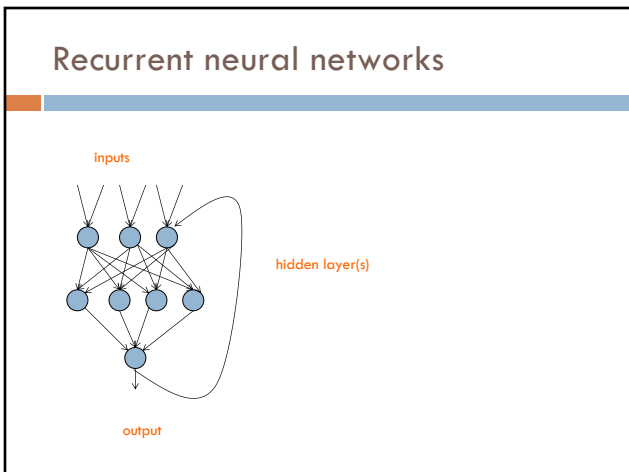
8



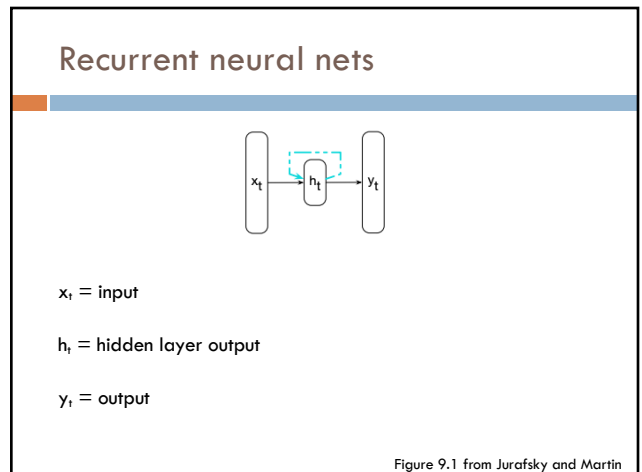
9



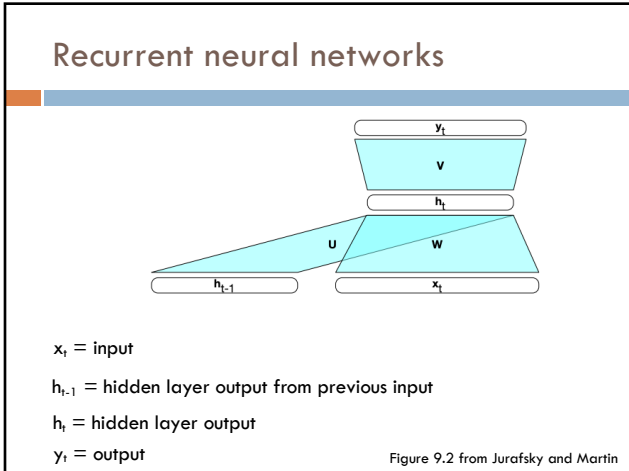
10



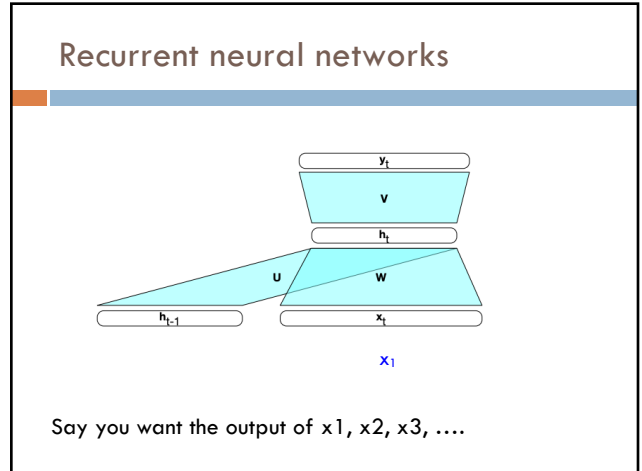
11



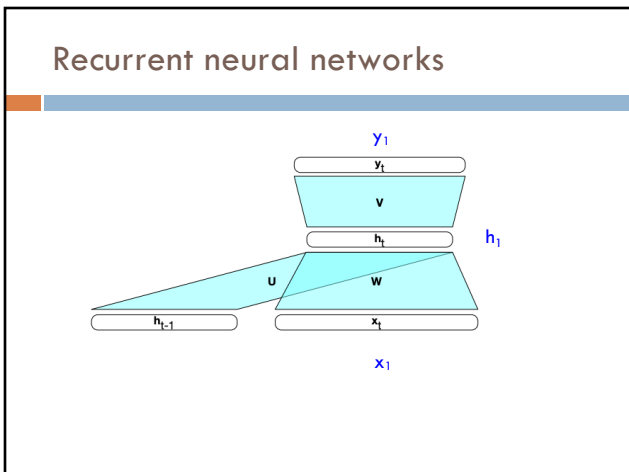
12



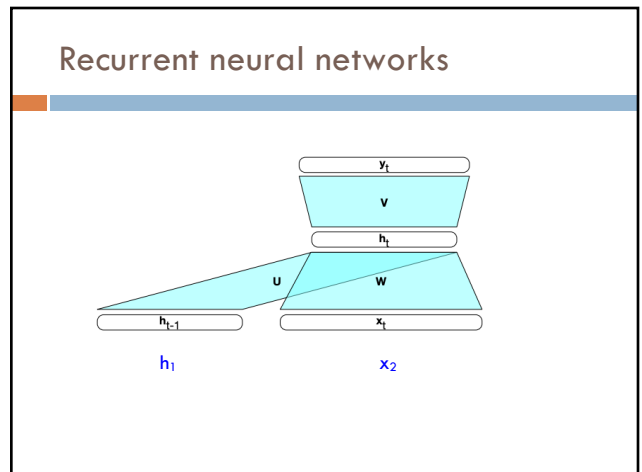
13



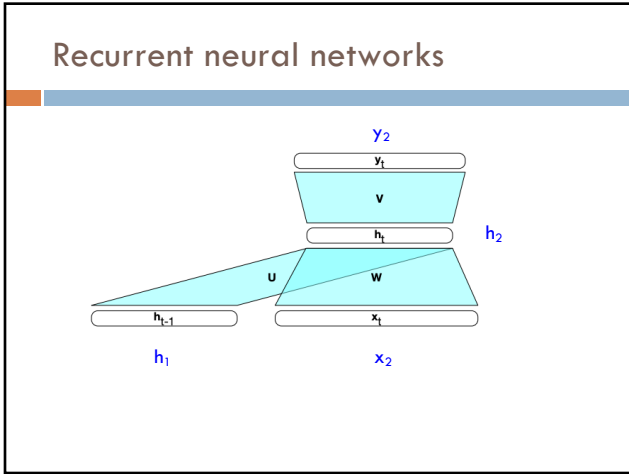
14



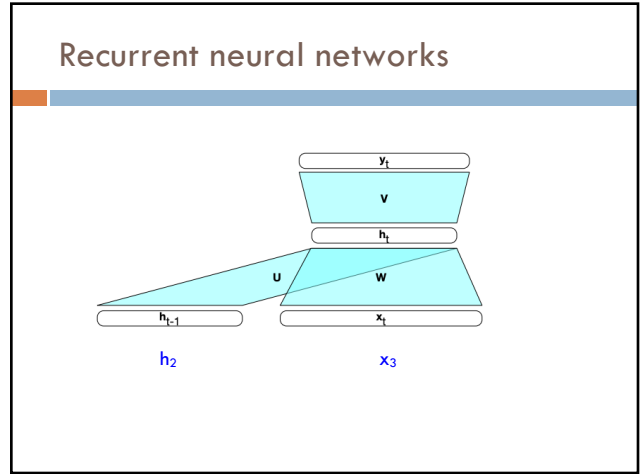
15



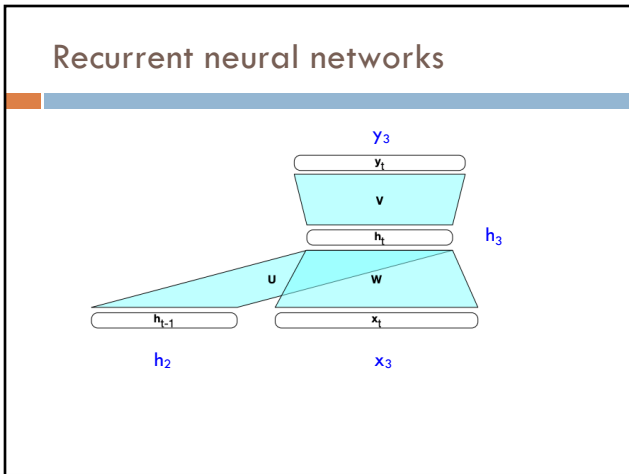
16



17



18



19

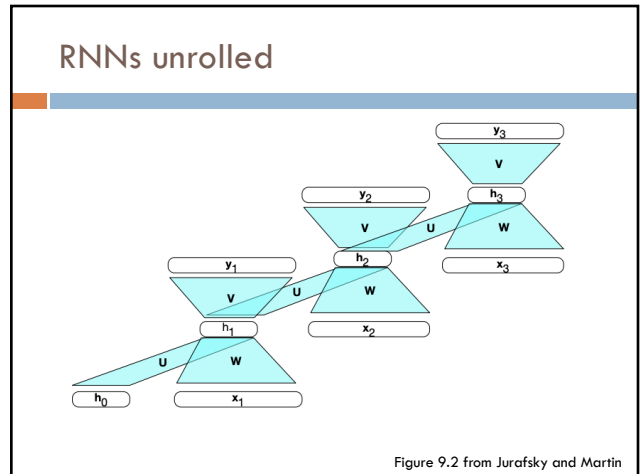


Figure 9.2 from Jurafsky and Martin

20

### Still just a single neural network

U, W and V are the weight matrices

$x_t$  = input  
 $h_{t-1}$  = hidden layer output from previous input  
 $h_t$  = hidden layer output  
 $y_t$  = output

Figure 9.2 from Jurafsky and Martin

21

### RNN language models

How can we use RNNs as language models  $p(w_1, w_2, \dots, w_n)$ ?  
 How do we input a word into a NN?

22

### "One-hot" encoding

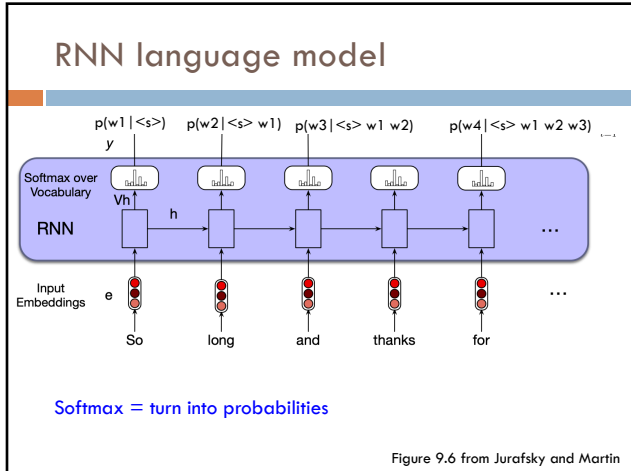
For a vocabulary of V words, have V input nodes

All inputs are 0 except the one corresponding to the word

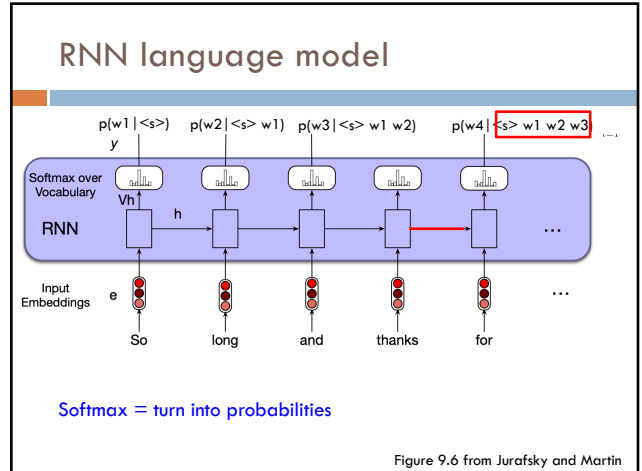
23

### RNN language model

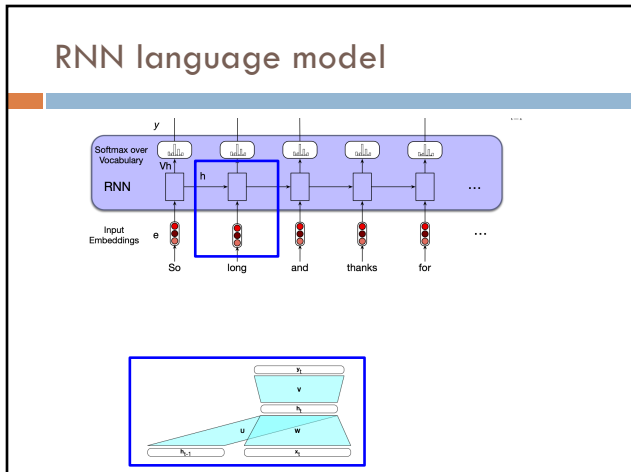
24



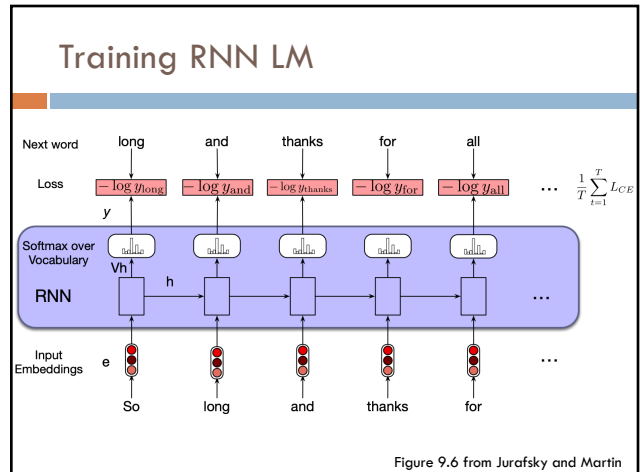
25



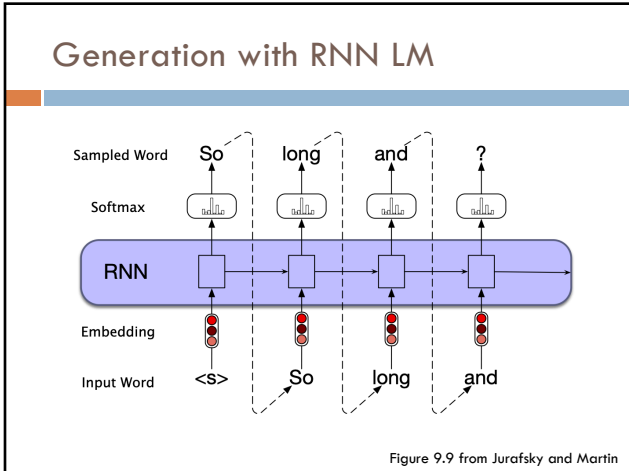
26



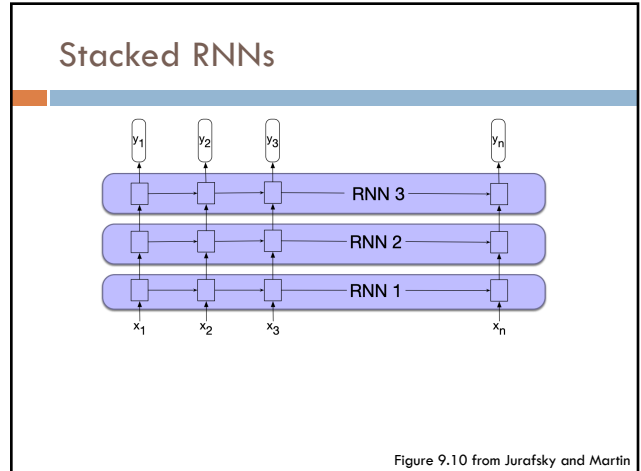
27



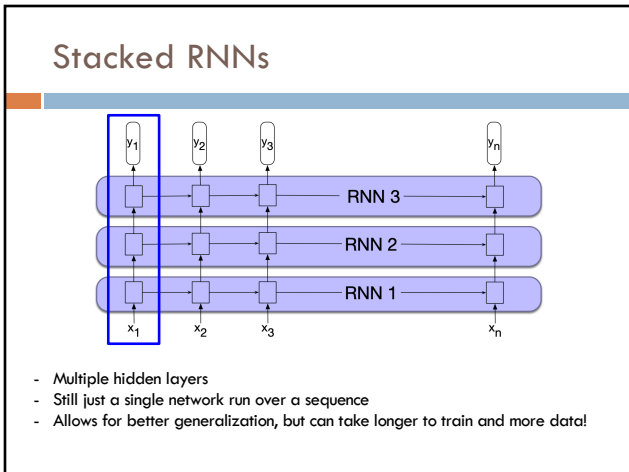
28



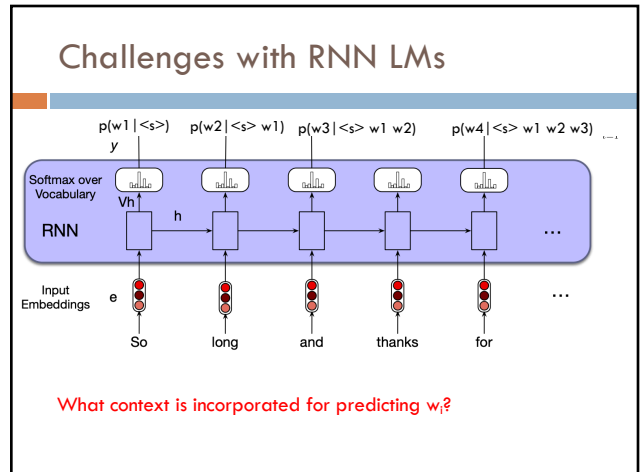
29



30

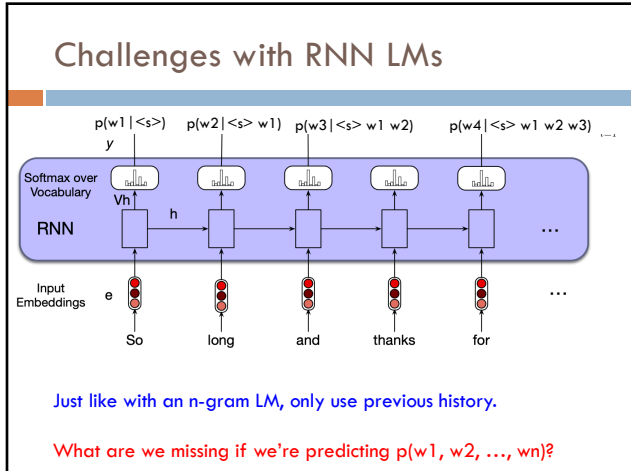


31

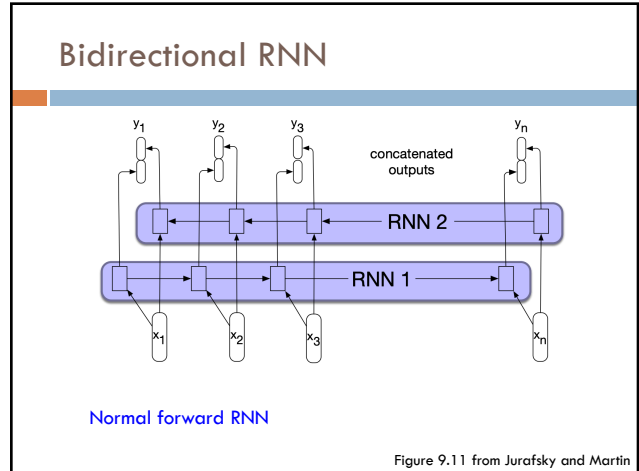


32

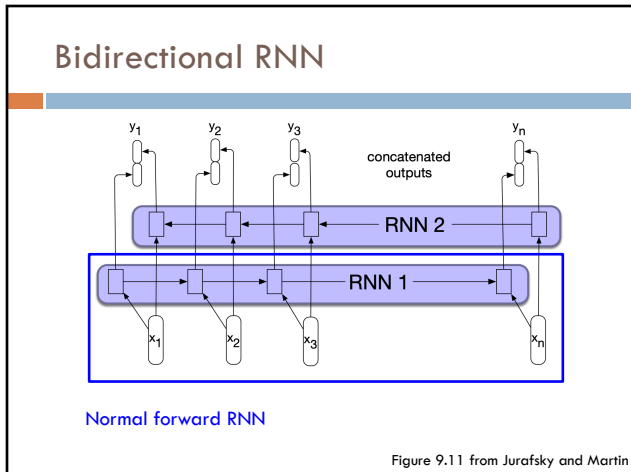




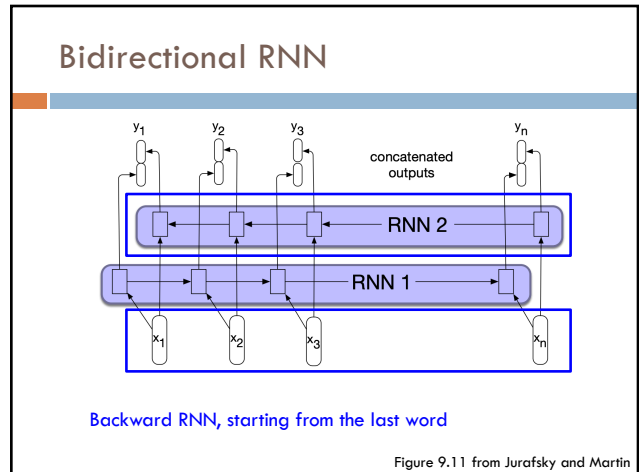
33



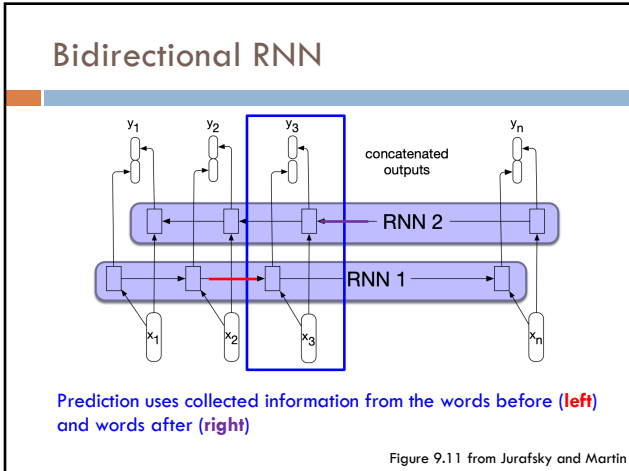
34



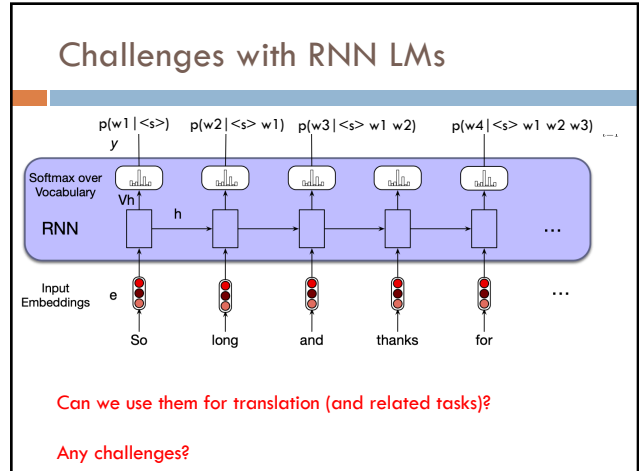
35



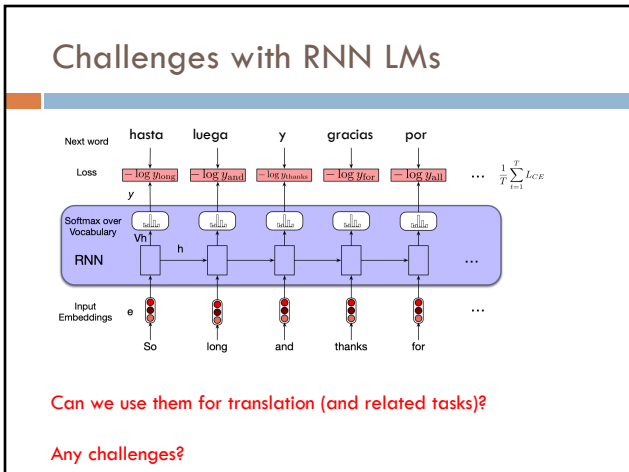
36



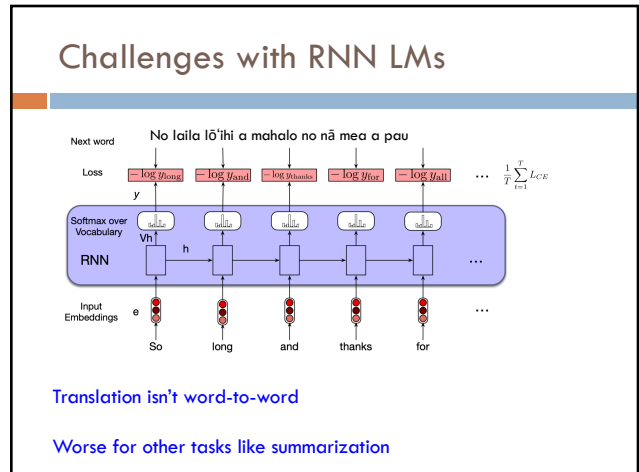
37



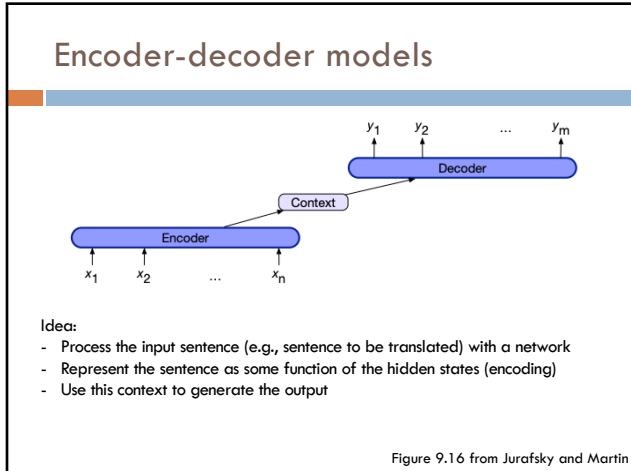
38



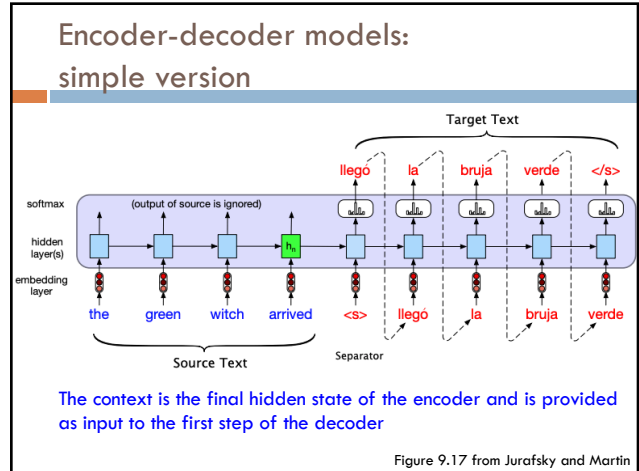
39



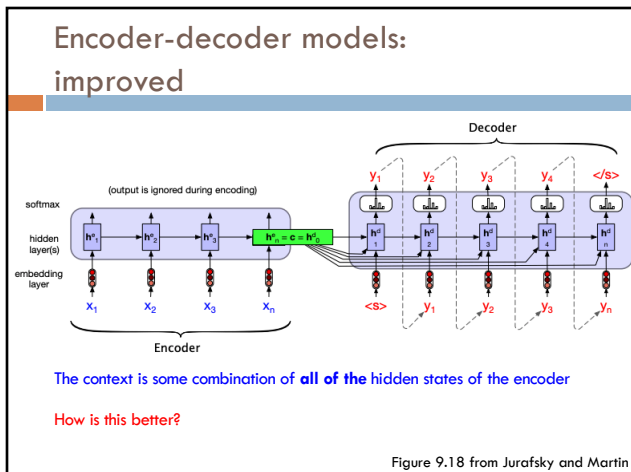
40



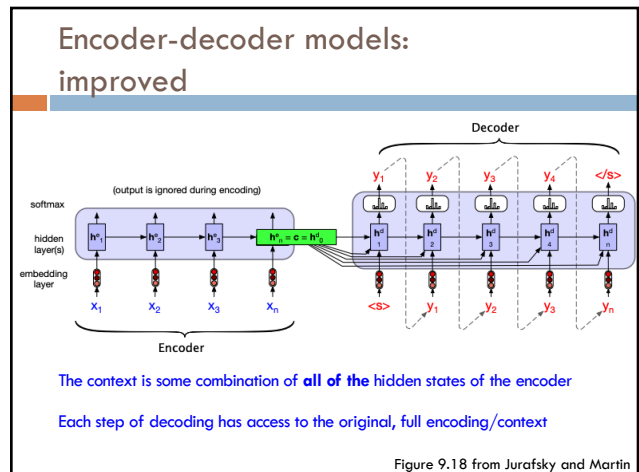
41



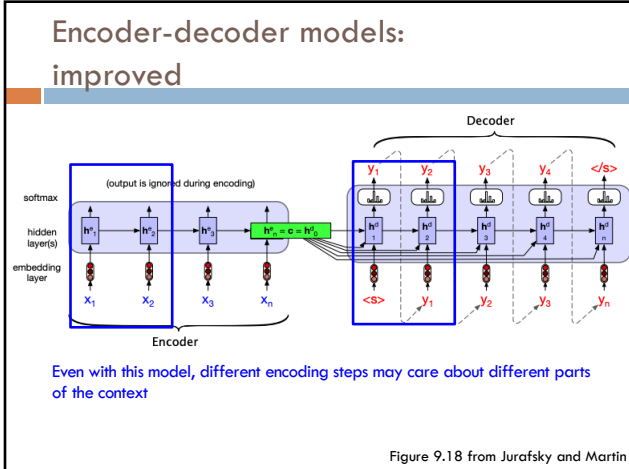
42



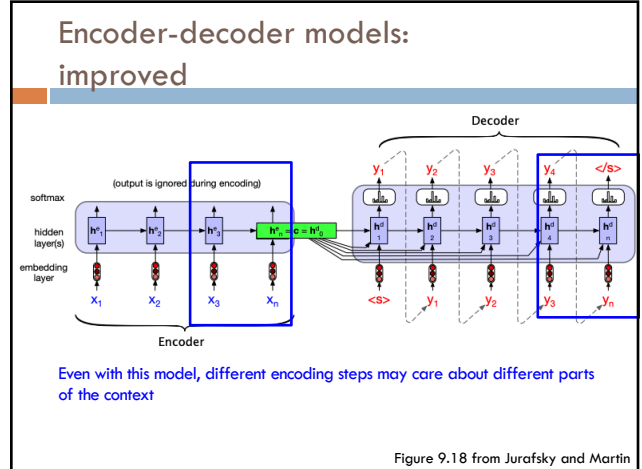
43



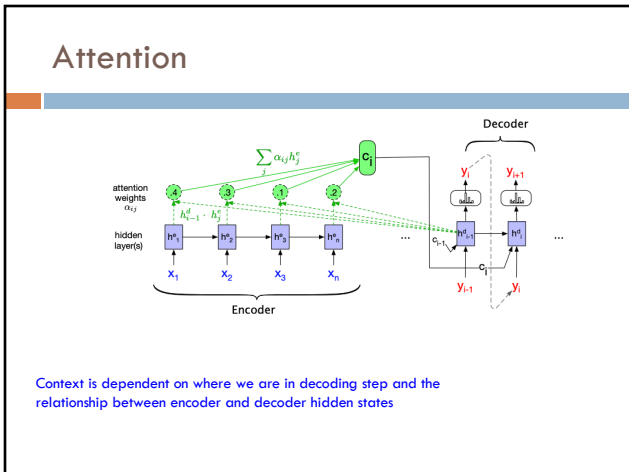
44



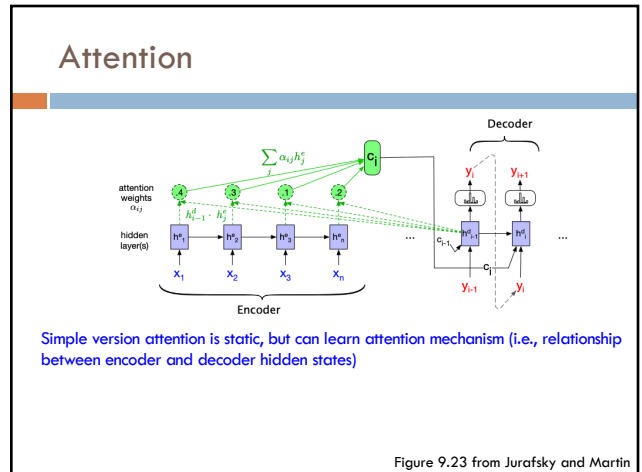
45



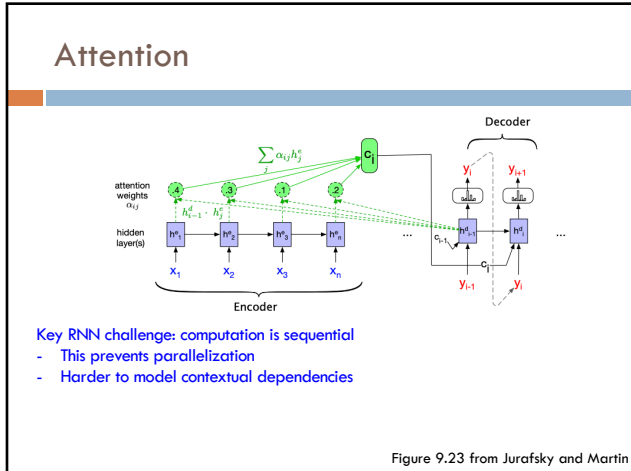
46



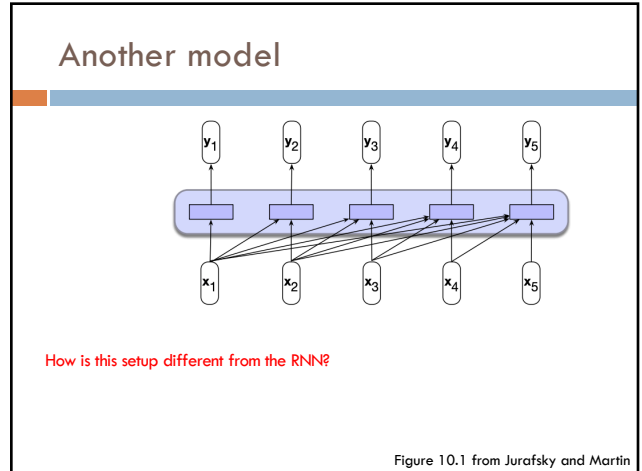
47



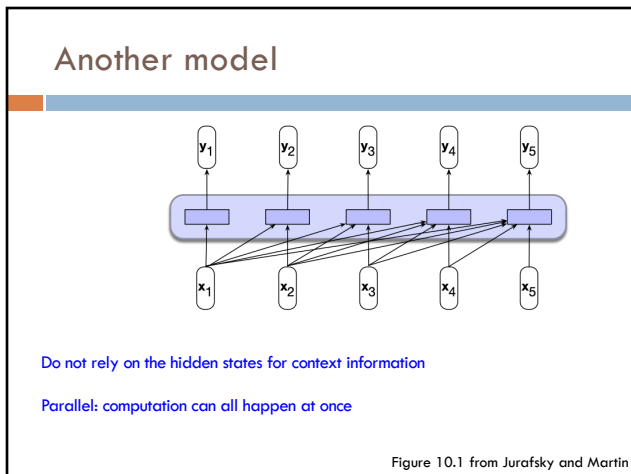
48



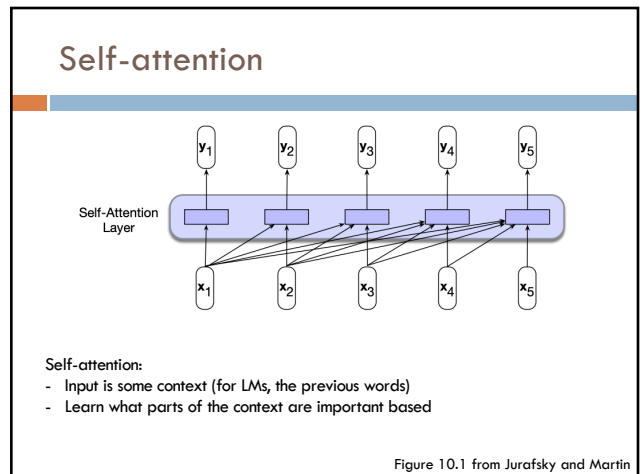
49



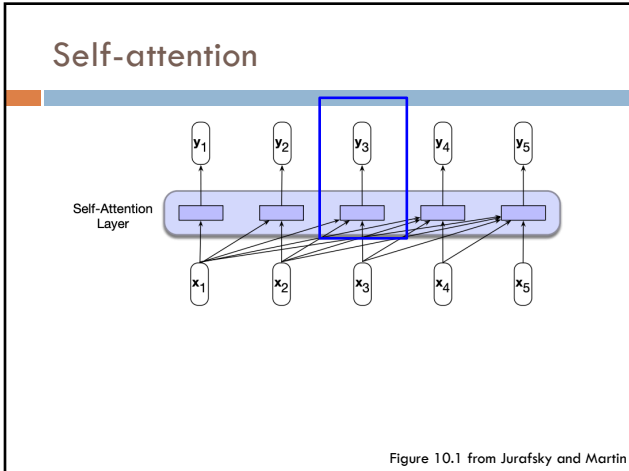
50



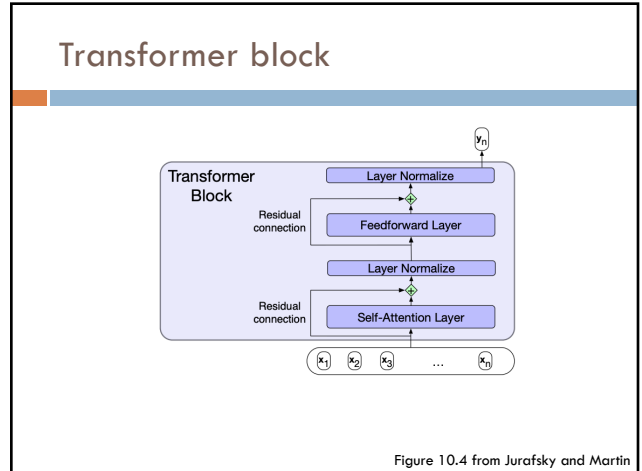
51



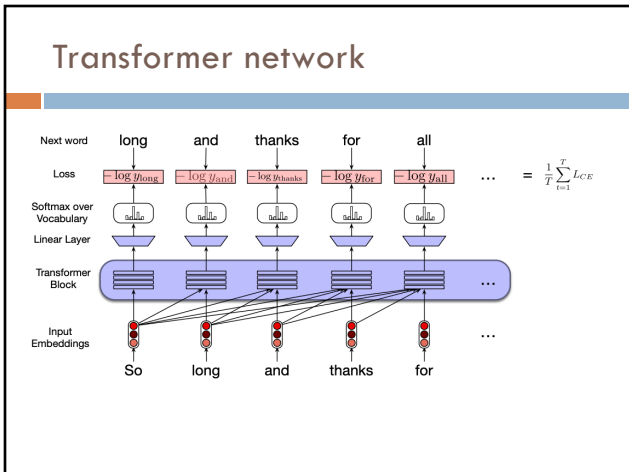
52



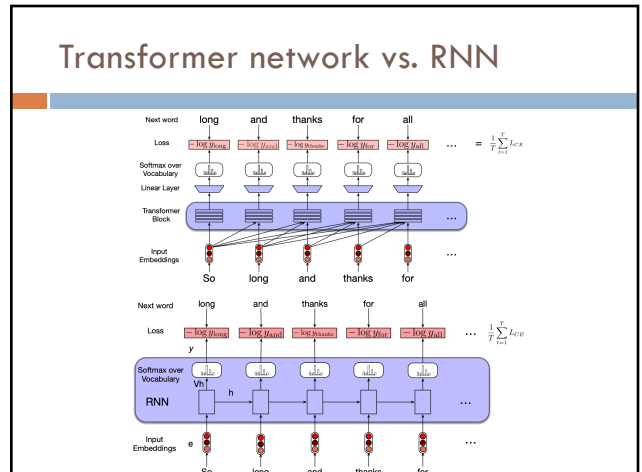
53



54



55



56

## GPT

**Generative:** outputs things

**Pre-trained:** previously trained on a large corpus

**Transformer:** uses the transformer network

57

## Pre-trained language models

Pre-trained language models are general purpose and are trained on a very large corpus

They can be used as/is to:

- Ask  $p(w_1 w_2 \dots w_n)$
- Generate text given some seed,  $p(w_i | w_1 w_2 \dots w_{i-1})$

They can also be “fine-tuned” for particular tasks: take the current weights and update them based on a specific application

58

## ChatGPT

ChatGPT<sup>[6]</sup> is an artificial intelligence (AI) chatbot developed by OpenAI and released in November 2022. It is built on top of OpenAI's GPT-3.5 and GPT-4 families of large language models (LLMs) and has been fine-tuned (an approach to transfer learning) using both supervised and reinforcement learning techniques.

59

## ChatGPT

The fine-tuning process leveraged both supervised learning as well as reinforcement learning in a process called reinforcement learning from human feedback (RLHF).<sup>[7][8]</sup> Both approaches use human trainers to improve the model's performance. In the case of supervised learning, the model was provided with conversations in which the trainers played both sides: the user and the AI assistant. In the reinforcement learning step, human trainers first ranked responses that the model had created in a previous conversation.<sup>[9]</sup> These rankings were used to create "reward models" that were used to fine-tune the model further by using several iterations of Proximal Policy Optimization (PPO).<sup>[7][10]</sup>

60