

Sequence Modeling  
Introduction to RNNs

Deep Neural Network  
Session 18  
Pramod Sharma  
pramod.sharma@prasami.com

2 Agenda


- Sequence Modeling
- Introduction to RNN
- Different Architectures
- Language Modelling
- Image Captioning

11/27/2024

pra-sami

3

## Examples – Sequence Modelling

Domain	Data Type	Output type
Speech Recognition	Audio	Words (text)
Music Creation	Nodes ( $\emptyset$ )	Audio 
Sentiment classification	... an enjoyable one-time-watch for the funny punchlines, far-out characters and performances. But the unconvincing story and the temperate screenplay prevent it from reaching its full potential ...	Integers ( Stars ratings from 1 to 5)
Machine Translation	डीएनएन व्याख्यानमाला आपले स्वागत आहे।	Welcome to DNN Lecture.
Named Entity Recognition	Mohan was driving a Maruti	<b>Mohan</b> was driving a <b>Maruti</b>
Video activity recognition	Sequence of Video Frames	Identify activity say running

11/27/2024

pra-sami

4

## Sequence Modeling – Named Entity Recognition

□ x : Mohan was driving a Maruti

□ y: 1 0 0 0 1

11/27/2024

pra-sami

5

## Sequence Modeling – Named Entity Recognition

□  $x$  : <Mohan Sharma> was driving a <Maruti 800>

□  $y$ : 1 0 0 0 1

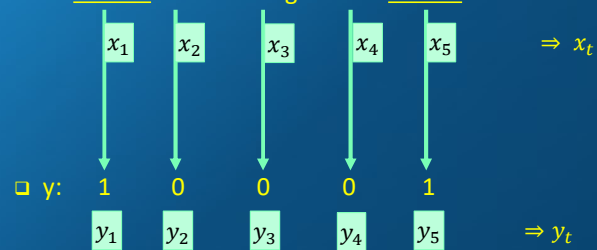
11/27/2024

pra-sâmi

6

## Sequence Modeling – Named Entity Recognition

□  $x$  : Mohan was driving a Maruti



11/27/2024

pra-sâmi

8

## Representing Words

□ Vocabulary = [a, aakash, aamaan... to zulu, zyzzogeton]

- ❖ Also referred as corpus
- ❖ Two more tokens <UNK> and <EOS>

□ Can be converted to one hot encoding

□ x : Mohan was driving a Maruti

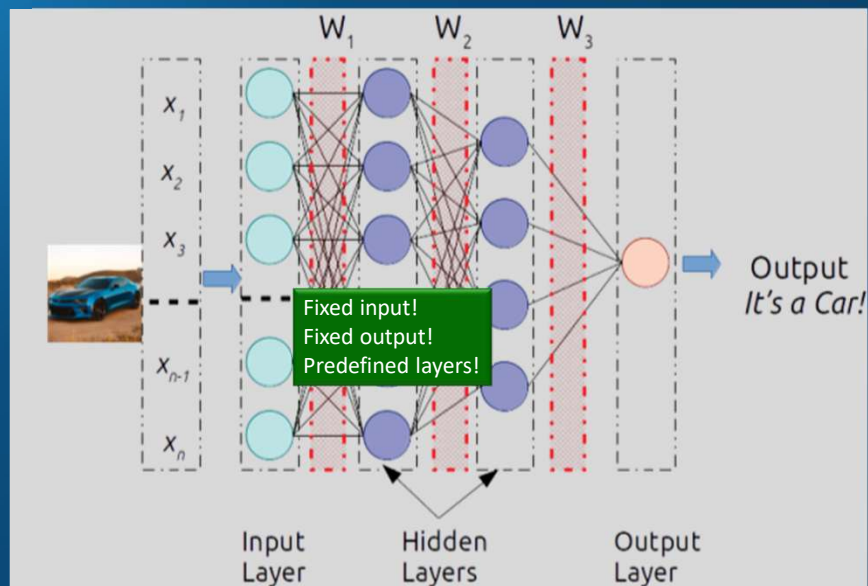
0	0	0	1	0
0	0	0	0	0
—	—	1	—	—
—	—	—	—	—
—	—	—	0	1
1	—	—	—	—
—	1	—	—	—
0	0	0	0	0

11/27/2024

pra-sāmi

9

## Using Standard Architecture



11/27/2024

pra-sāmi

10

## To Summarize....

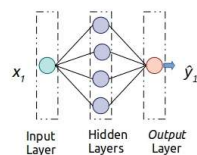
- ❑ Not all problems can be converted into one with fixed length inputs and outputs
- ❑ Problems such as Speech Recognition or Time-series Prediction require a system to store and use context information
- ❑ Hard/Impossible to choose a fixed context window
- ❑ There can always be a new sample longer than anything seen

11/27/2024

pra-samí

11

## What is Recurrent Neural Network...



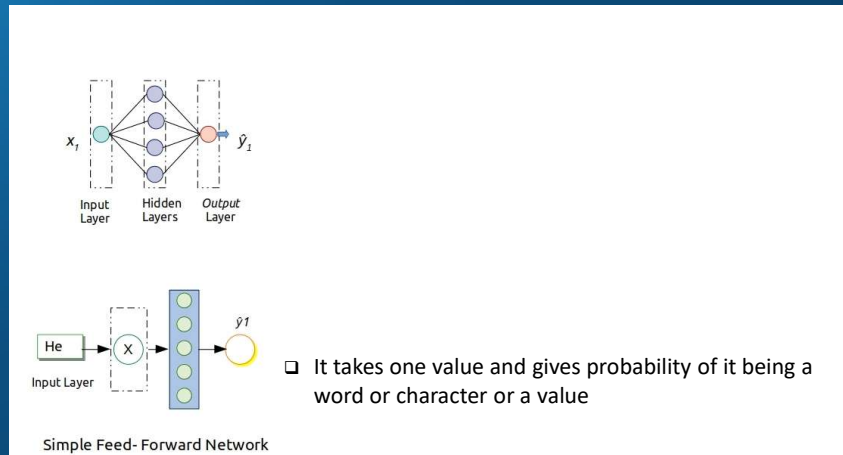
- ❑ Remember our little Neural Network...
- ❑ Let's simplify the layout a little

11/27/2024

pra-samí

12

## What is Recurrent Neural Network...

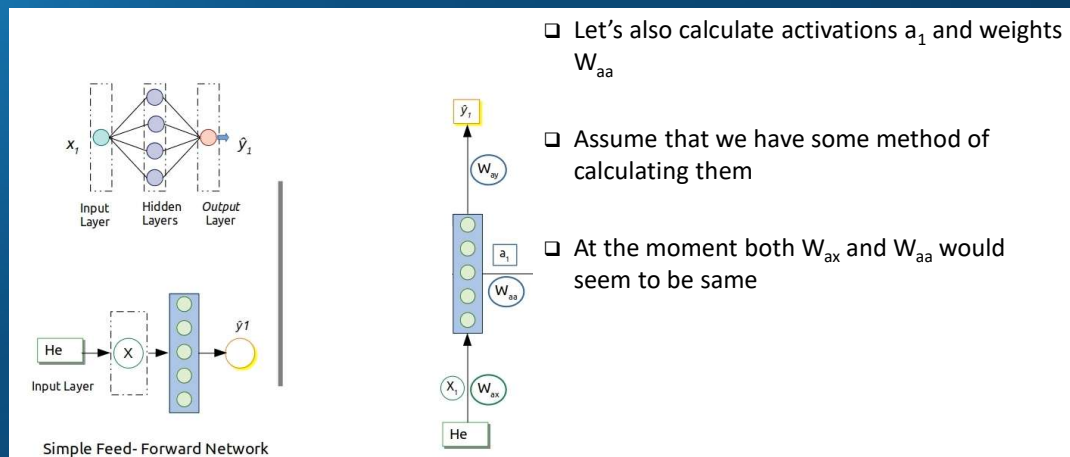


11/27/2024

pra-sami

13

## What is Recurrent Neural Network...

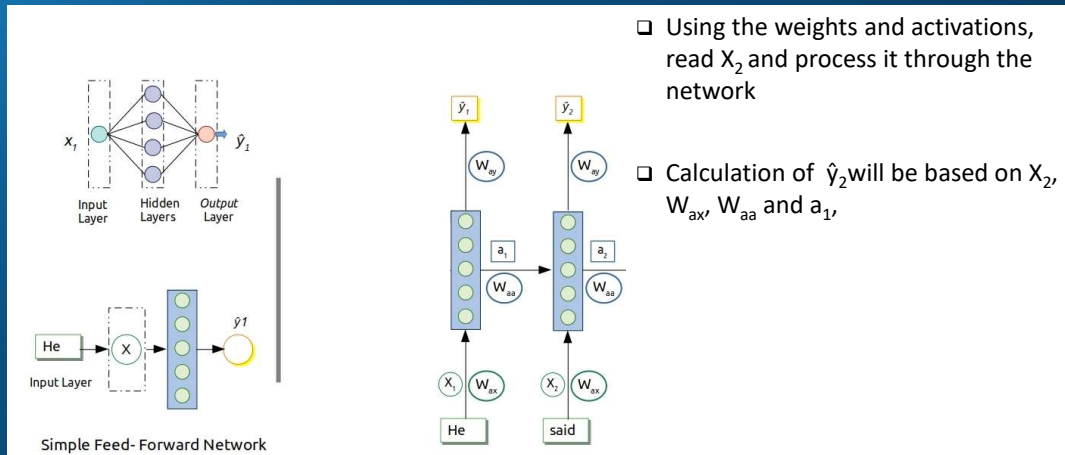


11/27/2024

pra-sami

14

## What is Recurrent Neural Network...

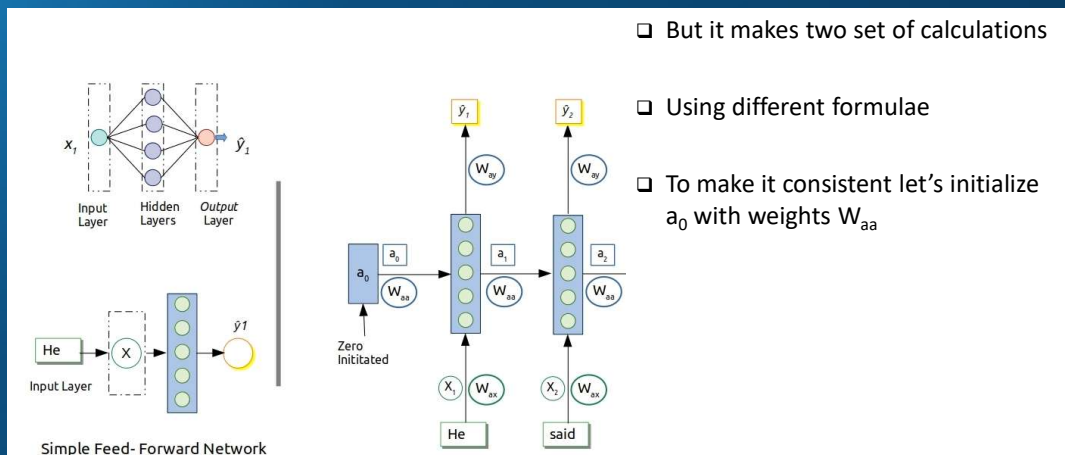


11/27/2024

pra-sami

15

## What is Recurrent Neural Network...

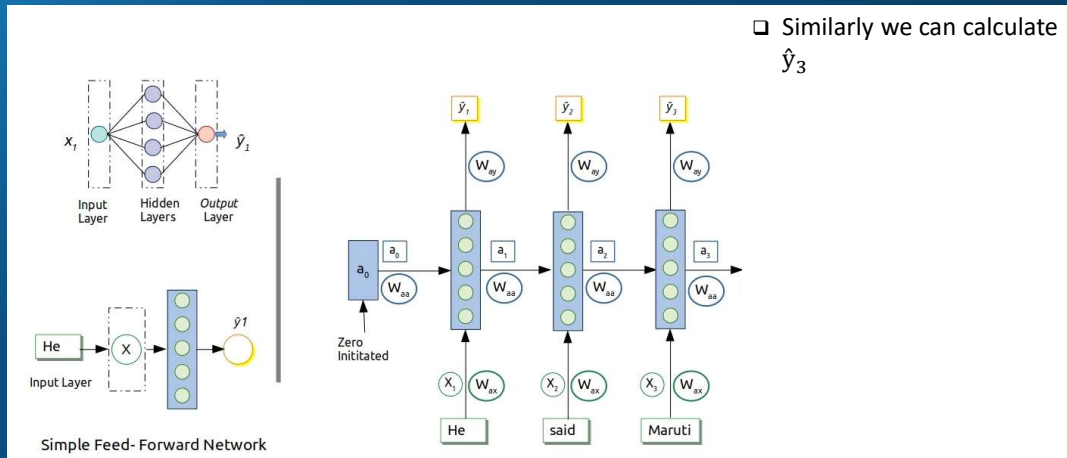


11/27/2024

pra-sami

16

## What is Recurrent Neural Network...

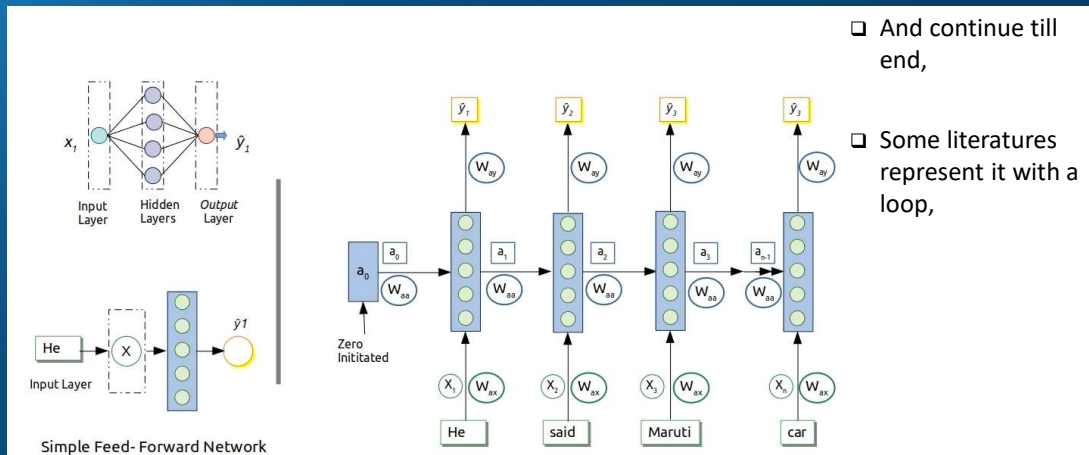


11/27/2024

pra-sami

17

## What is Recurrent Neural Network...



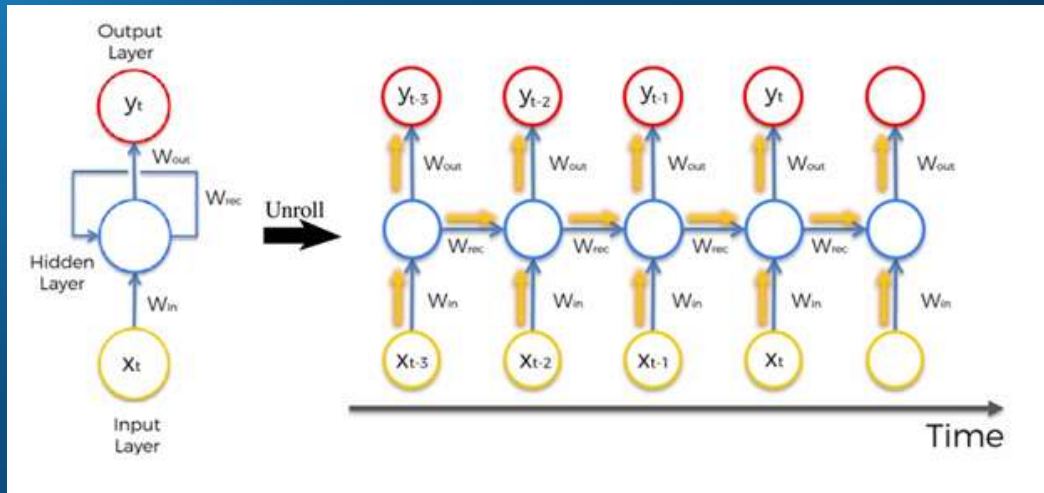
11/27/2024

pra-sami



18

## Alternate Representations

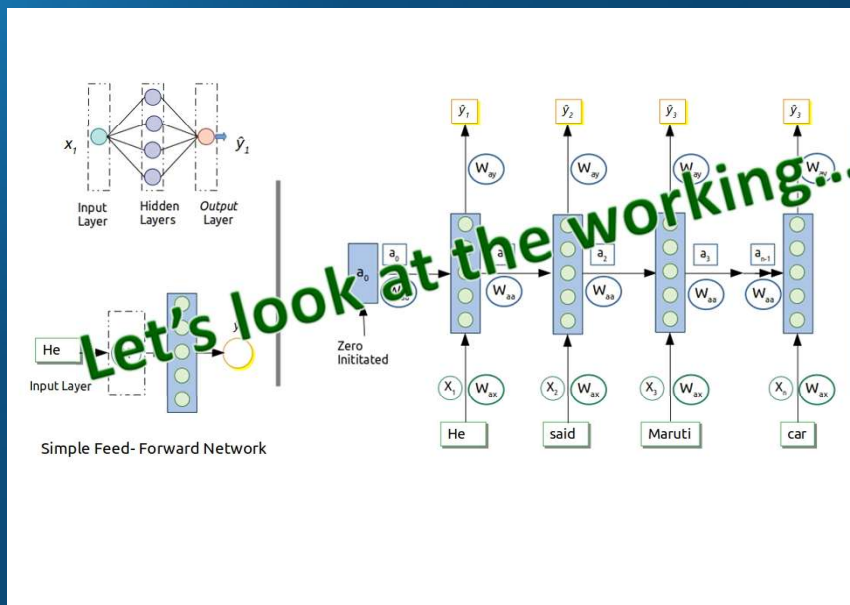


11/27/2024

pra-sami

19

## What is Recurrent Neural Network...

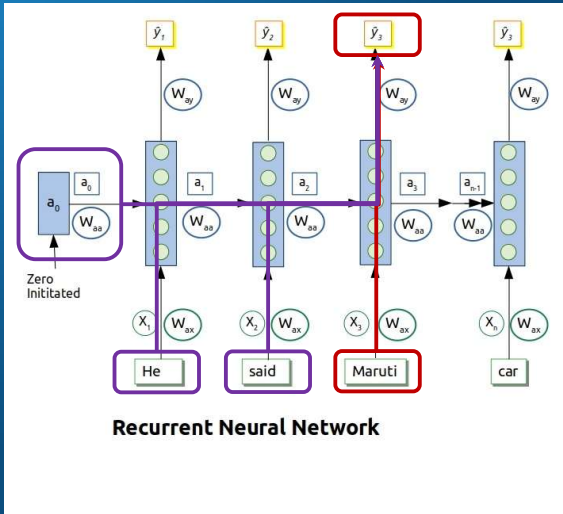


11/27/2024

pra-sami

20

## What is Recurrent Neural Network...



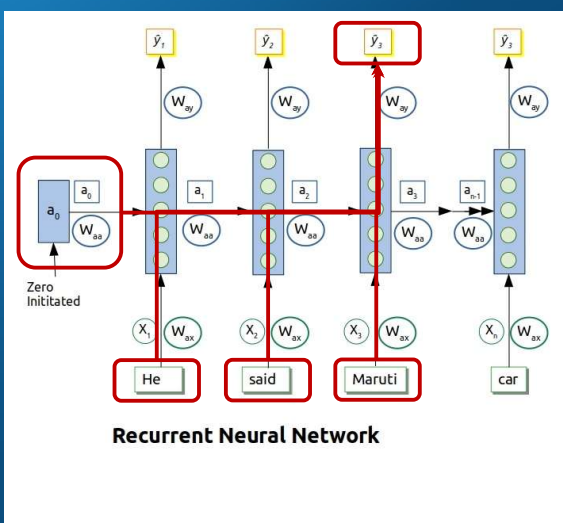
11/27/2024

pra-sami

- Taking activations from previous time step also
- The  $W_{ax}$  and  $W_{aa}$  are shared parameters across all time steps
- So, for calculation of  $\hat{y}_3$  would be influenced by those for  $\hat{y}_2$  and  $\hat{y}_1$

21

## What is Recurrent Neural Network...



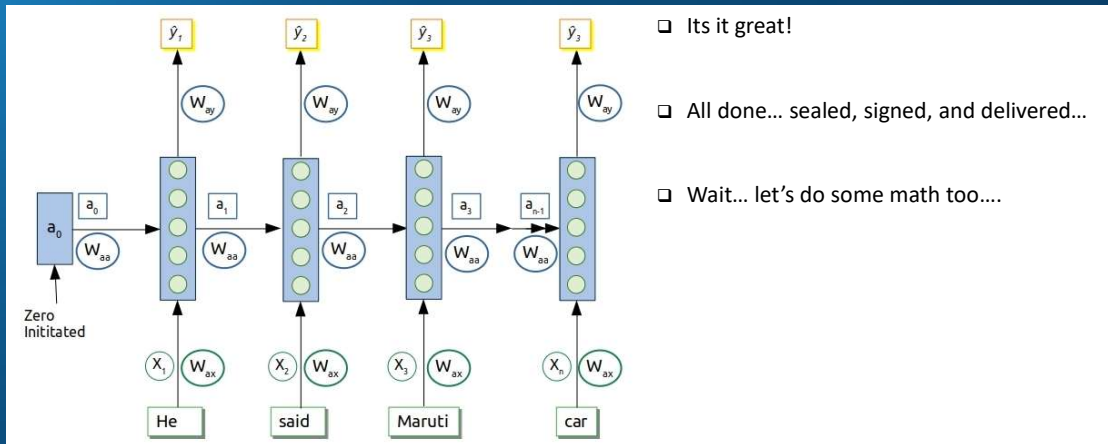
11/27/2024

pra-sami

- It is using the information till time step 3.
  - ❖ He said "Maruti..."
- However, it has no clue what comes next!!!
  - ❖ He said "Maruti is most fuel efficient car"
  - ❖ He said "Maruti is most expensive shop"
  - ❖ He said "Maruti is strongest"

22

## That's is Recurrent Neural Network...



11/27/2024

pra-sâmi

23

## What We Know So Far....

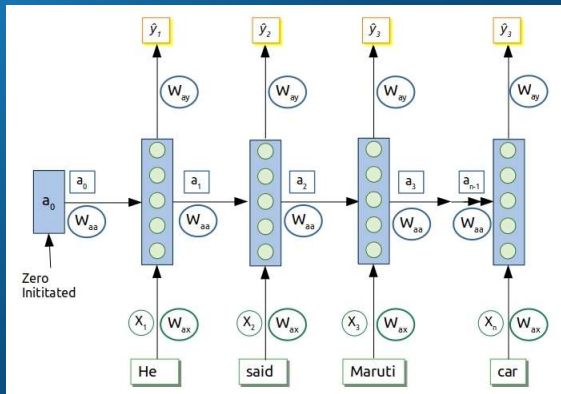
- Recurrent Neural Networks take the previous output or hidden states as inputs.
- The composite input at time 't' has some historical information about the happenings at time ' $T < t$ '.
- RNNs are useful as their intermediate values (state) can store information about past inputs for a time that is not fixed a priori
- Note that the weights are shared over time
- Essentially, copies of the RNN cell are made over time (unrolling/unfolding), with different inputs at different time steps

11/27/2024

pra-sâmi

24

## Forward Propagation



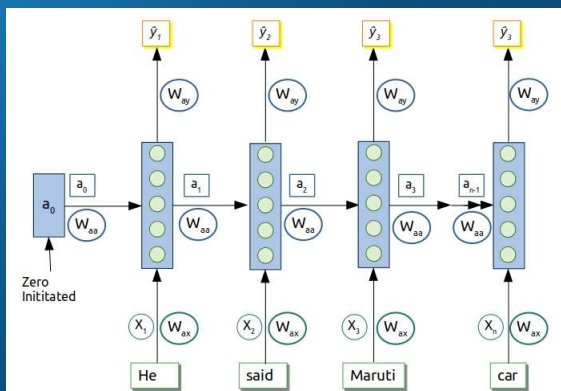
□ Let's work on equations

11/27/2024

pra-sami

25

## Forward Propagation



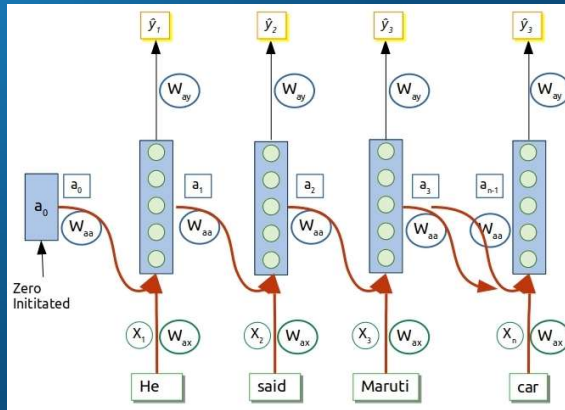
- To start with;  $a_0$  is vector of all zeros
  - ❖  $a_1 = g_1(a_0 \cdot W_{aa} + X_1 \cdot W_{ax} + b_a) \rightarrow \text{Tanh / ReLU}$
  - ❖  $\hat{y}_1 = g_2(a_1 \cdot W_{ay} + b_y) \rightarrow \text{Sigmoid/Softmax (for classification)}$
- Tanh Activation function is more prevalent in RNN
  - ❖ Sometime ReLU too is used
- For output layers, the activation function will depend on type of output
- Generally, at 't' we can write
  - ❖  $a_t = g_1(a_{t-1} \cdot W_{aa} + X_t \cdot W_{ax} + b_a)$
  - ❖  $\hat{y}_t = g_2(a_t \cdot W_{ay} + b_y)$

11/27/2024

pra-sami

26

## Forward Propagation



### Our equations

- ❖  $a_t = g_1(a_{t-1} \cdot W_{aa} + x_t \cdot W_{ax} + b_a)$
- ❖  $\hat{y}_t = g_2(a_t \cdot W_{ay} + b_y)$

### Can be written as:

- ❖  $a_t = g_1([a_{t-1}, x_t] W_a + b_a)$
- ❖  $\hat{y}_t = g_2(a_t \cdot W_y + b_y)$
- ❖ where  $W_a$  will be stacked matrix of  $W_{aa}$  and  $W_{ax}$
- ❖  $W_a = \begin{bmatrix} W_{aa} \\ W_{ax} \end{bmatrix}$
- ❖ Similarly ,
- ❖  $[a_{t-1}, x_t] = [a_{t-1}, | x_t]$

### We know that :

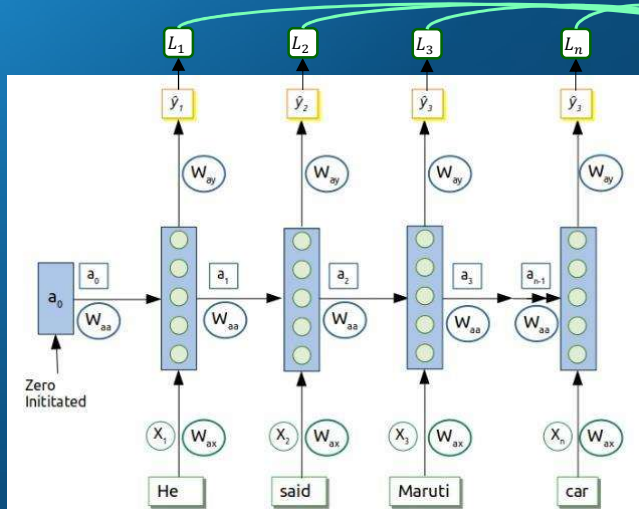
- ❖  $[a_{t-1}, | x_t] \cdot \begin{bmatrix} W_{aa} \\ W_{ax} \end{bmatrix} = a_{t-1} \cdot W_{aa} + x_t \cdot W_{ax}$

11/27/2024

pra-sami

28

## Back Propagation



### At time step 't'; Loss Function for single prediction

- ❖  $L_t(\hat{y}_t, y) = -y_t \cdot \log(\hat{y}_t) - (1 - y_t) \cdot \log(1 - \hat{y}_t)$

### Sum of losses at all time steps:

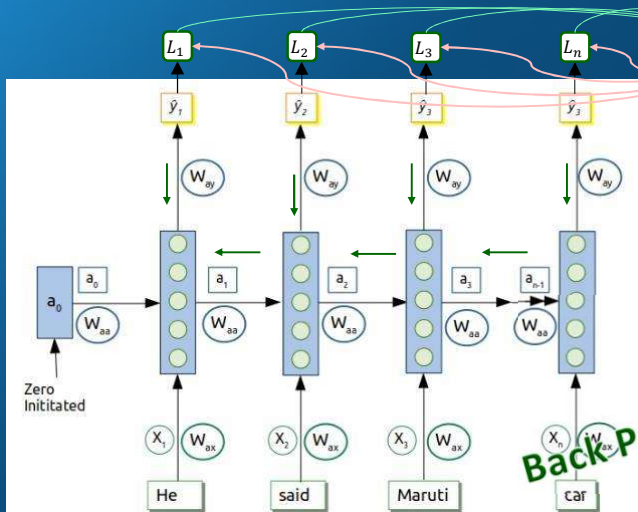
- ❖  $L(\hat{y}, y) = \sum_{t=1}^{T_x} L_t(\hat{y}_t, y)$

11/27/2024

pra-sami

30

## Back Propagation



□ Forward propagation:

$$a_t = g_1([a_{t-1}, x_t] \cdot W_a + b_a)$$

$$\hat{y}_t = g_2(a_t \cdot W_y + b_y)$$

□ Loss Function

$$L_t(\hat{y}, y) = -y_t \cdot \log(\hat{y}_t) - (1 - y_t) \cdot \log(1 - \hat{y}_t)$$

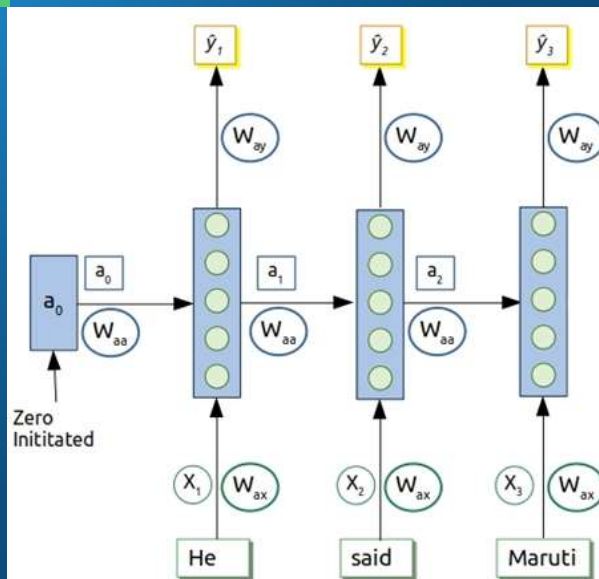
Back Propagation through Time.

11/27/2024

pra-sami

31

## Back Propagation Through Time...



Forward propagation:

$$a_t = g_1([a_{t-1}, x_t] \cdot W_a + b_a)$$

$$\hat{y}_t = g_2(a_t \cdot W_y + b_y)$$

Loss Function :

$$L_t(\hat{y}, y) = -y_t \cdot \log(\hat{y}_t) - (1 - y_t) \cdot \log(1 - \hat{y}_t)$$

Step 3:

$$\frac{dL_3}{dw_y} = \frac{dL_3}{d\hat{y}_3} \cdot \frac{d\hat{y}_3}{dw_y}$$

$$\begin{aligned} \frac{dL_3}{dw_a} &= \frac{dL_3}{d\hat{y}_3} \cdot \frac{d\hat{y}_3}{da_3} \cdot \frac{da_3}{dw_a} \\ &+ \frac{dL_3}{d\hat{y}_3} \cdot \frac{d\hat{y}_3}{da_3} \cdot \frac{da_3}{da_2} \cdot \frac{da_2}{dw_a} \\ &+ \frac{dL_3}{d\hat{y}_3} \cdot \frac{d\hat{y}_3}{da_3} \cdot \frac{da_3}{da_2} \cdot \frac{da_2}{da_1} \cdot \frac{da_1}{dw_a} \end{aligned}$$

There is a pattern here!

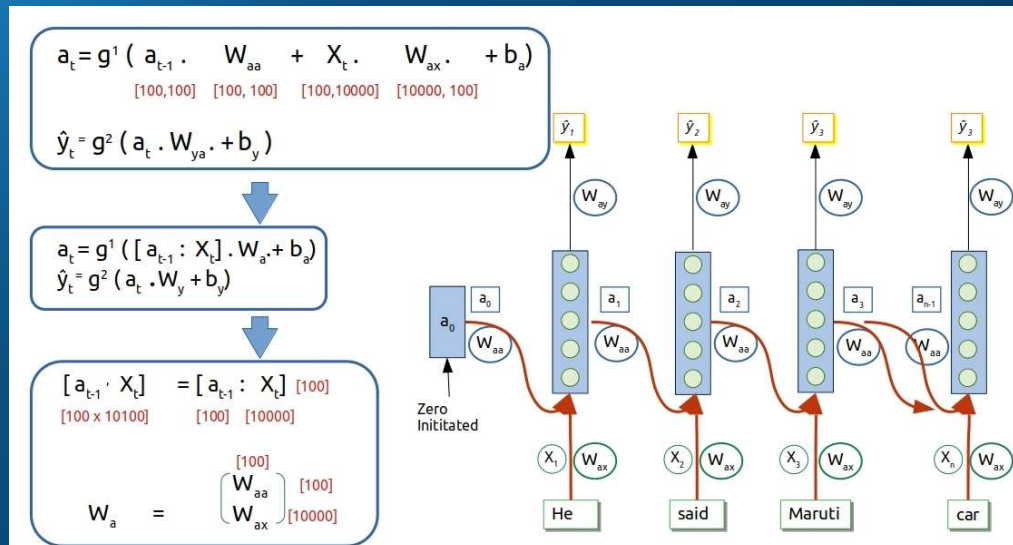
$$\begin{aligned} \frac{dL_3}{dw_x} &= \frac{dL_3}{d\hat{y}_3} \cdot \frac{d\hat{y}_3}{da_3} \cdot \frac{da_3}{dw_x} \\ &+ \frac{dL_3}{d\hat{y}_3} \cdot \frac{d\hat{y}_3}{da_3} \cdot \frac{da_3}{da_2} \cdot \frac{da_2}{dw_x} \\ &+ \frac{dL_3}{d\hat{y}_3} \cdot \frac{d\hat{y}_3}{da_3} \cdot \frac{da_3}{da_2} \cdot \frac{da_2}{da_1} \cdot \frac{da_1}{dw_x} \end{aligned}$$

11/27/2024

pra-sami

32

## Quickly check the dimension....

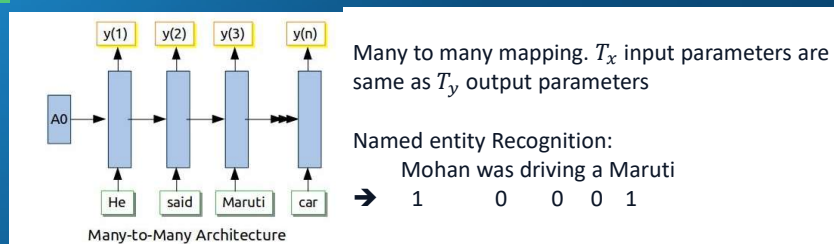


11/27/2024

pra-sami

33

## Type of Architectures



The Unreasonable Effectiveness of Recurrent Neural Networks  
- Andrej Karpathy

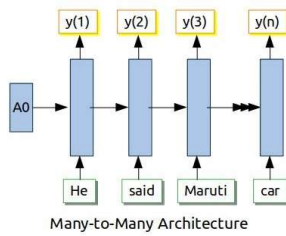
11/27/2024

pra-sami

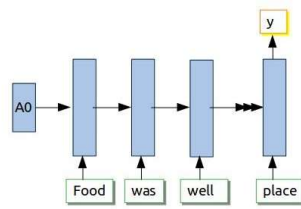


34

## Type of Architectures



Many-to-Many Architecture



Many-to-One Architecture

Many to one architecture.

Input is the 'review' written by a patron  
and output is an integer (star rating)

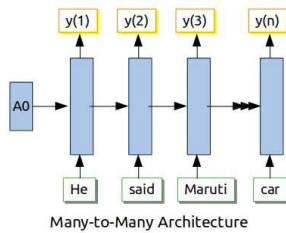
The Unreasonable Effectiveness of Recurrent Neural Networks  
- Andrej Karpathy

11/27/2024

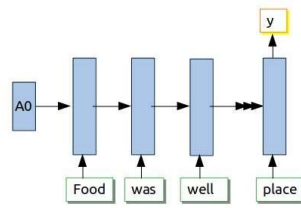
pra-sami

35

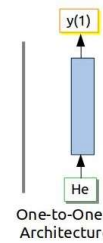
## Type of Architectures



Many-to-Many Architecture



Many-to-One Architecture

One-to-One  
Architecture

Of course there is one  
to one. i.e. Basic neural  
network...

The Unreasonable Effectiveness of Recurrent Neural Networks  
- Andrej Karpathy

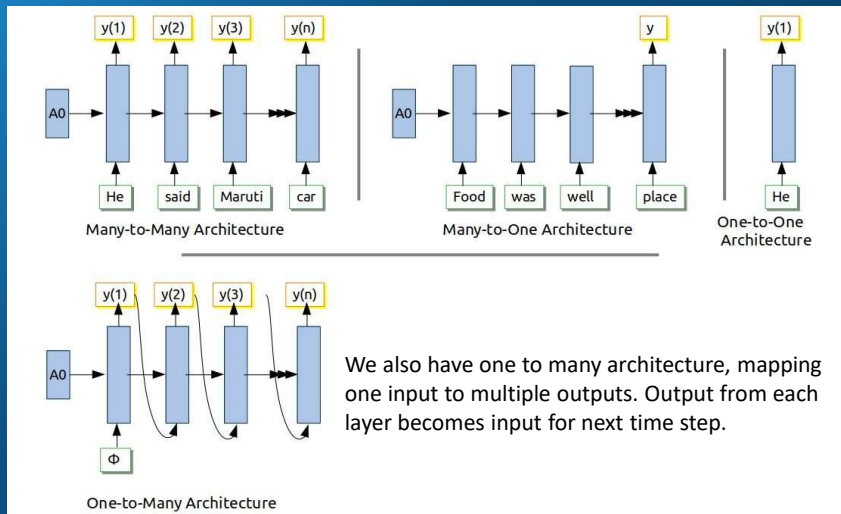
11/27/2024

pra-sami



36

## Type of Architectures



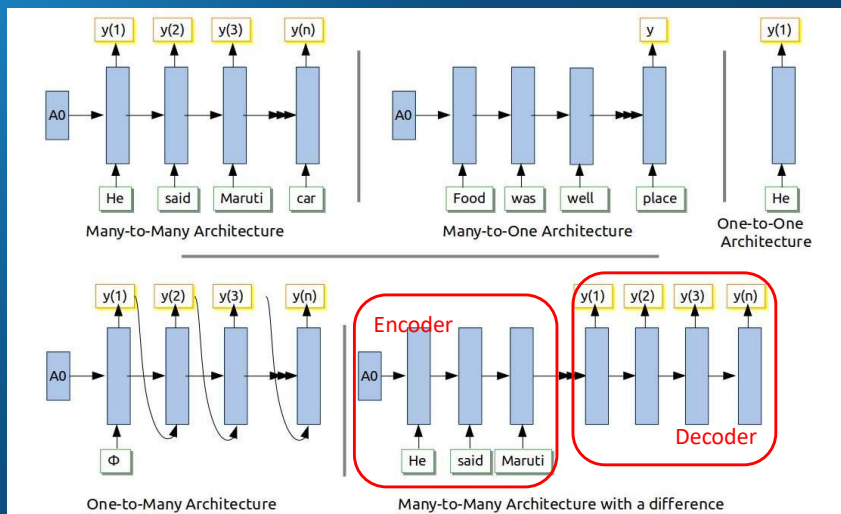
The Unreasonable Effectiveness of Recurrent Neural Networks  
- Andrej Karpathy

11/27/2024

pra-sami

37

## Type of Architectures



The Unreasonable Effectiveness of Recurrent Neural Networks  
- Andrej Karpathy

11/27/2024

pra-sami

डीएनएन व्याख्यानमाला आपले स्वागत आहे।

➔ Welcome to DNN Lecture

In this Architecture, we have two completely different parts. One side reading sentences in one language, and other side translating in different language.

We can have  $T_x$  and  $T_y$  different which is a case in machine translations

38

## Language Modelling

### Speech Recognition

- ❑ Toad met Pit....
- ❑ Todd met Pete...
- ❑ Given any sentence, what is the probability of that being a valid sentence
- ❑ So what language model would do is to calculate probability of a sentence with that combination of words
  - ❖  $P(\text{Toad met Pit}) = 4.6 \times 10^{-15}$
  - ❖  $P(\text{Todd met Pete}) = 9.3 \times 10^{-9}$
- ❑ Mathematically  $P(\text{sentence}) = P(y_1, y_2, y_3, \dots y_n)$

11/27/2024

pra-sâmi

39

## How to Model?

- ❑ Training set : Large corpus of English text
  - ❖ Adults need eight hours of sleep a day!

Adults	need	eight	hours	of	sleep	a	day	↓	<EOS>
$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$y_6$	$y_7$	$y_8$	-	$y_9$

- ❑ First step is to tokenize the sentence
- ❑ Add a token at end and at the beginning <EOS> ( $y_9$ )
- ❑ Remember we have limited tokens (say we only have 10,000 tokens).
- ❑ Unknown words will be given a token <unk>

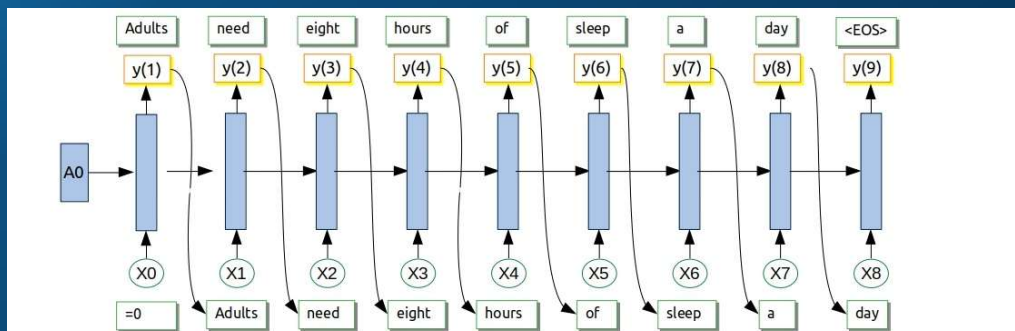
11/27/2024

pra-sâmi

40

## RNN Model

- At the onset RNN tries to predict probabilities of each word in the corpus of being first word in this sentence.
- i.e.  $P[a]$ ,  $P[aakash]$ ,  $P[aamaan]$ ... to  $P[zulu]$ ,  $P[zyzzogeton]$ 
  - ❖ This would be an array of 10002 elements



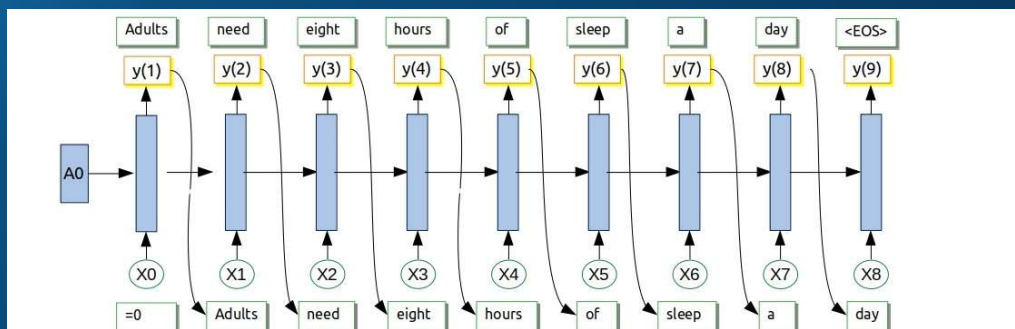
11/27/2024

pra-sami

41

## RNN Model

- Thus we can calculate error between  $\hat{y}_1$  and "Adults"
- Given first word "Adults", again RNN predicts the probabilities for second word, thus combined probability, and it continues...
  - ❖ i.e.  $P[a | \text{Adult}]$ ,  $P[aakash | \text{Adult}]$ ,  $P[aamaan | \text{Adult}]$ ... to  $P[zulu | \text{Adult}]$ ,  $P[zyzzogeton | \text{Adult}]$
- Somewhere in that bunch there will be a probability  $P[\text{need} | \text{Adult}]$



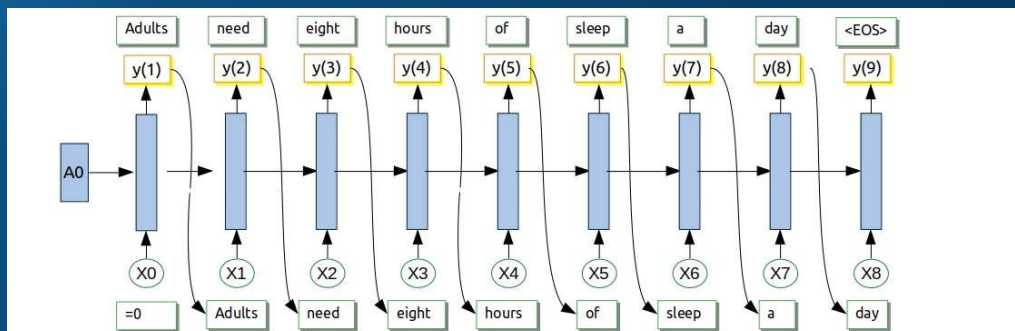
11/27/2024

pra-sami

42

## RNN Model

- At third step we can calculate error between  $\hat{y}_2$  and "need".
- Given first word "Adults", and second word as "need", again RNN predicts the probabilities for third word
- i.e.  $P[a \mid \text{Adult, need}]$ ,  $P[\text{aakash} \mid \text{Adult, need}]$ ,  $P[\text{aamaan} \mid \text{Adult, need}]$ ... to  $P[\text{zulu} \mid \text{Adult, need}]$ ,  $P[\text{zyzzogeton} \mid \text{Adult, need}]$
- Somewhere in that bunch there will be a probability  $P[\text{eight} \mid \text{Adult, need}]$



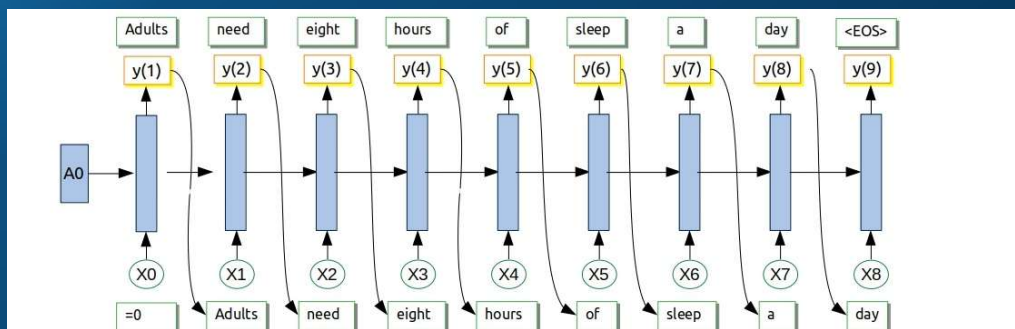
11/27/2024

pra-sami

43

## RNN Model

- Thus we can calculate error between  $\hat{y}_3$  and "eight".
- It continues from left to right till end,  $X_8$
- Given all previous words, what is the probability of this word being <EOS>.



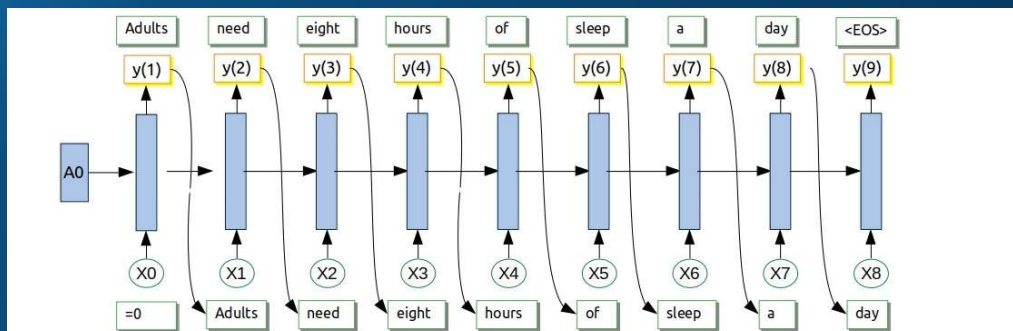
11/27/2024

pra-sami

44

## RNN Model

- RNN is trying to predict one word at a time from left to right.
- Given that we are going to use logits and subsequently softmax for loss function, our loss function will be
- $\ell(\hat{y}, y) = -y * \log(\hat{y})$  as  $\hat{y}$  is very close to 0 for all other words
  - ❖ since its remaining part  $[(1 - y) * \log(1 - \hat{y})]$  is insignificantly small we can ignore it.



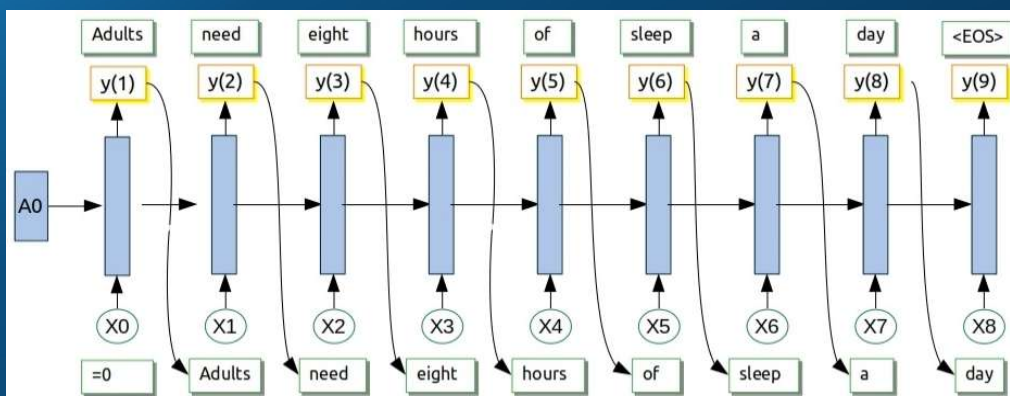
11/27/2024

pra-sami

45

## RNN Model

- Thus for overall sentence, Cost will be
  - ❖  $J(\hat{y}, y) = \sum \ell(\hat{y}, y)$
  - ❖  $J(\hat{y}, y) = -\frac{1}{m} \sum y * \log(\hat{y})$
  - ❖ Which we will be minimizing.



11/27/2024

pra-sami

46

## RNN Model

- Suppose you have sentence with 3 words
- You want to know probability of it being a sentence
- Given a sentence  $y_1, y_2, y_3$
- $P(y_1, y_2, y_3) = P[y_1] * P[y_2 | y_1] * P[y_3 | y_1, y_2]$

11/27/2024

pra-sāmi

47

## Word representation

- Vocabulary = [a, aakash, aamaan... to zulu, zyzzogeton]
  - ❖ Also referred as corpus
  - ❖ Two more tokens <UNK> and <EOS>
- Can be converted to one hot encoding

Man (5468)	Women (8701)	King (4823)	Queen (7157)	Apple (56)	Oranges (7259)
0	0	0	0	0	0
0	0	0	0	1	0
—	—	1	—	0	—
—	—	—	—	—	—
□ 1	—	—	—	—	—
—	—	—	1	—	1
—	1	—	—	—	—
—	—	—	—	—	—
0	0	0	0	0	0

11/27/2024

pra-sāmi

This representation is treating words independently....

48

## Featured Representation

	Man (5468)	Women (8701)	King (4823)	Queen (7157)	Apple (56)	Oranges (7259)
Gender	-1	1	-0.95	0.97	0	0.001
Royal	0.01	0.02	0.90	0.98	0.05	-0.01
Age	0.05	0.02	0.7	0.68	0.001	-0.4
Food	0.001	0.002	0.0001	0.0002	0.95	0.90

Feature representing a huge corpus can drastically be reduced...

□ Man → Women  $\approx$  King → ???

□ In terms of algorithm, we can use this using Similarity Coefficients

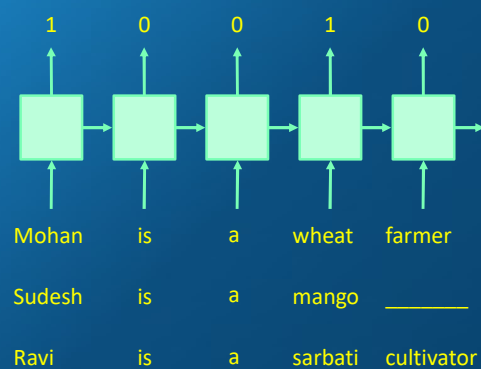
- ❖ Find a word  $W$  :  $\text{argmax} (e_w, e_{\text{king}} - e_{\text{man}} + e_{\text{women}})$
- ❖ Cosine sim (  $u, v$  ) =  $\frac{(u^T \cdot v)}{\|u\|_2 \cdot \|v\|_2}$
- ❖ Euclidian distances or Manhattan distances can also be used

11/27/2024

pra-sāmi

49

## Named entity and word embedding



Words → Embedding Layer → Dense Layer → Softmax Layer  
 $W_d, b_d$        $W_o, b_o$   
 5 words      5 x 300      10000 probabilities

11/27/2024

pra-sāmi

50

## Sampling a Sequence from a Well Trained Model

- ❑ Imagine we have super trained RNN network
- ❑ We ask it to predict first word,
  - ✦ which results in probability words in corpus to be first word,
- ❑ Pick a word from the probabilities to be first word (`np.random.choice()`)
- ❑ Enter this word as input to timestamp '2' to generate second word, again pick a word at random and pass it to third time stamp.
- ❑ and you will generate a sentence till you reach a <EOS>
- ❑ Alternatively, you can limit the sentence to say 20 words
  
- ❑ Voila!!!
  
- ❑ Remember 2016 US Election, someone fabricated how Trump would have answered questions during press conference
- ❑ Obviously it would not make exact sense. But in general it will be same.

11/27/2024

pra-sâmi

51

## RNN Model

- ❑ In some cases, it is advantageous to have character based RNN instead of word based RNN.
- ❑ Both formats have their own advantages.

11/27/2024

pra-sâmi



52

## Sequence to sequence : Image Captioning

- ❑ Given an image, produce a sentence describing its contents
- ❑ Inputs: Image feature (from a CNN)
- ❑ Outputs: Multiple words (let's consider one sentence)



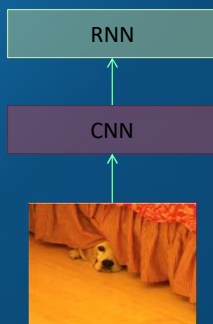
: The dog is hiding

11/27/2024

pra-sâmi

53

## Image Captioning

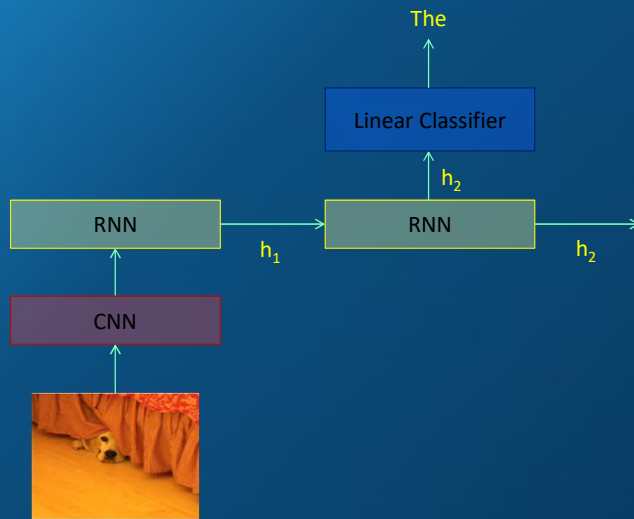


11/27/2024

pra-sâmi

54

## Image Captioning

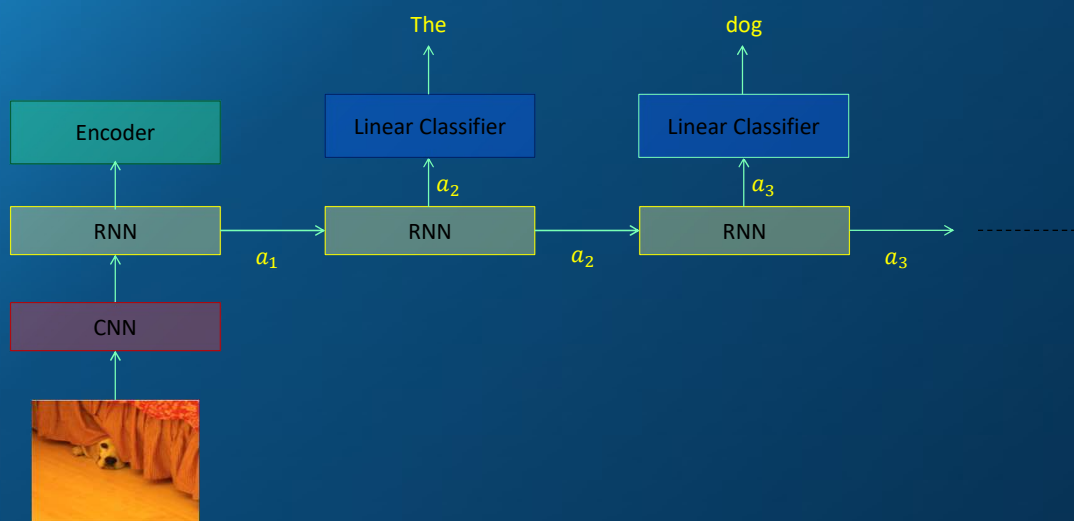


11/27/2024

pra-sami

55

## Image Captioning

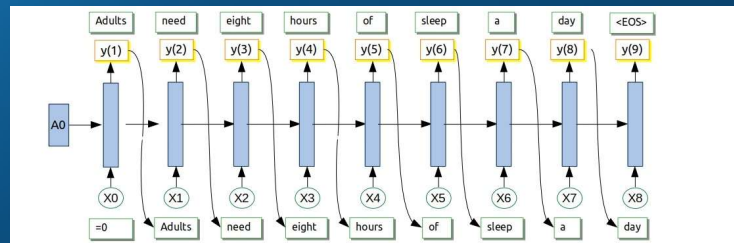


11/27/2024

pra-sami

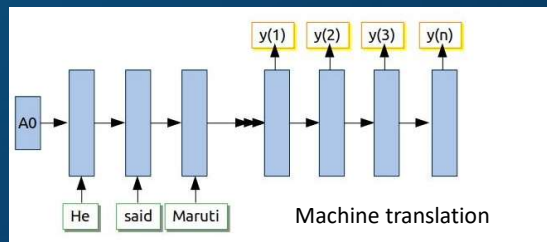
56

## Machine translation



Language Model

### Conditional language Model



Machine translation

11/27/2024

pra-sami

57

## Sequence to sequence : Bleu Score

- 'Dog', 'bed', 'hiding'
- Le chien est sous le lit
- कुत्ता बिस्तर के नीचे है.
- कुत्ता पलंगाच्या खाली आहे.



: The dog is hiding

- Reference 1: The Dog is hiding under the bed
- Reference 2: There is a dog under the bed
- MT Output : The dog the dog hiding under the bed

"BLEU: a Method for Automatic Evaluation of Machine Translation" By [Kishore Papineni](#), [Salim Roukos](#), [Todd Ward](#), [Wei Jing Zhu](#).

11/27/2024

pra-sami

58

## RNN Outputs: Image Captions



[Show and Tell: A Neural Image Caption Generator, CVPR 15](#)

11/27/2024

pra-sâmi

59

## Reflect...

- ❑ What is the key difference between Recurrent Neural Networks (RNNs) and Feedforward Neural Networks?
  - ❖ A) RNNs use activation functions
  - ❖ B) RNNs have cycles and feedback loops in their connections
  - ❖ C) RNNs cannot handle sequential data
  - ❖ D) RNNs use more neurons than Feedforward Networks
- ❑ Answer: B) RNNs have cycles and feedback loops in their connections
- ❑ Which of the following is a typical application of RNNs?
  - ❖ A) Image classification
  - ❖ B) Sentiment analysis of text
  - ❖ C) Object detection
  - ❖ D) Image segmentation
- ❑ Answer: B) Sentiment analysis of text
- ❑ What type of data is RNN most suitable for?
  - ❖ A) Tabular data
  - ❖ B) Sequential data like time series or text
  - ❖ C) Randomly ordered data
  - ❖ D) Static data like images
- ❑ Answer: B) Sequential data like time series or text
- ❑ What is the problem of vanishing gradients in RNNs?
  - ❖ A) The model grows too large over time
  - ❖ B) The gradients used for backpropagation become very small, making learning slow or ineffective
  - ❖ C) The model loses information about long sequences
  - ❖ D) The model overfits due to too much data
- ❑ Answer: B) The gradients used for backpropagation become very small, making learning slow or ineffective

11/27/2024

pra-sâmi

60

## Reflect...

- ❑ Which of the following is an RNN variant designed for processing sequences in both directions?
  - ❖ A) Unidirectional RNN
  - ❖ B) Bidirectional RNN
  - ❖ C) Convolutional Neural Network (CNN)
  - ❖ D) Autoencoder
- ❑ Answer: B) Bidirectional RNN
- ❑ What are the three types of gates used in LSTM networks?
  - ❖ A) Input gate, Output gate, Forget gate
  - ❖ B) Memory gate, Activation gate, Forget gate
  - ❖ C) Input gate, Reset gate, Output gate
  - ❖ D) Update gate, Forget gate, Reset gate
- ❑ Answer: A) Input gate, Output gate, Forget gate
- ❑ Why are Gated Recurrent Units (GRUs) considered simpler than LSTMs?
  - ❖ A) GRUs use a single gate instead of multiple gates
  - ❖ B) GRUs combine the forget and input gates into a single update gate
  - ❖ C) GRUs do not have a hidden state
  - ❖ D) GRUs do not require backpropagation
- ❑ Answer: B) GRUs combine the forget and input gates into a single update gate
- ❑ Which loss function is commonly used when training RNNs for sequence-to-sequence tasks like translation?
  - ❖ A) Mean Squared Error (MSE)
  - ❖ B) Cross-Entropy Loss
  - ❖ C) Hinge Loss
  - ❖ D) Triplet Loss
- ❑ Answer: B) Cross-Entropy Loss

11/27/2024

pra-sâmi

61

## Reflect...

- ❑ Which of the following techniques is used to solve the vanishing gradient problem in RNNs?
  - ❖ A) Weight initialization
  - ❖ B) Long Short-Term Memory (LSTM) networks
  - ❖ C) Gradient clipping
  - ❖ D) Data augmentation
- ❑ Answer: B) Long Short-Term Memory (LSTM) networks
- ❑ In an RNN, how does the hidden state affect the model?
  - ❖ A) It acts as the output of the network
  - ❖ B) It serves as a temporary memory to retain information from previous time steps
  - ❖ C) It stores the weights of the model
  - ❖ D) It determines the learning rate of the model
- ❑ Answer: B) It serves as a temporary memory to retain information from previous time steps
- ❖ Which component of the RNN is responsible for learning long-term dependencies?
  - ❖ A) Activation function
  - ❖ B) Bias term
  - ❖ C) Hidden state
  - ❖ D) Cell state (in LSTMs)
- ❑ Answer: D) Cell state (in LSTMs)
- ❑ What is the difference between an LSTM and a traditional RNN?
  - ❖ A) LSTMs have multiple layers of neurons
  - ❖ B) LSTMs use gates to control the flow of information and avoid vanishing gradients
  - ❖ C) LSTMs use feedforward connections
  - ❖ D) LSTMs are slower to train than RNNs
- ❑ Answer: B) LSTMs use gates to control the flow of information and avoid vanishing gradients

11/27/2024

pra-sâmi

62

**THANK YOU**

11/27/2024

pra-samī