SEQUENCE MODELING
INTRODUCTION TO RNNS

Deep Neural Networks

Session 18

Pramod Sharma
pramod.sharma@prasami.com

2

Agenda

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

pra-sâmi

3

Examples – Sequence Modelling

Domain	Data Type	Output type
Speech Recognition	Audio	Words (text)
Music Creation	Nodes (∅)	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	Mohan was driving a Maruti
Video activity recognition	Sequence of Video Frames	Identify activity say running

pra-sâmi

4

Sequence Modeling – Named Entity Recognition

- x : Mohan was driving a Maruti

- y: 1 0 0 0 1

pra-sâmi

5

Sequence Modeling – Named Entity Recognition

- x : <Mohan Sharma> was driving a <Maruti 800>

- y: 1 0 0 0 1

6

Sequence Modeling – Named Entity Recognition

- x : Mohan was driving a Maruti

x_1 x_2 x_3 x_4 x_5 $\Rightarrow x_t$

- y: 1 0 0 0 1

y_1 y_2 y_3 y_4 y_5 $\Rightarrow y_t$

- T_x is length of input and T_y is length of output

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

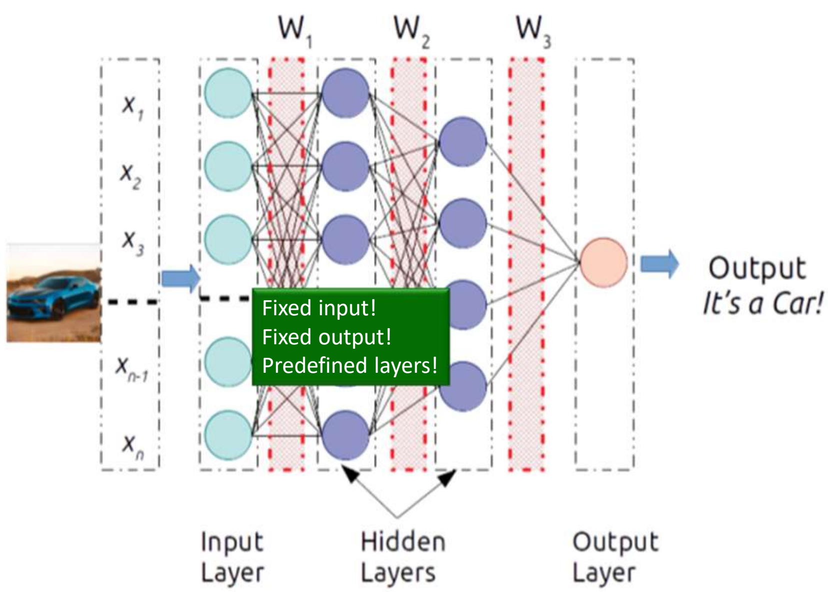
Mohan	was	driving	a	Maruti
0	0	0	1	0
0	0	0	0	0
–	–	1	–	–
–	–	–	–	–
–	–	–	–	–
–	–	–	0	1
1	–	–	–	–
–	1	–	–	–
0	0	0	0	0

5/24/2024

pra-sâmi

9

Using Standard Architecture



5/24/2024

pra-sâmi

10

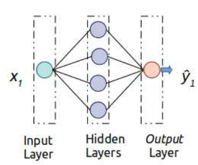
To Summarize….

- 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

pra-sâmi

11

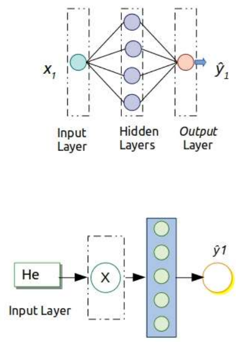
What is Recurrent Neural Network…



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

pra-sâmi

What is Recurrent Neural Network…



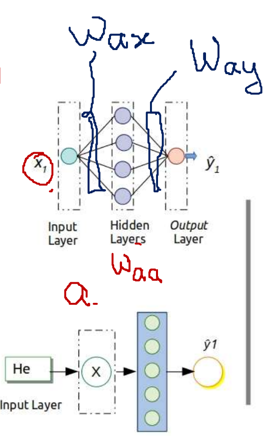
Simple Feed- Forward Network

- It takes one value and gives probability of it being a word or character or a value

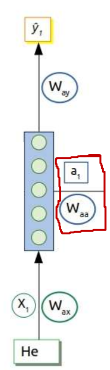
pra-sâmi

What is Recurrent Neural Network…

$(X.W_{aa} + b_{aa})$



Simple Feed- Forward Network

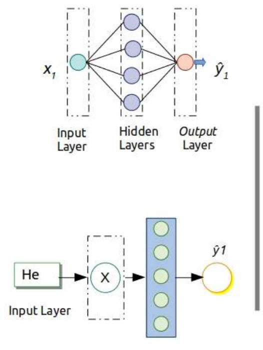


- Let's also calculate activations a_1 and weights W_{aa}
- Assume that we have some method of calculating them
- At the moment both W_{ax} and W_{aa} would seem to be same

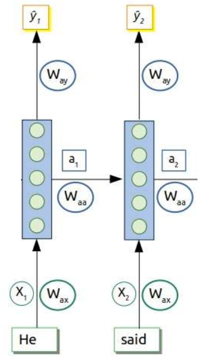
pra-sâmi

14

What is Recurrent Neural Network…



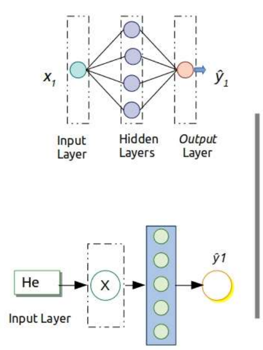
Simple Feed- Forward Network



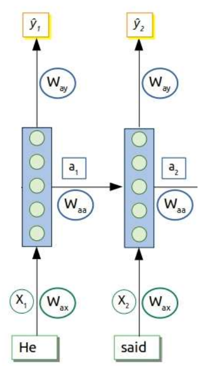
- Using the weights and activations, read X_2 and process it through the network
- Calculation of $\hat{y}_2$ will be based on X_2, W_{ax}, W_{aa} and a_1,

15

What is Recurrent Neural Network…



Simple Feed- Forward Network

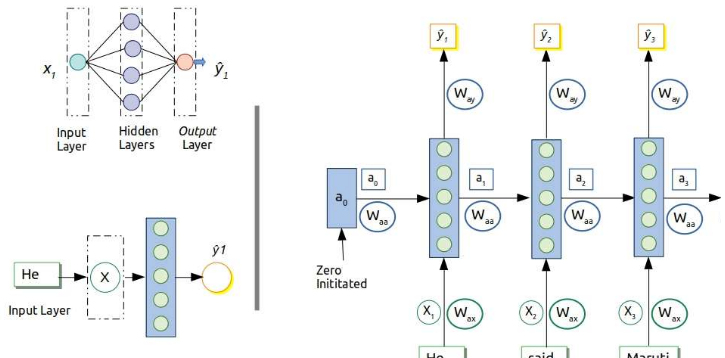


- But it makes two set of calculations
- Using different formulae
- To make it consistent let's initialize a_0 with weights W_{aa}

16

What is Recurrent Neural Network…

- Similarly we can calculate $\hat{y}_3$



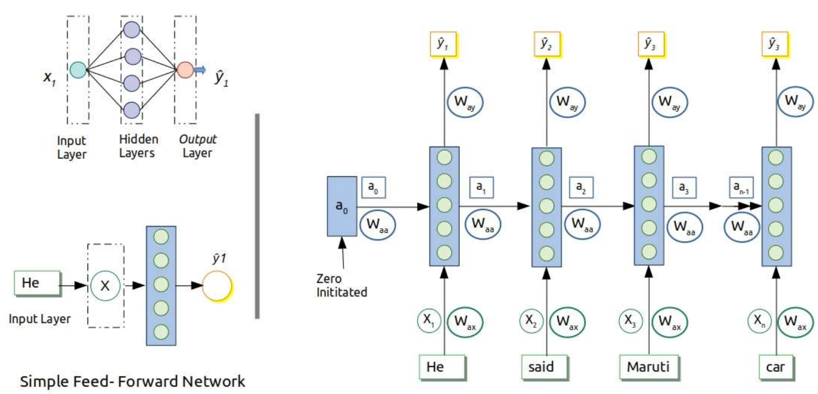
Simple Feed- Forward Network

pra-sâmi

17

What is Recurrent Neural Network…

- And continue till end,
- Some literatures represent it with a loop,

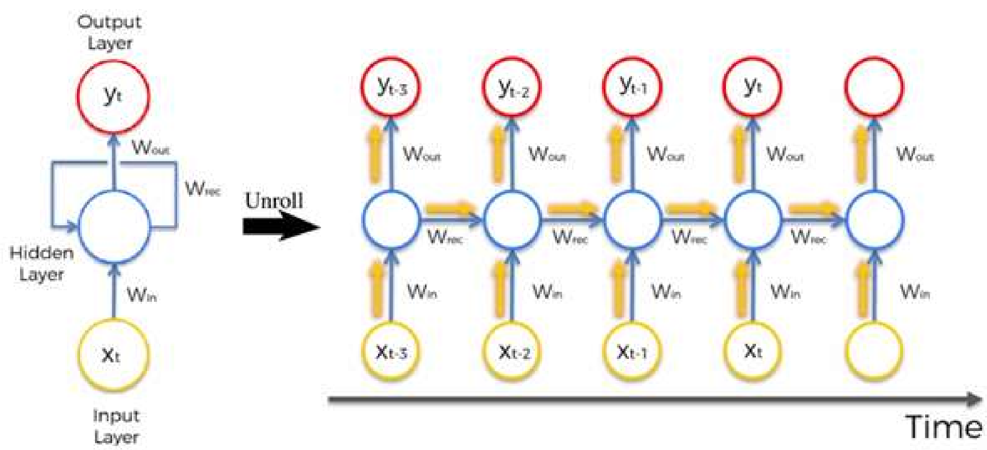


Simple Feed- Forward Network

pra-sâmi

18

Alternate Representations

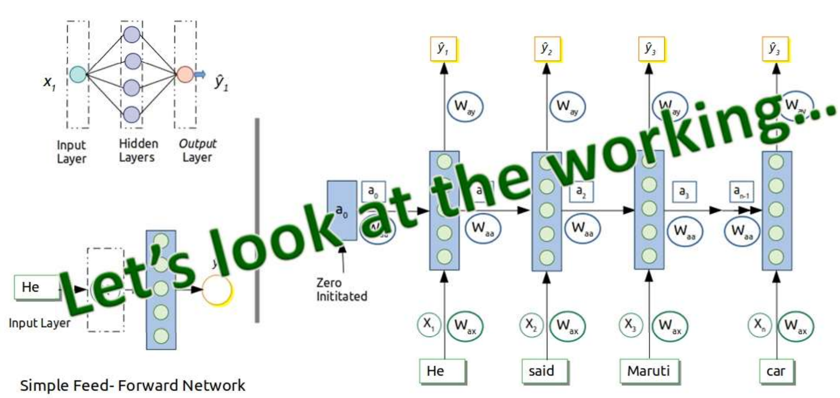


5/24/2024

pra-sâmi

19

What is Recurrent Neural Network…

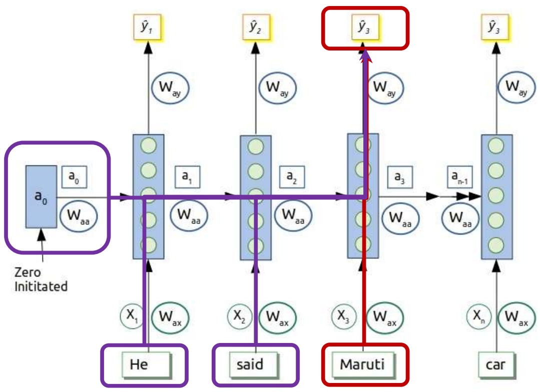


5/24/2024

pra-sâmi

20

What is Recurrent Neural Network…

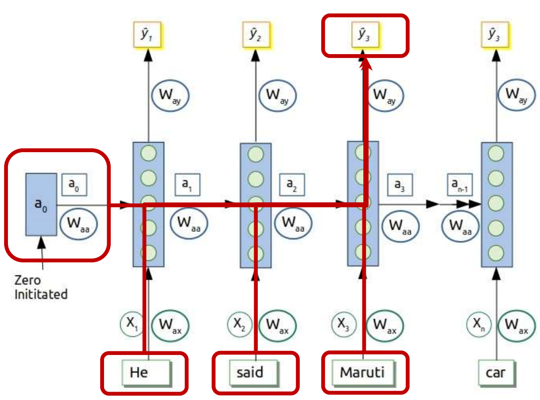


Recurrent Neural Network

- 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…

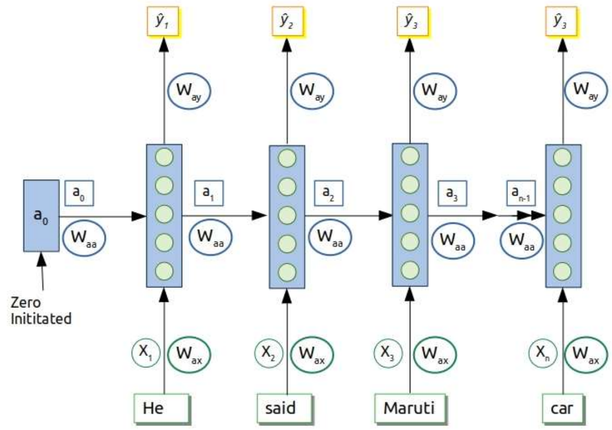


Recurrent Neural Network

- 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…



Recurrent Neural Network

- Its it great!
- All done… sealed, signed, and delivered…
- Wait… let's do some math too….

5/2

pra-sâmi

23

What We Know So Far….

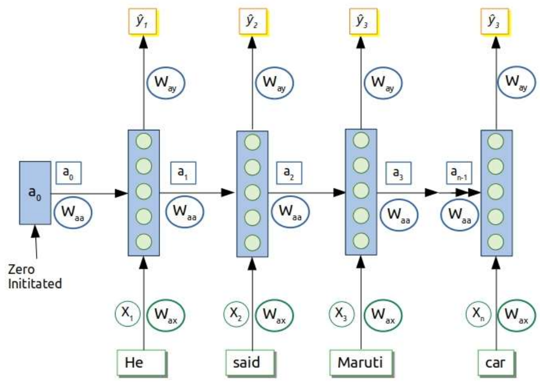
- Recurrent Neural Networks take the previous output or hidden states as inputs.
- The composite input at time 't' has some historical information about the happenings at time 'T' < 't'.
- RNNs are useful as their intermediate values (state) can store information about past inputs for a time that is 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

5/24/2024

pra-sâmi

24

Forward Propagation

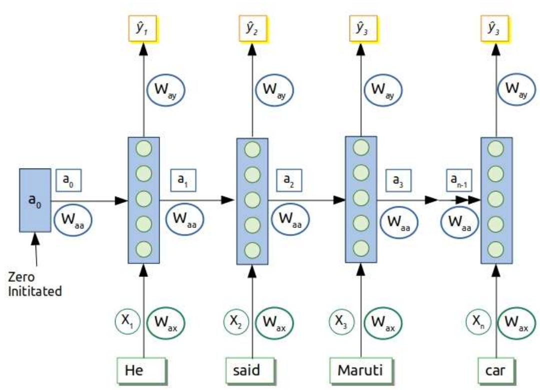


Recurrent Neural Network

- Let's work on equations

25

Forward Propagation

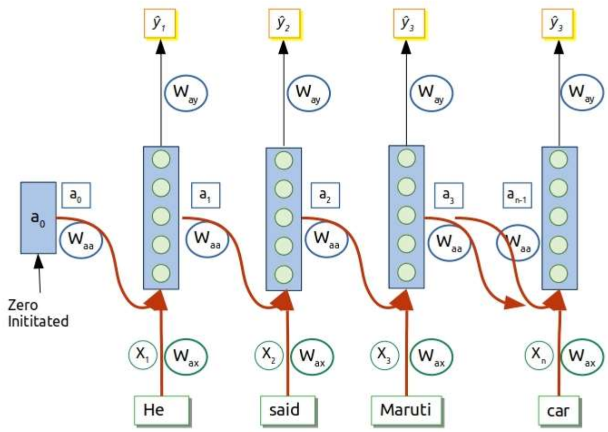


Recurrent Neural Network

- To start with; a_0 is vector of all zeros
  - $a_1 = g_1\ (a_0\ .\ W_{aa} + X_1\ .\ W_{ax} + b_a)$ ➔ Tanh / ReLU
  - $\hat{y}_1 = g_2\ (a_1 .\ W_{ay} + b_y)$ ➔ 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
  - $a_t = g_1\ (a_{t-1}\ .\ W_{aa} + X_t\ .\ W_{ax} + b_a)$
  - $\hat{y}_t = g_2\ (a_t .\ W_{ay} + b_y)$

26

Forward Propagation



Recurrent Neural Network

- Our equations

  $a_t = g_1(a_{t-1}.W_{aa} + x_t.W_{ax} + b_a)$

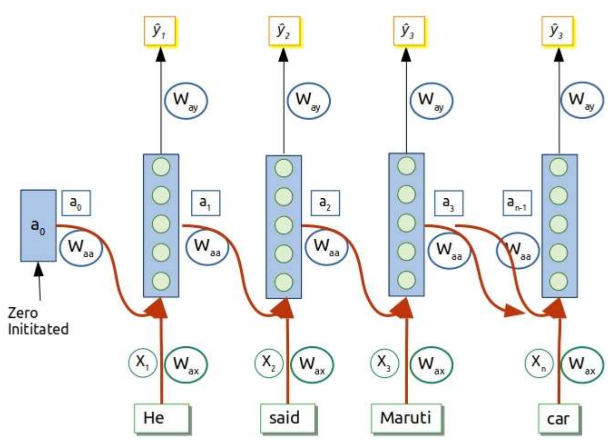
  $\hat{y}_t = g_2\big(a_t.W_{ya} + b_y\big)$

5/24/2024

pra-sâmi

27

Forward Propagation



Recurrent Neural Network

- Our equations

  $a_t = g_1(a_{t-1}.W_{aa} + x_t.W_{ax} + b_a)$

  $\hat{y}_t = g_2\big(a_t.W_{ay} + b_y\big)$

- Can be written as:

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

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

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 :

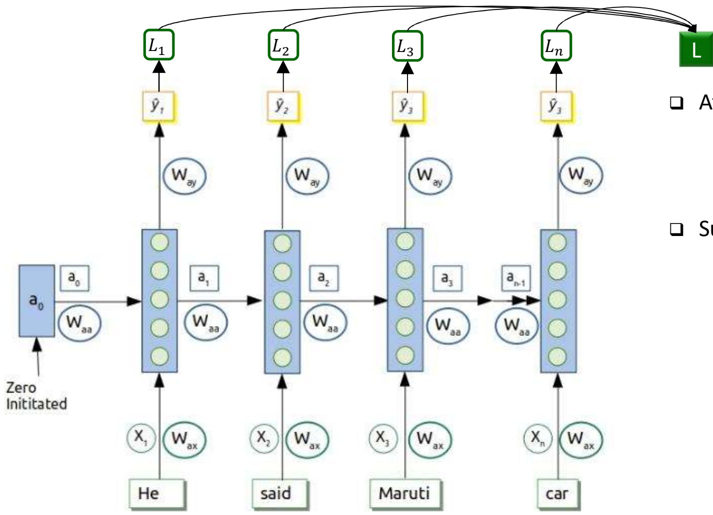
$[a_{t-1} \mid x_t] \; . \; [\begin{matrix} \mathrm{W_{aa}} \\ \mathrm{W_{ax}} \end{matrix}] = a_{t-1}.W_{aa} + x_t.W_{ax}$

5/24/2024

pra-sâmi

28

Back Propagation



Recurrent Neural Network

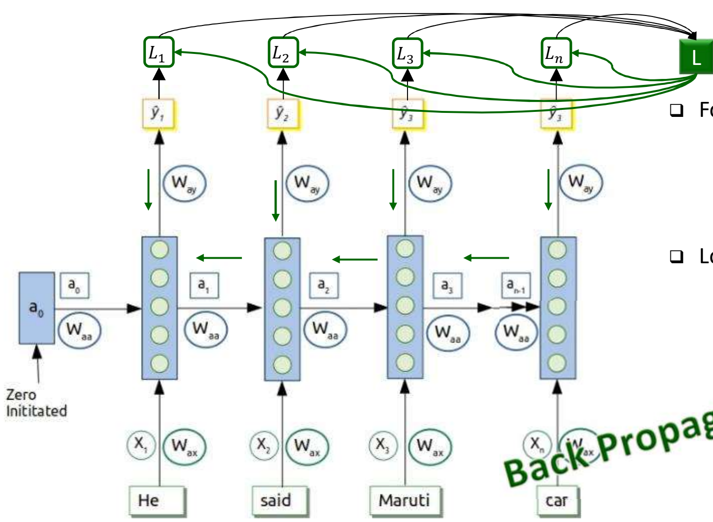
- At time step 't'; Loss Function for single prediction
  - $L_t(\hat{y}_t, y) = - y_t \,.\, \log(\hat{y}_t) - (1- y_t) \,.\, \log(1 - \hat{y}_t)$
- Sum of losses at all time steps:
  - $L(\hat{y}, y) = \sum_{t=1}^{T_x} L_t(\hat{y}_t, y)$

5/24/2024

pra-sâmi

30

Back Propagation



Recurrent Neural Network

- Forword propagation:

$$a_t = g_1([a_{t-1}, x_t] \,.\, W_a + b_a)$$
$$\hat{y}_t = g_2(a_t . W_y + b_y)$$

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

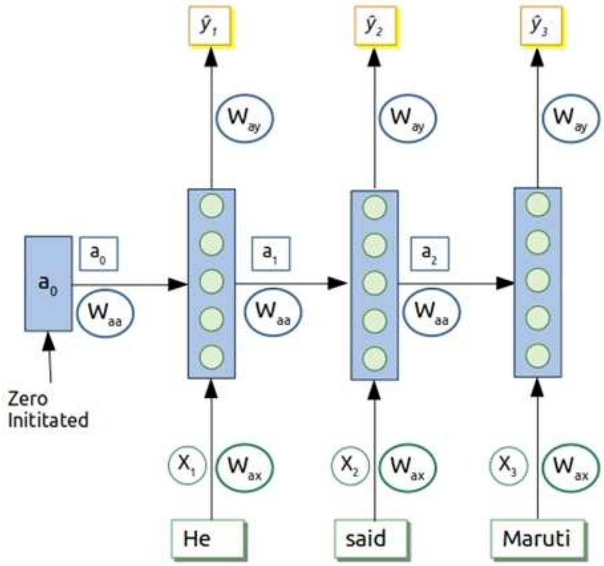
Back Propagation through Time.

5/24/2024

pra-sâmi

31

Back Propagation Through Time…



Forword propagation:

$$a_t = g_1([a_{t-1}, x_t].W_a + b_a)$$
$$\hat{y}_t = g_2(a_t.W_y + b_y)$$

Loss Function :

$$L_t(\hat{y}, y) = -y_t.\log(\widehat{y_t}) - (1 - y_t).\log(1 - \widehat{y_t})$$

Step 3:

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

$$\frac{dL_3}{dw_a} = \frac{dL_3}{d\hat{y}_3}.\frac{d\hat{y}_3}{da_3}.\frac{da_3}{dw_a} + \frac{dL_3}{d\hat{y}_3}.\frac{d\hat{y}_3}{da_3}.\frac{da_3}{da_2}.\frac{da_2}{dw_a} + \frac{dL_3}{d\hat{y}_3}.\frac{d\hat{y}_3}{da_3}.\frac{da_3}{da_2}.\frac{da_2}{da_1}.\frac{da_1}{dw_a}$$

There is a pattern here!

$$\frac{dL_3}{dw_x} = \frac{dL_3}{d\hat{y}_3}.\frac{d\hat{y}_3}{da_3}.\frac{da_3}{dw_x} + \frac{dL_3}{d\hat{y}_3}.\frac{d\hat{y}_3}{da_3}.\frac{da_3}{da_2}.\frac{da_2}{dw_x} + \frac{dL_3}{d\hat{y}_3}.\frac{d\hat{y}_3}{da_3}.\frac{da_3}{da_2}.\frac{da_2}{da_1}.\frac{da_1}{dw_x}$$

5/24/2024 pra-sâmi

33

Quickly check the dimension….

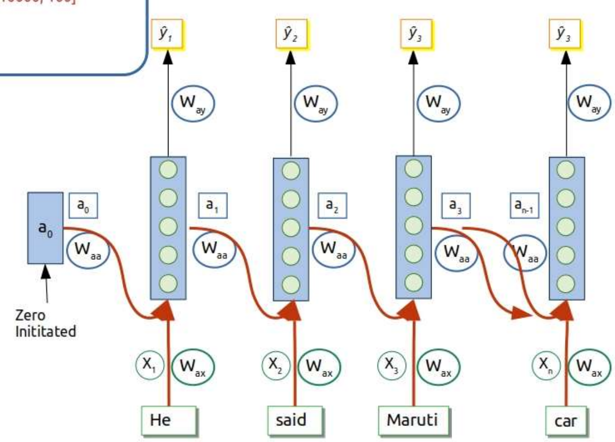
$$a_t = g^1(\underset{[100,100]}{a_{t-1}} . \underset{[100,\ 100]}{W_{aa}} + \underset{[100,10000]}{X_t} . \underset{[10000,\ 100]}{W_{ax}} . + b_a)$$

$$\hat{y}_t = g^2(a_t . W_{ya}. + b_y)$$

$$a_t = g^1([a_{t-1} : X_t].W_a.+ b_a)$$
$$\hat{y}_t = g^2(a_t .W_y + b_y)$$

$$\underset{[100 \times 10100]}{[a_{t-1}, X_t]} = [\underset{[100]}{a_{t-1}} : \underset{[10000]}{X_t}]\ [100]$$

$$W_a = \overset{[100]}{\begin{pmatrix} W_{aa} \\ W_{ax} \end{pmatrix}} \begin{matrix} [100] \\ [10000] \end{matrix}$$

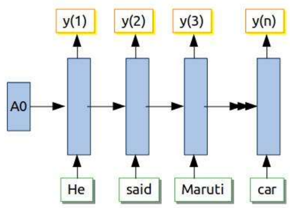


Recurrent Neural Network

5/24/2024 pra-sâmi

34

Type of Architectures



Many-to-Many Architecture

Many to many mapping. T_x input parameters are same as T_y output parameters

Named entity Recognition:
Mohan was driving a Maruti
➔ 1 0 0 0 1

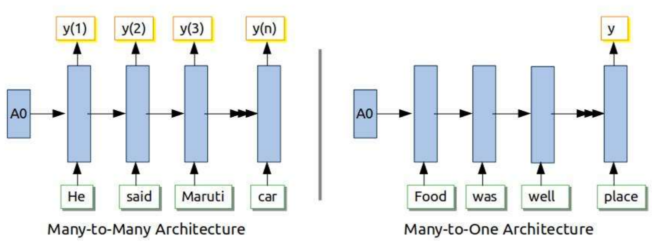
5/24/2024

The Unreasonable Effectiveness of Recurrent Neural Networks - Andrej Karpathy

pra-sâmi

35

Type of Architectures



Many-to-One Architecture

Many to one architecture.

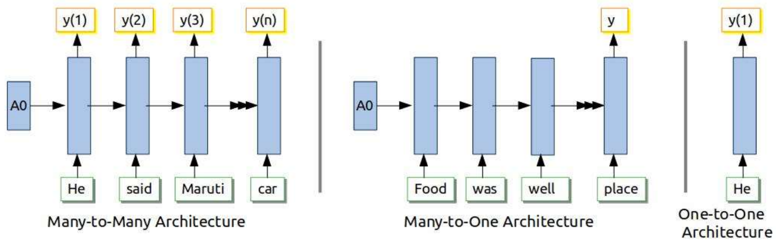
Input is the 'review' written by a patron and output is an integer (star rating)

5/24/2024

The Unreasonable Effectiveness of Recurrent Neural Networks - Andrej Karpathy

pra-sâmi

36 Type of Architectures

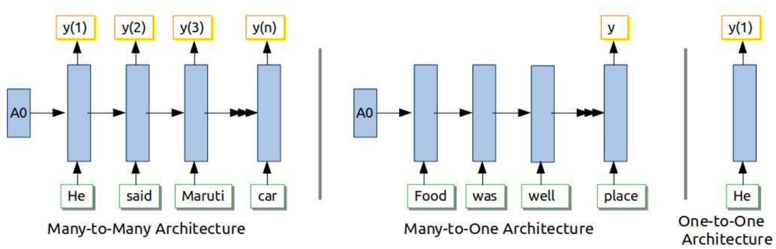


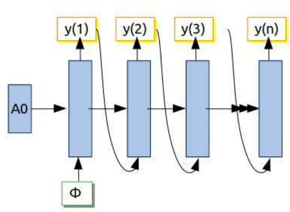
Of course there is one to one. i.e. Basic neural network…

The Unreasonable Effectiveness of Recurrent Neural Networks
- Andrej Karpathy

pra-sâmi

37 Type of Architectures





One-to-Many Architecture

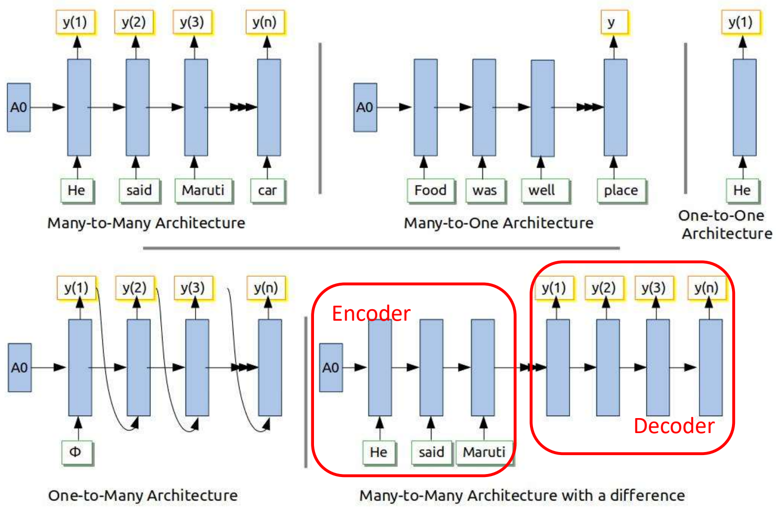
We also have one to many architecture, mapping one input to multiple outputs. Output from each layer becomes input for next time step.

The Unreasonable Effectiveness of Recurrent Neural Networks
- Andrej Karpathy

pra-sâmi

38

Type of Architectures



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

The Unreasonable Effectiveness of Recurrent Neural Networks
- Andrej Karpathy

39

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(Toad met Pit) = $4.6\ x\ 10^{-1}$
  - P(Todd met Pete) = $9.3\ x\ 10^{-9}$
- Mathematically P(sentence) = P($y_1, y_2, y_3, \ldots\ y_n$)

40

How to Model?

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

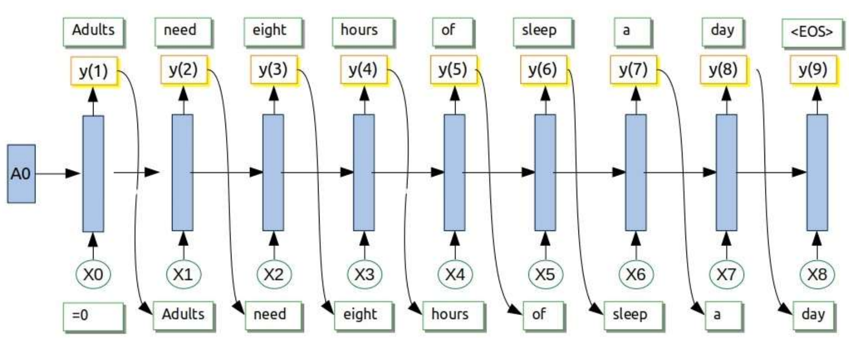
Adults	need	eight	hours	of	sleep	a	day	!	<EOS>
y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	-	y_9

- 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>

41

RNN Model

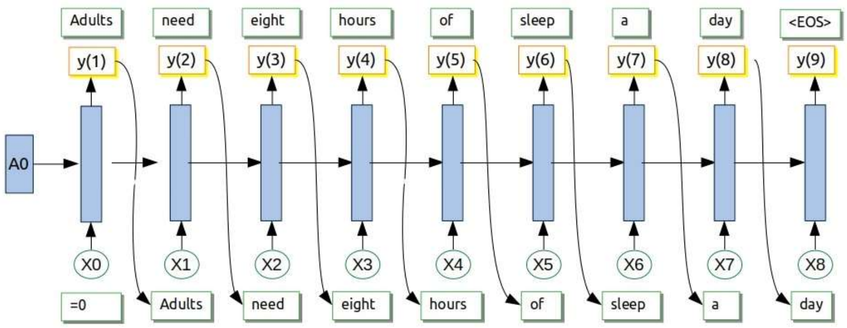
- At the onset RNN tries to predict probabilities of each word in the corpus of being first word in this sentence.
- i.e. P[a], P[aakash], P[aamaan]… to P[zulu], P[zyzzogeton]
  - This would be an array of 10002 elements



42

RNN Model

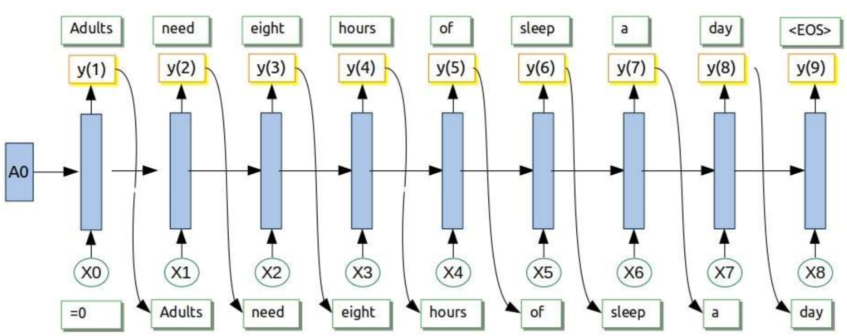
- Thus we can calculate error between $\hat{y}_1$ and "Adults"
- Given first word "Adults", again RNN predicts the probabilities for second word, thus combined probability, and it continues…
  - i.e. P[a|Adult], P[aakash|Adult], P[aamaan |Adult]… to P[zulu |Adult], P[zyzzogeton |Adult]
- Somewhere in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



43

RNN Model

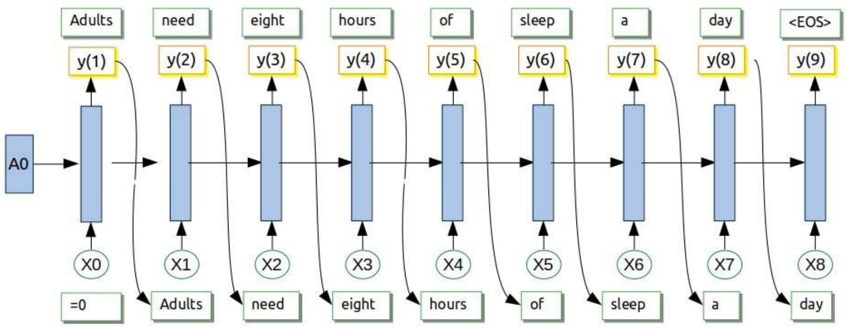
- At third step we can calculate error between $\hat{y}_2$ and "need".
- Given first word "Adults", and second word as "need", again RNN predicts the probabilities for third word
- i.e. P[a | Adult, need], P[aakash | Adult , need], P[aamaan | Adult , need]… to P[zulu |Adult , need], P[zyzzogeton |Adult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



44

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>.

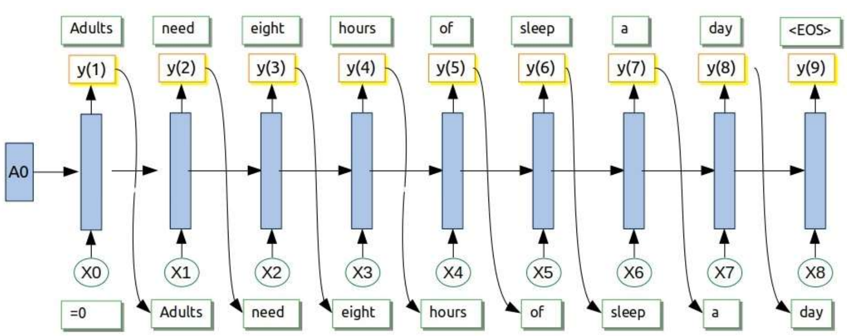


pra-sâmi

45

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
- ℓ (ŷ, y) = -y * log (ŷ) as ŷ is very close to 0 for all other words
  - since its remaining part [(1 – y) * log (1 – ŷ)] is insignificantly small we can ignore it.

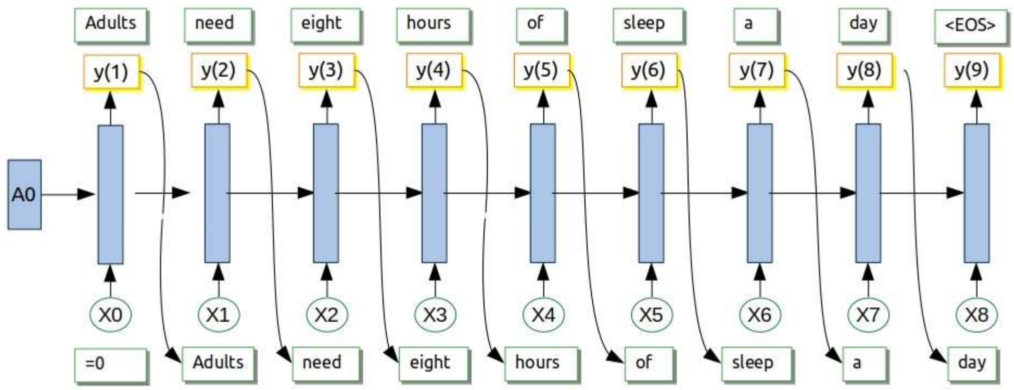


pra-sâmi

46

RNN Model

- Thus for overall sentence, Cost will be
  - J(ŷ, y) = Σ ℓ (ŷ, y)
  - J(ŷ, y) = - $\frac{1}{m}$Σ y * log(ŷ)
  - Which we will be minimizing.



pra-sâmi

47

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]

pra-sâmi

48

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	Women	King	Queen	Apple	Oranges
(5468)	(8701)	(4823)	(7157)	(56)	(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

This representation is treating words independently….

5/24/2024

pra-sâmi

49

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 ≈ King → ????

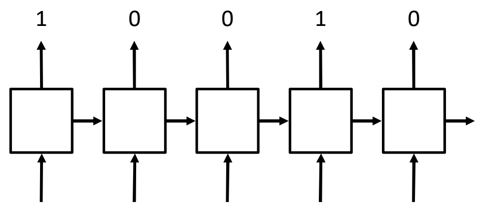
- In terms of algorithm, we can use this using Similarity Coefficients
  - Find a word W : argmax ($e_w, \;\; e_{king} - e_{man} + e_{women}$)
  - Cosine sim (u, v) = $\frac{(u^T.v)}{||u||_2.||v||_2}$
  - Euclidian distances or Manhattan distances can also be used

5/24/2024

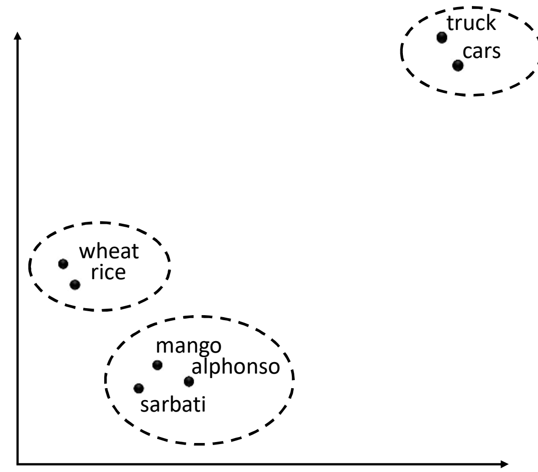
pra-sâmi

50

Named entity and word embedding



Mohan	is	a	wheat	farmer
Sudesh	is	a	mango	______
Ravi	is	a	sarbati	cultivator



Words	→	Embedding Layer	→	Dense Layer W_d, b_d	→	Softmax Layer W_o, b_0
5 words		5 x 300				10000 probabilities

51

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.

52

RNN Model

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

5/24/2024

pra-sâmi

53

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)

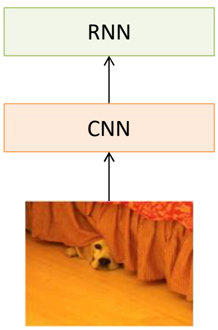
: The dog is hiding

5/24/2024

pra-sâmi

54

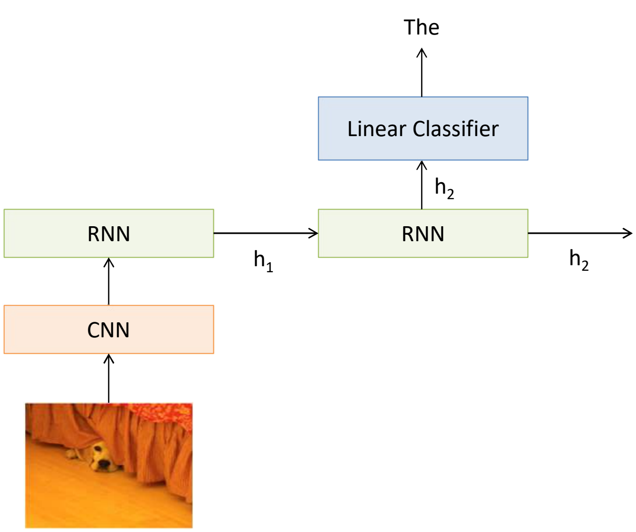
Image Captioning



5/24/2024

55

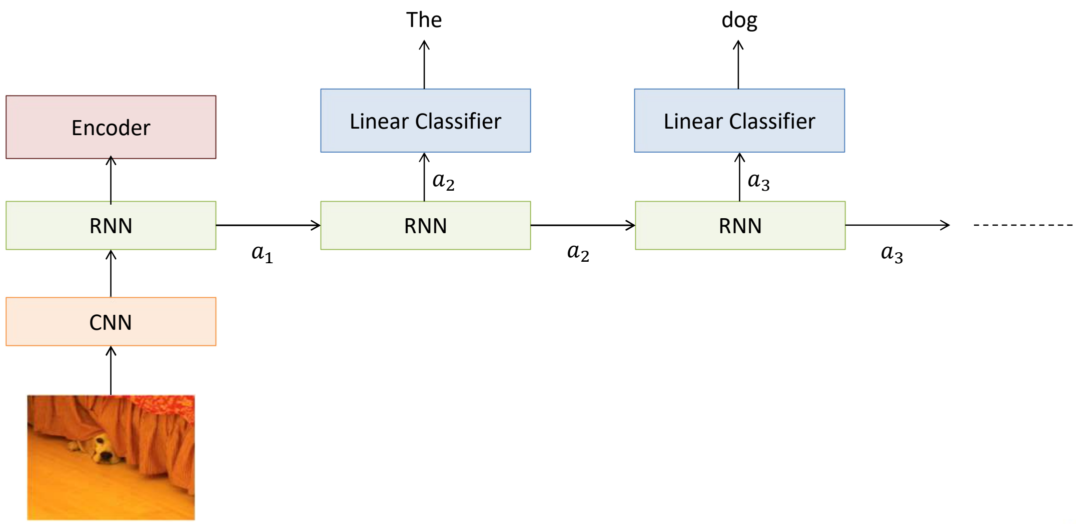
Image Captioning



5/24/2024

56

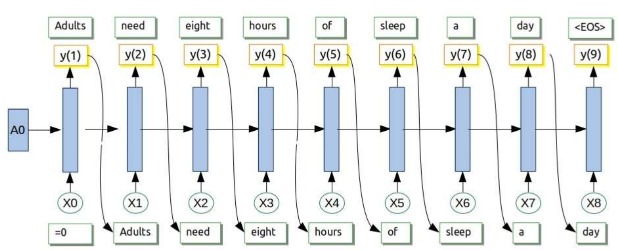
Image Captioning



pra-sâmi

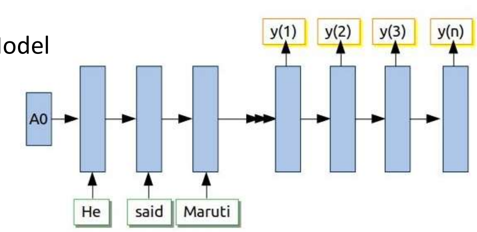
57

Machine translation



Language Model

- Conditional language Model



Machine translation

pra-sâmi

58

Sequence to sequence : Bleu Score

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

5/24/2024

pra-sâmi

59

RNN Outputs: Image Captions

Show and Tell: A Neural Image Caption Generator, CVPR 15

5/24/2024

pra-sâmi

60