Seq2Seq Data

08 December 2023 17:12

cgpa | iq | placement

DNN

1> Tabular → [ANN] x

2> Image data → [2d grid] → CNN x

dog

cat

3> Sequential → textual

→ timeseries

RNN

LSTMs GRUs

Encoder Decoder Archi

4> Seq2seq data difficult

[input seq → output seq] machine translation

nice to meet you

↓

आप से मिल कर अच्छा लगा

input → sent → eng → variable length

output → sent → hindi → variable leng

3 words eng → 3 words

→ variable length

[lstm / gru] → input

[output]

Before Starting

08 December 2023 19:24

Prerequisite

- RNN / LSTM

Plan of attack

→ simple version

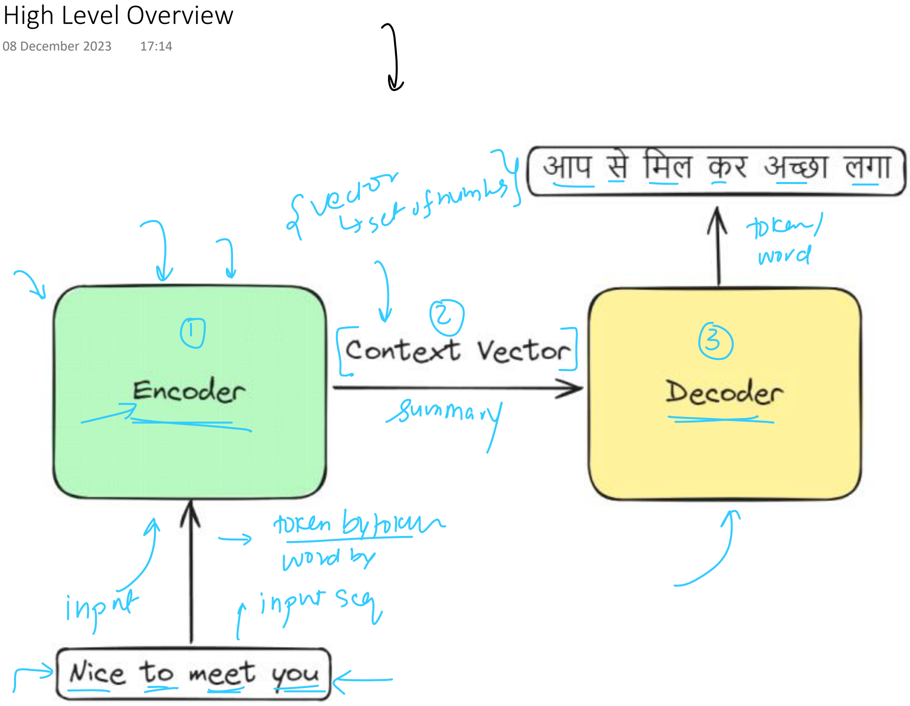
→ deep

→ improvements

High Level Overview

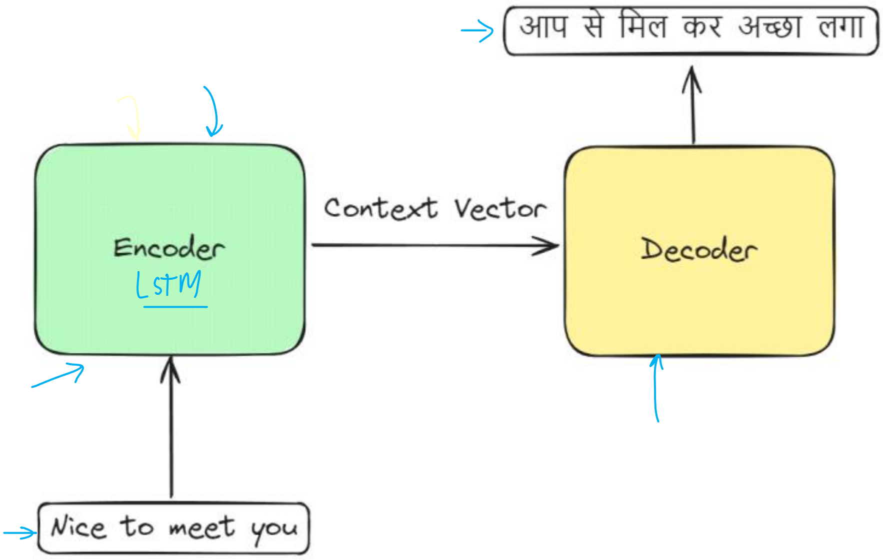
08 December 2023 17:14

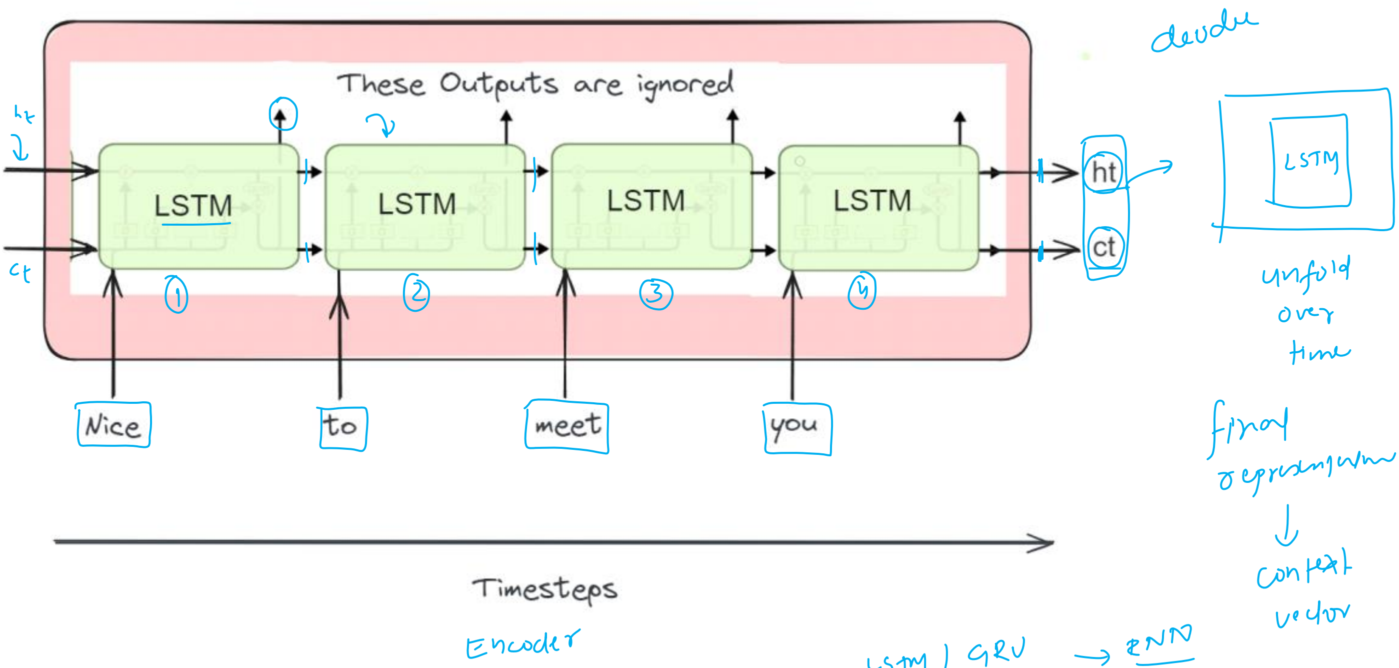
Machine tran → exam



What's under the hood?

08 December 2023 17:14

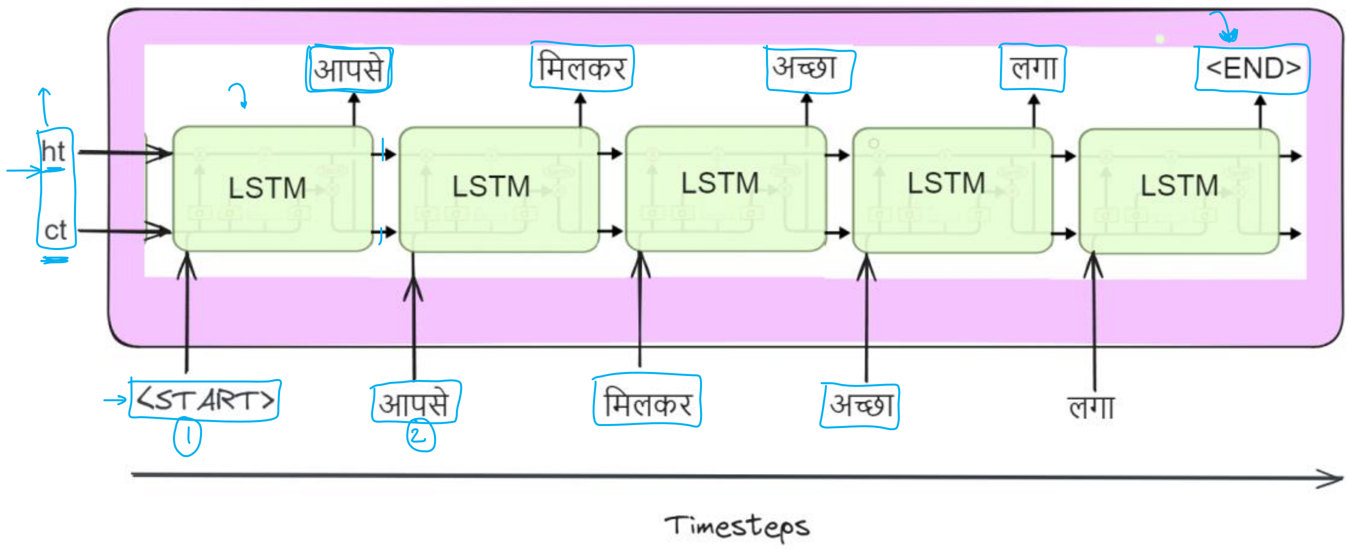




unfold over time

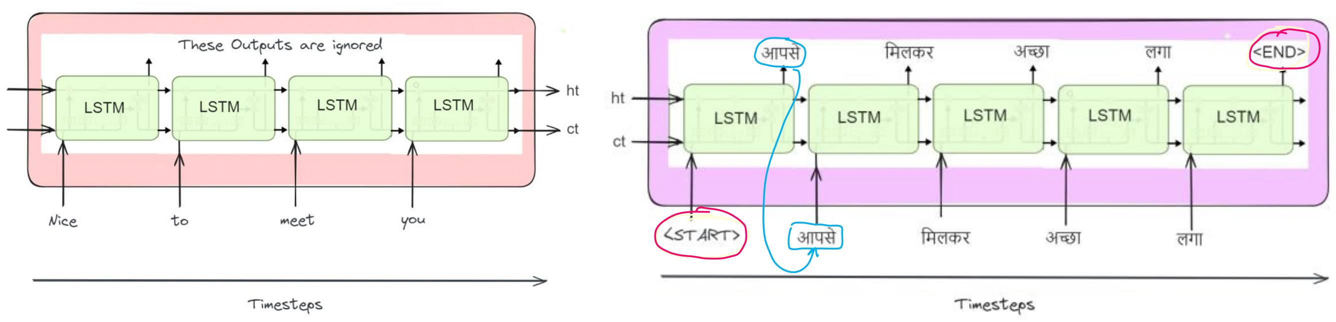
final representation ↓ context vector

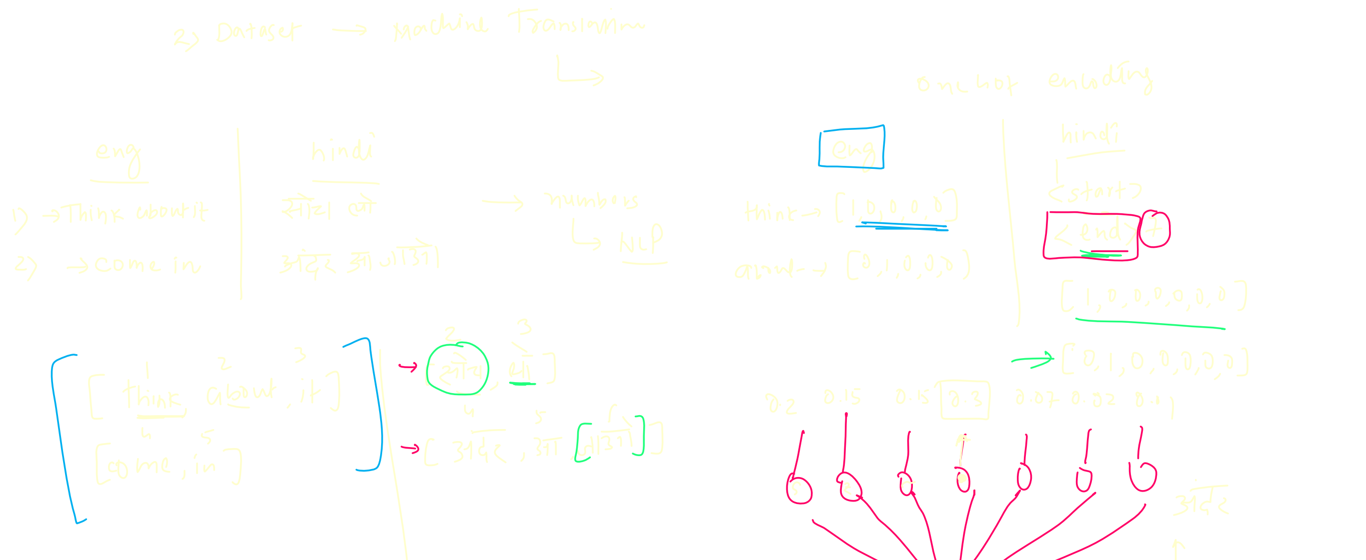
LSTM / GRU → RNN


ht
ct
आपसे
मिलकर
अच्छा
लगा
<END>
LSTM
LSTM
LSTM
LSTM
LSTM
<START>
1
आपसे
2
मिलकर
अच्छा
लगा
Timesteps

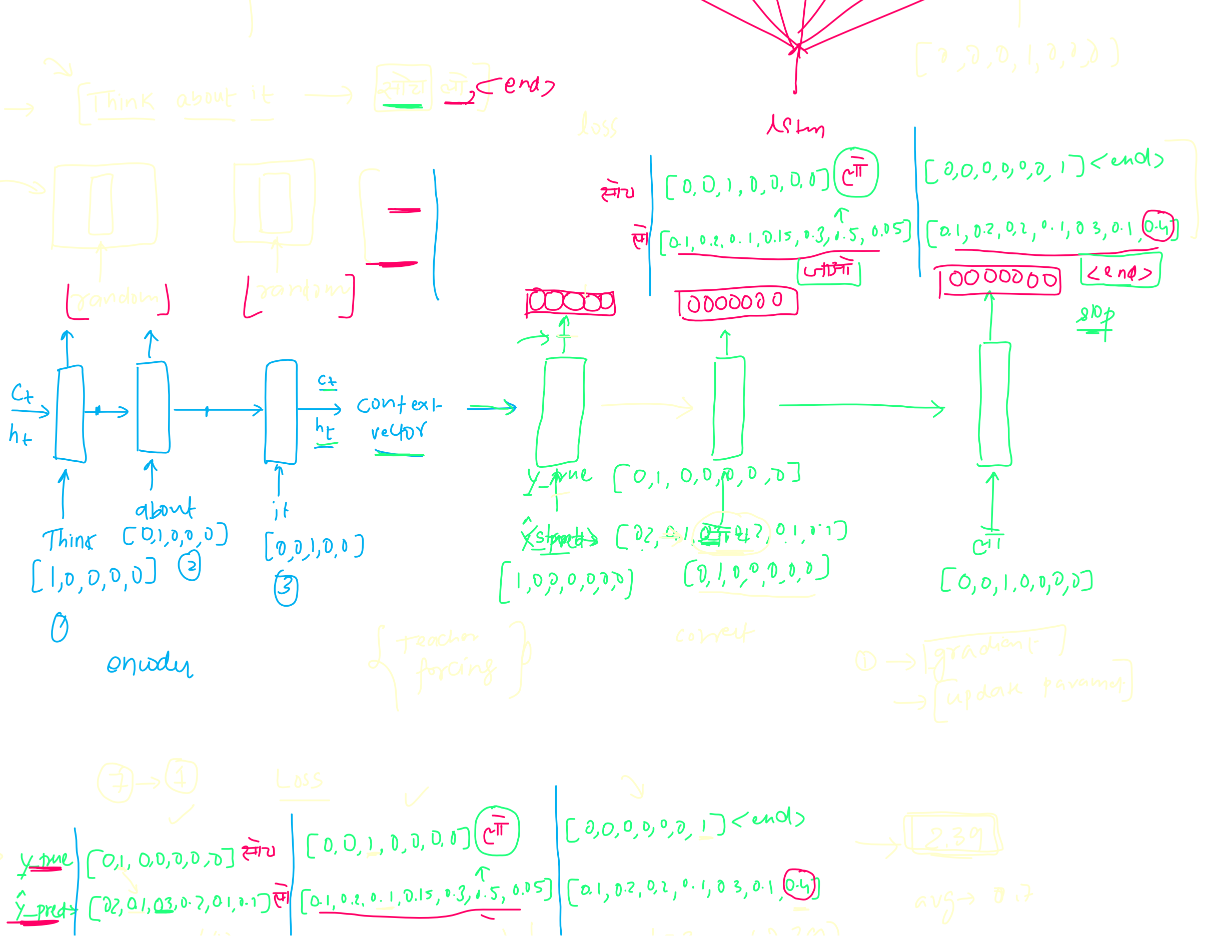
Training the Architecture using Backpropagation

08 December 2023 17:15





Teacher forcing techniue is there which means only corrected wala is given as input in given time step no matter whatever the output of previous.(If not do this then training become slow)



→ y_true [0, 1, 0, 0, 0, 0, 0] स्वागत | [0, 0, 1, 0, 0, ...] | avg → 0.7

→ $\hat{y}$_pred → [0.2, 0.1, 0.3, 0.2, 0.1, 0.1] से | [0.1, 0.2, 0.1, 0.15, 0.3, 0.5, 0.05] | [0.1, 0.2, 0.2, 0.1, 0.3, 0.1, 0.4]

t=1 (1) | t=2 (1) | t=3 (0.39)

Loss → categorical cross entropy

1) fwd
2) loss
3) Backprop

$$L = -\sum_{i=1}^{7} y_i^{true} \log(y_i^{pred})$$

$L_{t=1} = -1 \times \log(0.1)$

$= 1$

$L_{t=2} = 1$

$L_{t=3} = -1 \times \log(0.4)$

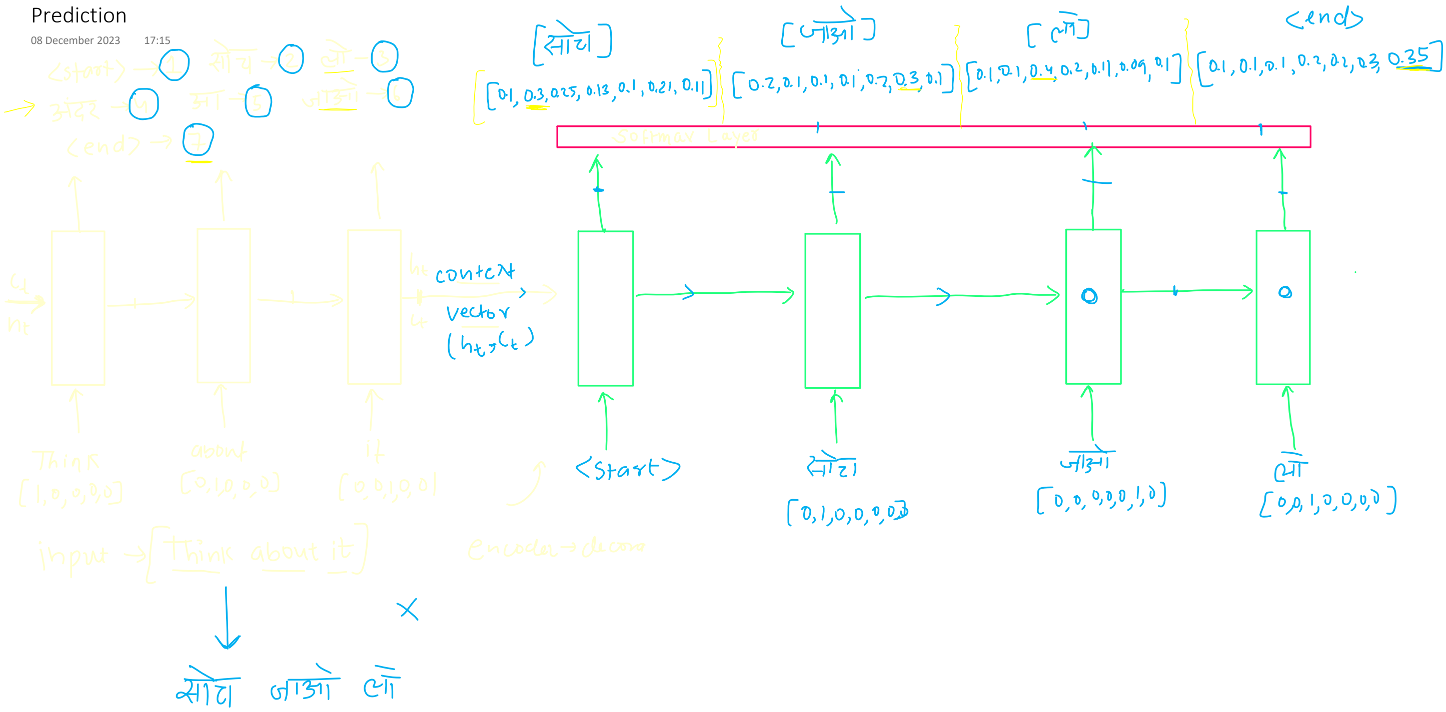
$= -1 \times -0.39 = 0.39$

training

③ (initial wts) forward prop → output + true → loss → gradient → weight update (Adam) → learning rate → new weights

Prediction

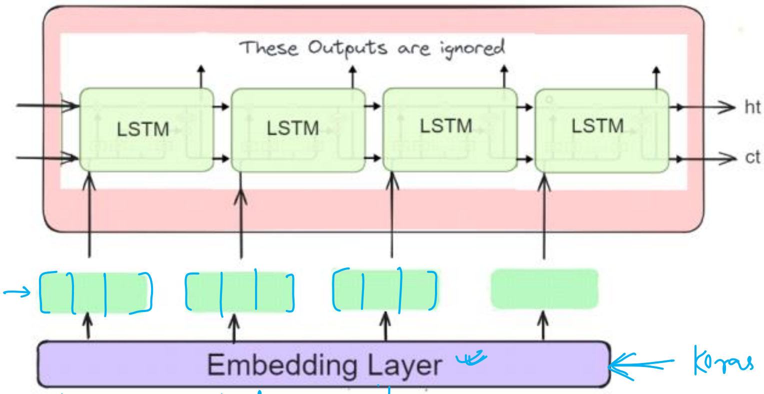
08 December 2023 17:15



Improvement 1 - Embeddings

08 December 2023 17:16

encoder



<start

encoder

decoder

Keras

think about it

Think

[1,0,0,0,0]

5 words

3 dim

low dimen / dense

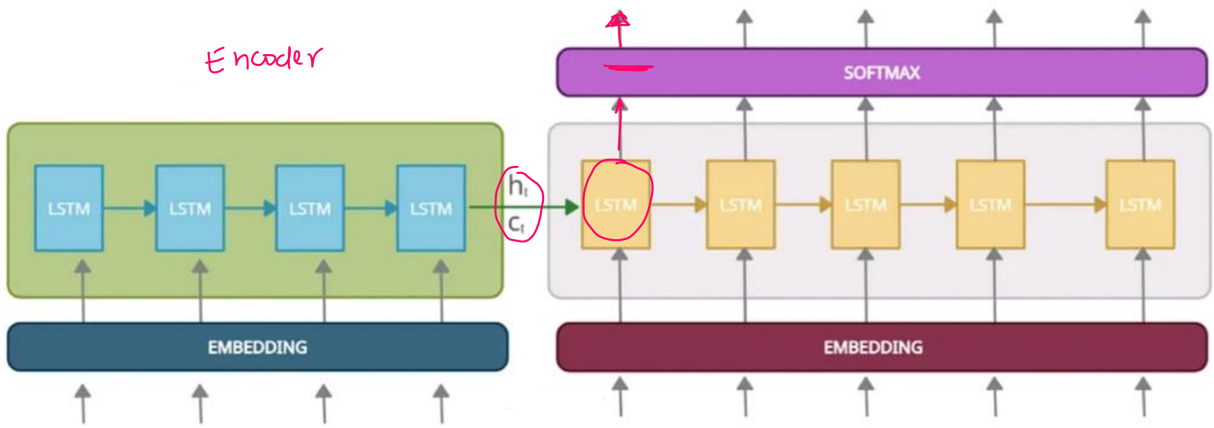
summary of word

pretrained → Word2vec / Glove

train

Decoder

Encoder



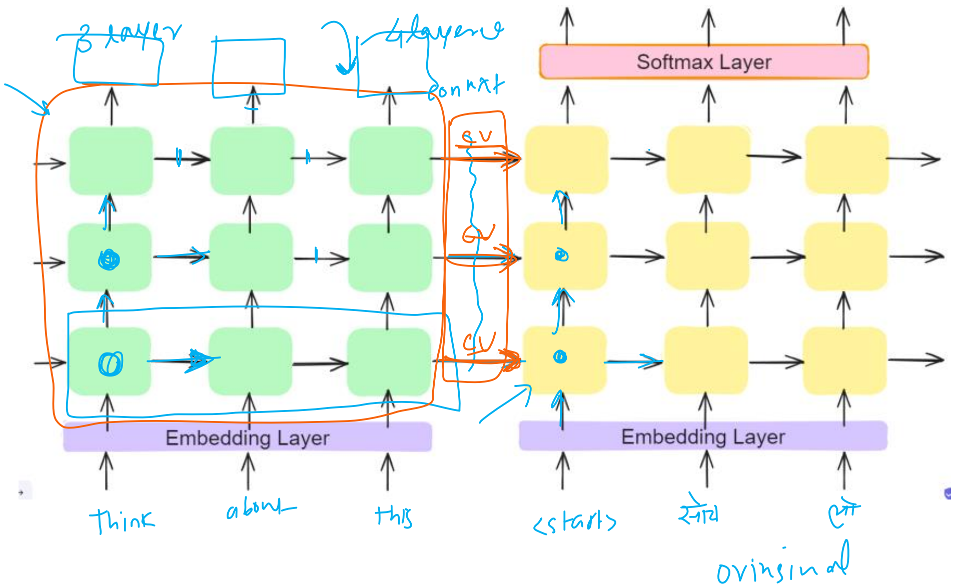
Nice to meet you

<start>

Improvement 2 - Deep LSTMs

08 December 2023 17:16

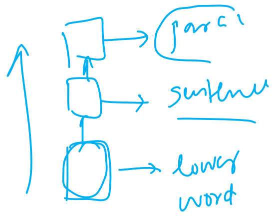
long para single layer



3> deep learning NN learning

1) long term dependency

2) layered representation

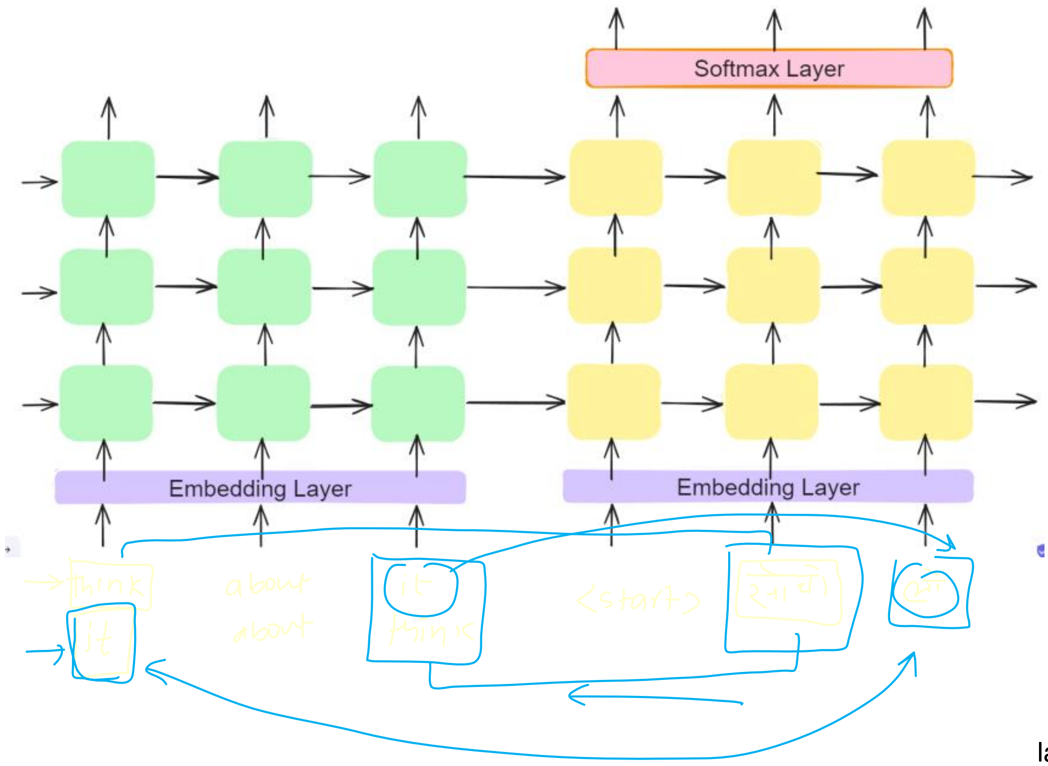


the phone battery is bad but the display is great

Here liek deep rnn , you can think like (one captures at word level , one captures at sentence level , one captures at para level.

Improvement 3 - Reversing the Input

08 December 2023 17:16



language
↓
initial context
eng - french

This reversing works for well , where like some langauga start words have more meaning .

The Sutskever Architecture

08 December 2023 17:18

<start> <end>

Application to Translation: The model focused on translating English to French, demonstrating the effectiveness of sequence-to-sequence learning in neural machine translation.

Special End-of-Sentence Symbol: Each sentence in the dataset was terminated with a unique end-of-sentence symbol ("<EOS>"), enabling the model to recognize the end of a sequence.

Dataset: The model was trained on a subset of 12 million sentences, comprising 348 million French words and 304 million English words, taken from a publicly available dataset.

Vocabulary Limitation: To manage computational complexity, fixed vocabularies for both languages were used, with 160,000 most frequent words for English and 80,000 for French. Words not in these vocabularies were replaced with a special "UNK" token.

Reversing Input Sequences: The input sentences (English) were reversed before feeding them into the model, which was found to significantly improve the model's learning efficiency, especially for longer sentences.

Word Embeddings: The model used a 1000-dimensional word embedding layer to represent input words, providing dense, meaningful representations of each word.

Architecture Details: Both the input (encoder) and output (decoder) models had 4 layers, with each layer containing 1,000 units, showcasing a deep LSTM-based architecture.

Output Layer and Training: The output layer employed a Softmax function to generate the probability distribution over the target vocabulary. The model was trained end-to-end with these settings.

Performance - BLEU Score: The model achieved a BLEU score of 34.81, surpassing the baseline Statistical Machine Translation (SMT) system's score of 33.30 on the same dataset, marking a significant advancement in neural machine translation.