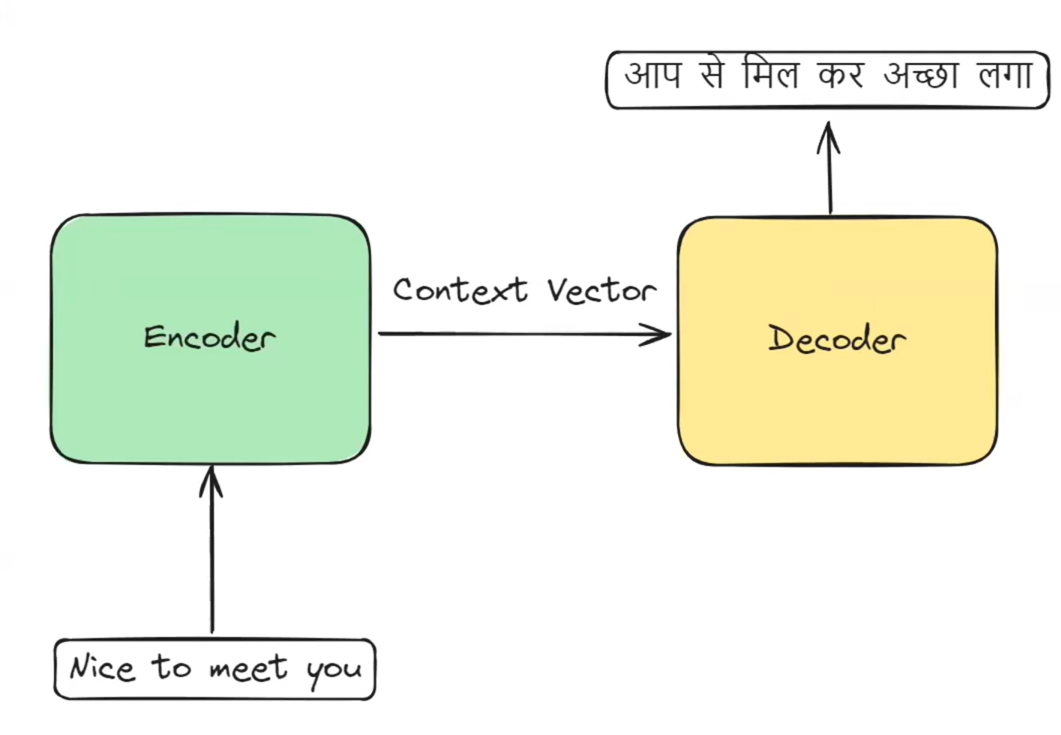
**Encoder Decoder Sequence-to-sequence Architecture**

Challenges in seq2seq Problems

1. I/P → Variable length
2. 0/P → Variable Length
3. No guarantee I/P Len = O/P Len

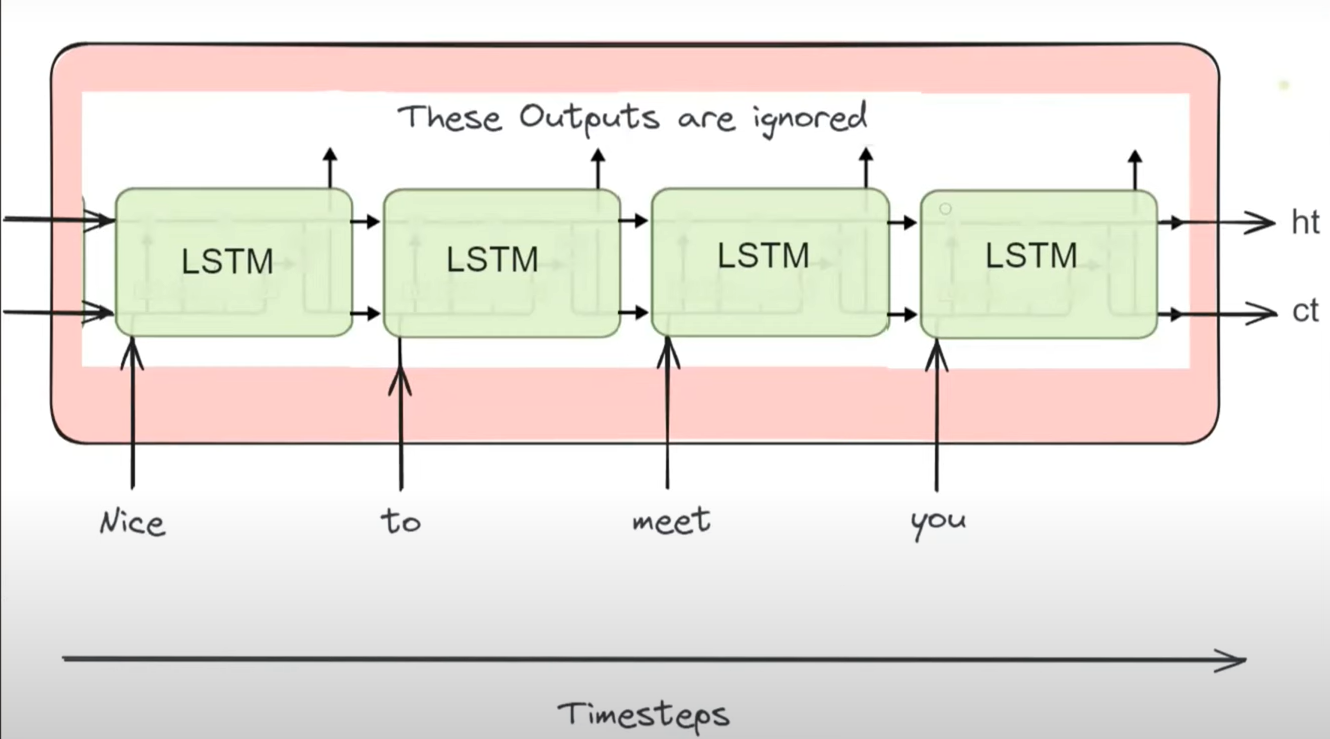
\* We Learn how to handle variable length in → LSTM / GRU but in I/P not in O/P

**High Level Overview**



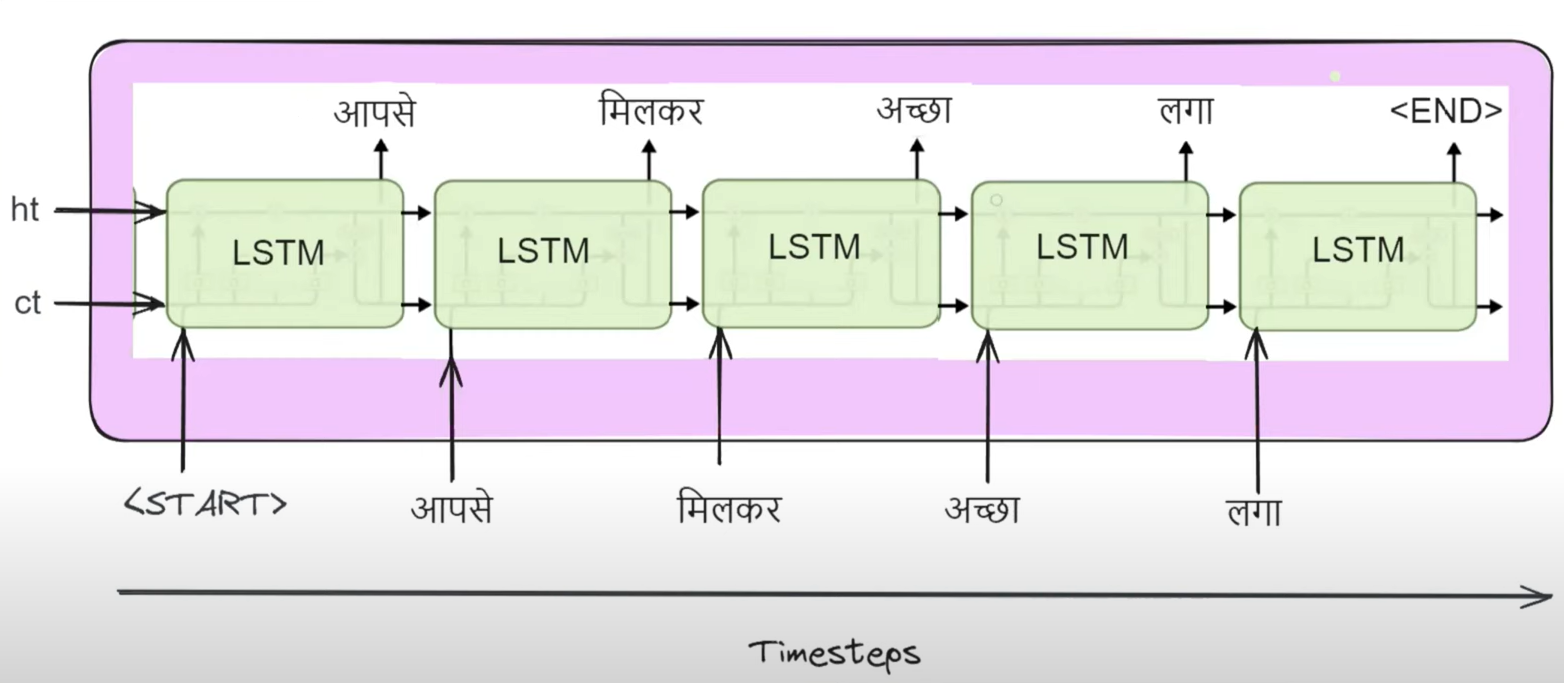
* This architecture consist two block encoder & Decoder conected through context vector.
* Encoder try to understond / summarize the I/P and give a o/p vectar this vector is called a context vector.
* Decoder try to understand this context vector after then print OlP word/ token by word basis.

**Encoder**



* The summarization pass through hidden state and cell state that become context vector

**Decoder**



**Training the Architecture using Backpropagation**

Dataset → Machine Translation

|  |  |  |
| --- | --- | --- |
| S. No | English | Hindi |
| 1 | Think about it. | सोचो लो |
| 2 | Come in | अंदर आ जाओ |

1. First convert the dataset into toxen (Tokenization)

[Think – 1, about – 2, it – 3, come – 4, in - 5］| [सोचो – 1, लो - 2, अंदर – 3, आ – 4, जाओ - 5]

1. Second we have to convert into numbers, we have So many option we use for example one-hot -encoding

|  |  |
| --- | --- |
| Eng | Hindi |
| Think → [ 1,0,0,0,0]  About → [0,1, 0,0,0]  …….  In → [0, 0, 0, 0, 1] | <Start> → [1, 0, 0, 0, 0, 0, 0]  सोचो 🡪 [0,1, 0, 0, 0, 0,0]  ……….  जाओ 🡪 [0,0, 0, 0, 0, 1,0]  <End> 🡪 [0,0, 0, 0, 0, 0,1] |

Row 1→ [Think about it 🡪 सोचो लो]

**Decoder**

yact = [0, 0, 1, 0, 0, 0, 0] (लो)

yhat = [0.1, 0.2, 0.3, 0.2, 0.1, 0.1, 0.15] (जाओ)

yact = [0, 0, 0, 0, 0, 0, 1] (<END>)

yhat = [0.2, 0.15, 0.3, 0.2, 0.1, 0.1, 0.5] (<END>)

yact = [0, 1, 0, 0, 0, 0, 0] (सोचो)

yhat = [0.2, 0.1, 0.3, 0.2, 0.1, 0.5, 0.15] (लो)

Softmax

Softmax

Softmax

Teacher Forcing

Teacher Forcing

LSTM

LSTM

LSTM

Context

Vector

ht

ct

ht

ct

LSTM

LSTM

LSTM

T1 T2 T3 T1 T2 T3

Think about it

[1,0,0,0,0] [0,1,0,0,0] [0,0,1,0,0]

<Start> सोचो लो

**Teacher Forcing**: During training we always insert correct I/P weather the prediction is wrong or correct This technique we called Teacher Forcing.

1. Loss calculates

LOSS → Categorical Cross entropy

Total Loss = 1+1+0.39 = 2.39

1. Back propagation
   1. Gradient calculation
   2. Update Parameters

Gradient → Calculate the gradient of the loss with each of the trainable (LSTM + DL+ Softmax) parameters.

* The gradient represent /measure how much each parameter contributes in Loss and in which direction so we can adjust those parameters so that Loss will minimize.
* Aften calculating. gradient we update the weights on the basis of those gradient.
* To Perform this task, we will Use optimizer like SGD, Adam, RMS Prop etc.
* Adjustment of parameter weight basis on learning rate.

**Prediction**

<start> 1, सोचो – 2, लो - 3, अंदर – 4, आ – 5, जाओ – 6, <end> - 7

Let’s assume we will give a new sentence.

|  |  |  |  |
| --- | --- | --- | --- |
| सोचो | जाओ | लो | < end> |
| [0.1, 0.3, 0.25, 0.13, 0.1, 0.21, 0.11] | [0.2, 0.1, 0.1, 0.1, 0.2, 0.3, 0.1] | [0.1, 0.1, 0.4, 0.2, 0.11, 0.09, 0.1] | [0.1, 0.1, 0.2, 0.1, 0.3, 0.25, 0.35] |

Softmax

LSTM

LSTM

LSTM

LSTM

Context

Vector

ht

ct

ht

ct

LSTM

LSTM

LSTM

Think about it

[1,0,0,0,0] [0,1,0,0,0] [0,0,1,0,0]

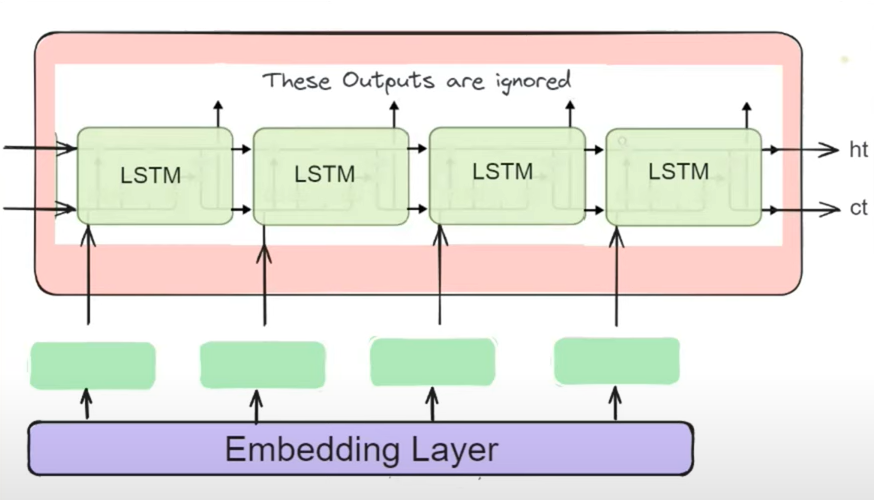
<Start> सोचो जाओ लो

Vector vector vector

I/P → Think about it ------- O/P 🡪 सोचो जाओ लो

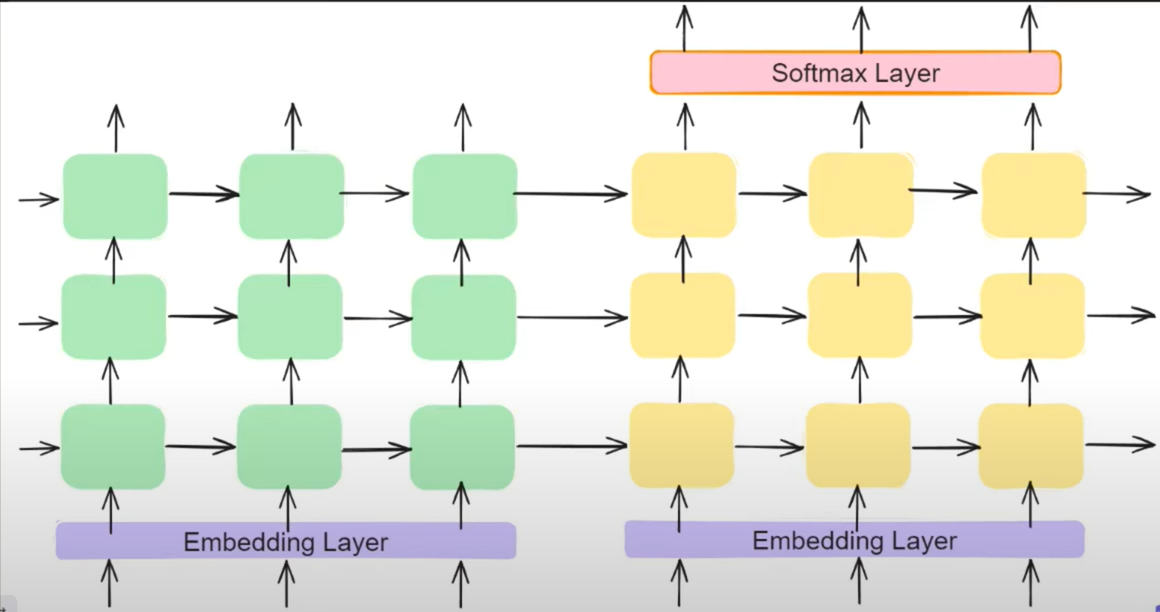
**Improvement 1 → Embedding**

Embedding is low dimensional representation & Dense used in both encoder & Decoder.



For Embedding we will use 🡪 word2vec or Glove

**Improvement 2→ Deep LSTM’s**



Advantage

1. Handle Long term dependency / Long Sentence property / Efficiently

why → More context vector are Capable to store summary of I/P

1. It able to understand Layered Representation. This is very useful to understand hierarchical data

Paragraph level understanding

Sentence Level understanding

Word Level understanding

LSTM

LSTM

LSTM

1. Whenever we increase the parameters of DL NN so it increases the learning capability 🡪 Better Generalization