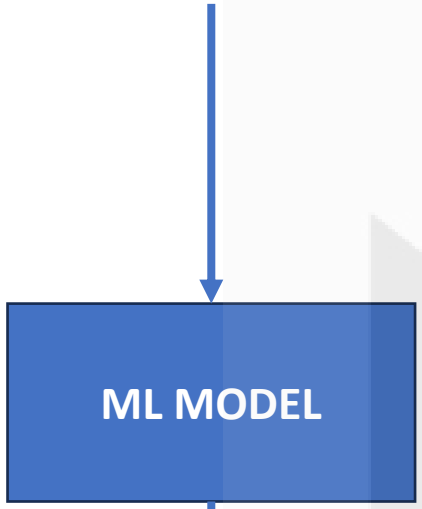
Seq2Seq Models

MUKESH KUMAR

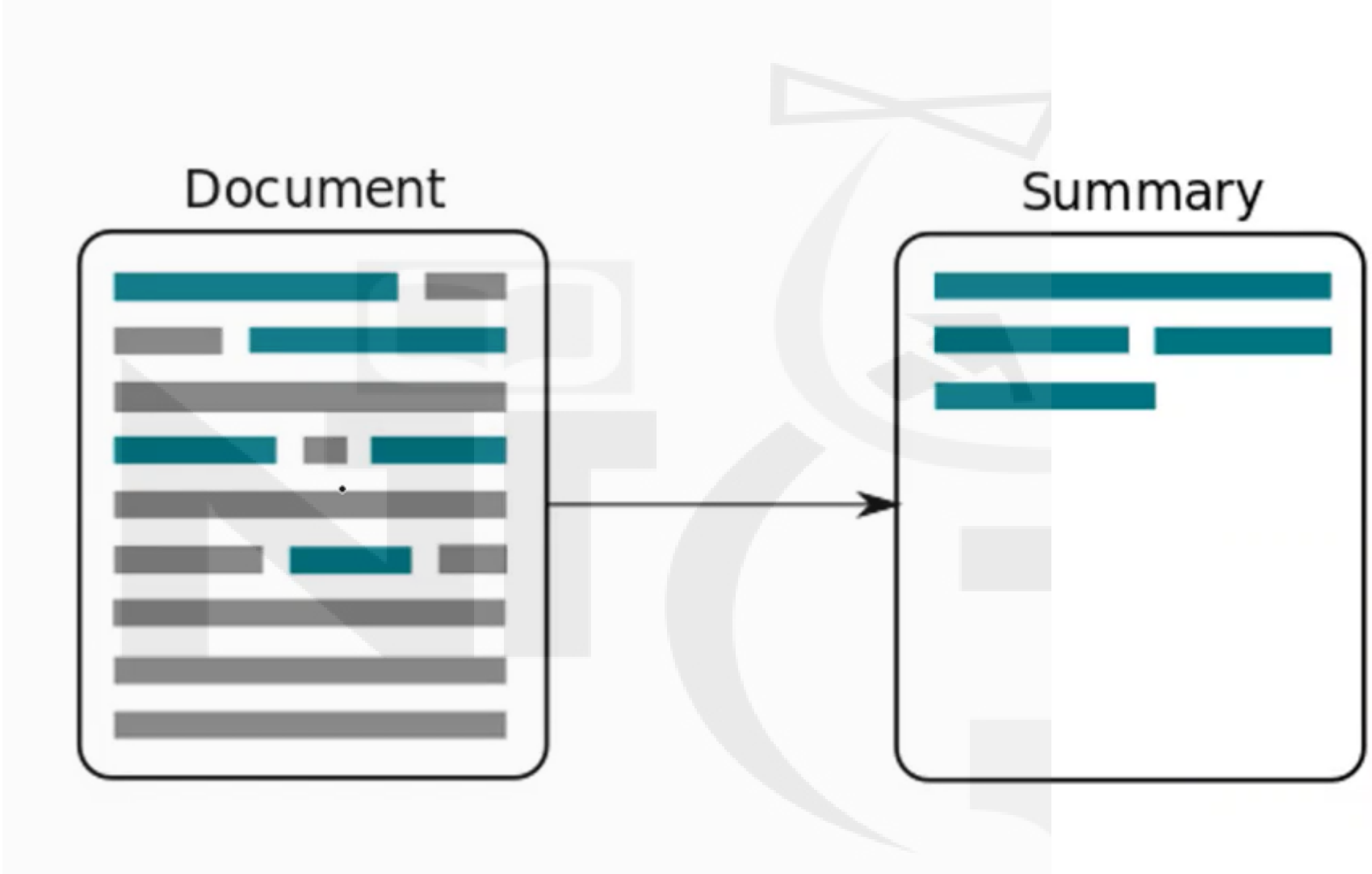
Sequence of words in audio form (input)



Sequence of words in TEXT form (output)

Convert Speech to Text

e.g Alexa, Siri, Google Home



Large sequence of words (input)

ML MODEL

Small Sequence of words (output)

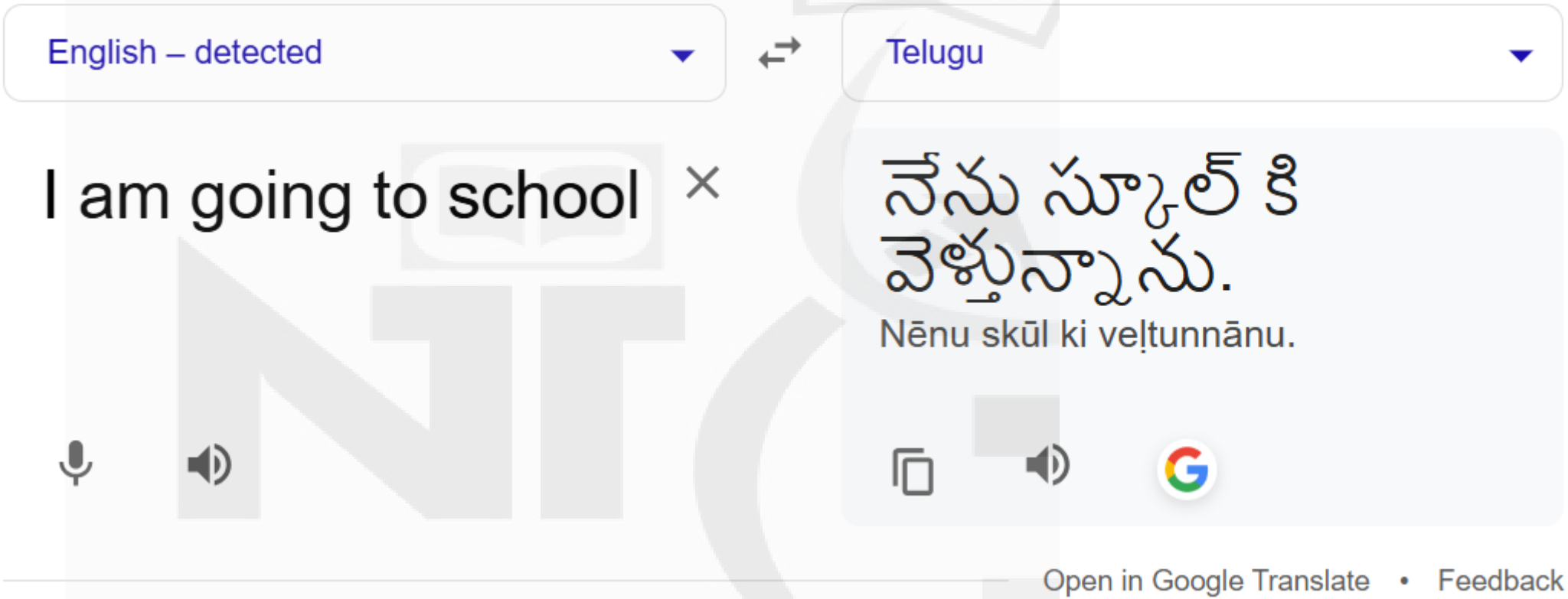
Describe a picture

Sequence of PIXEL
(input)

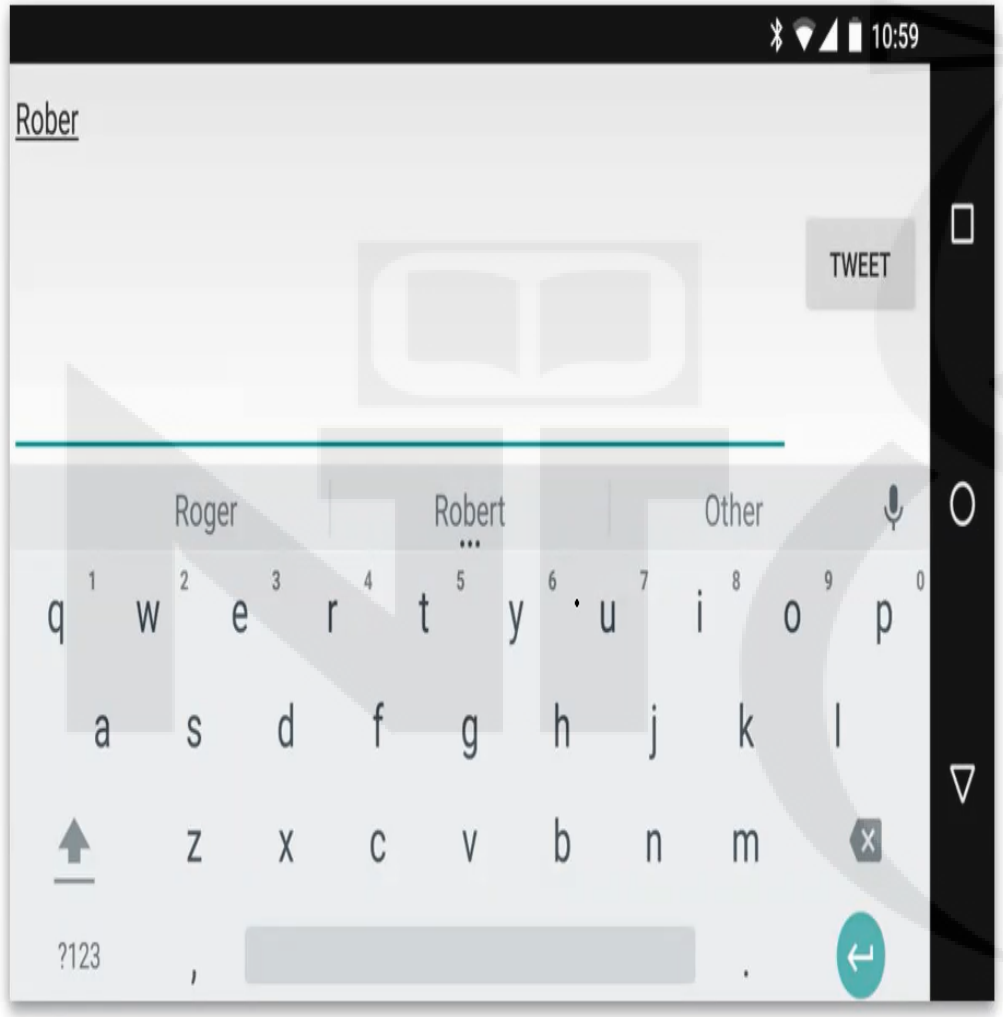
ML MODEL

Sequence of words
(output)

Tendulkar playing cricket

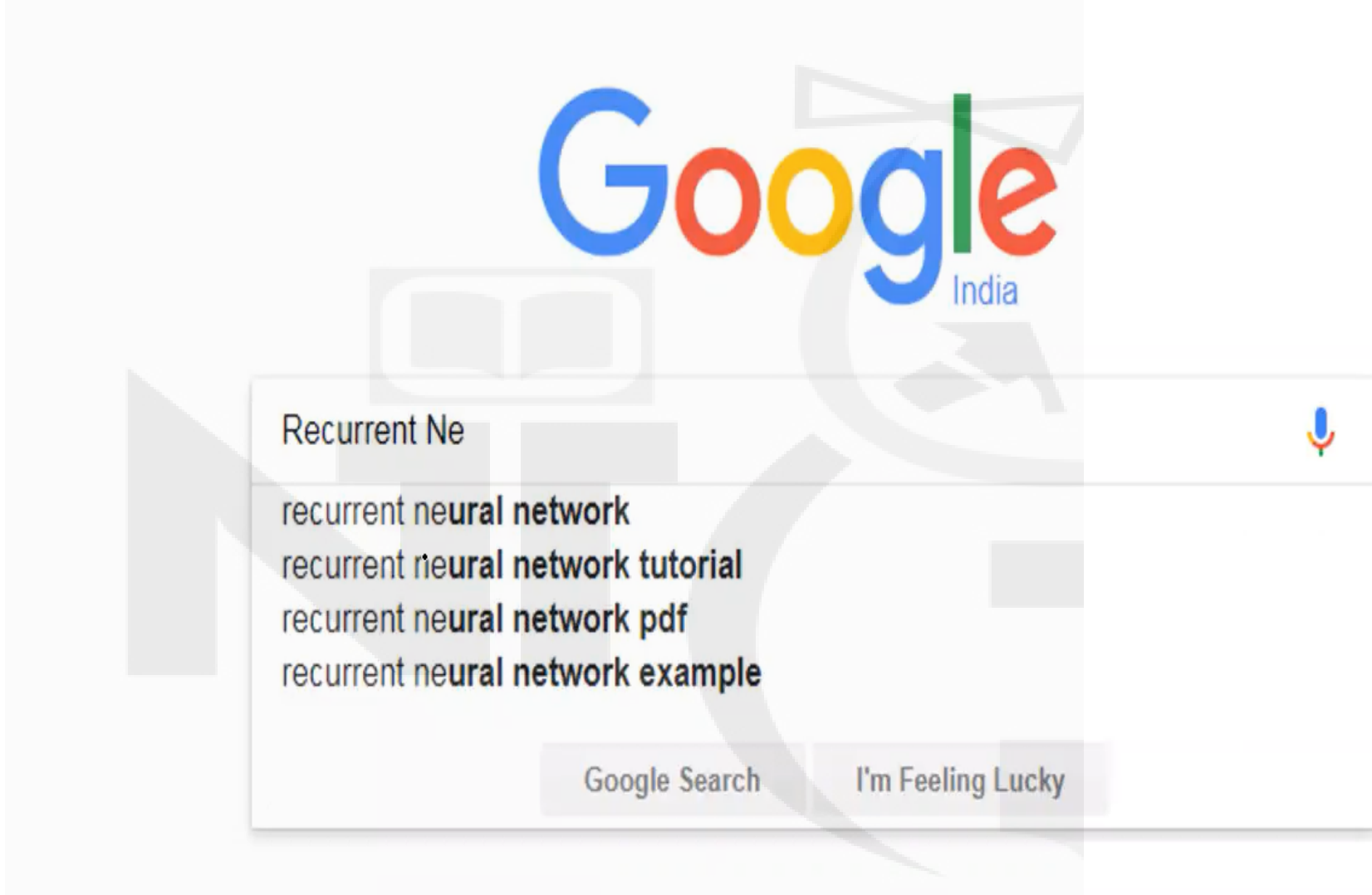


How to Translate???



Mobile phone keyboard

Predict next word(s) as we type in...

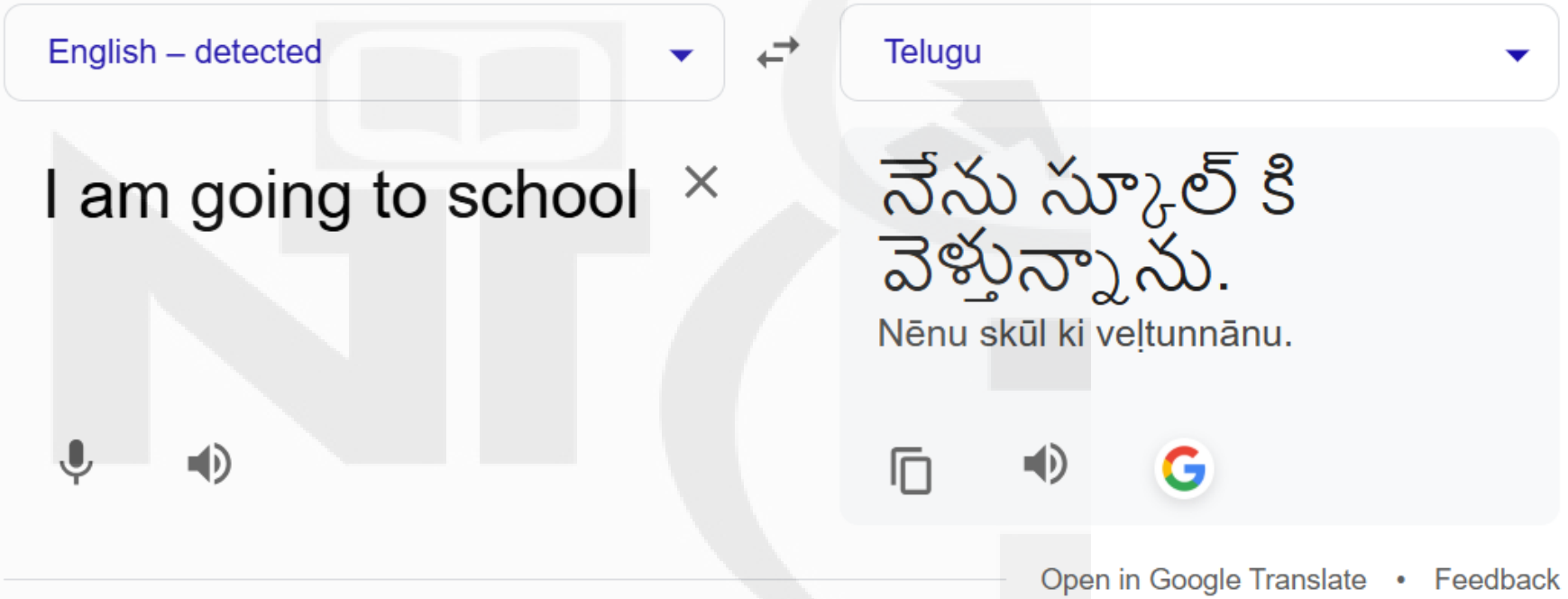


Search suggestions

How to build models that convert a sequence into another sequence?

Understanding Language Translations

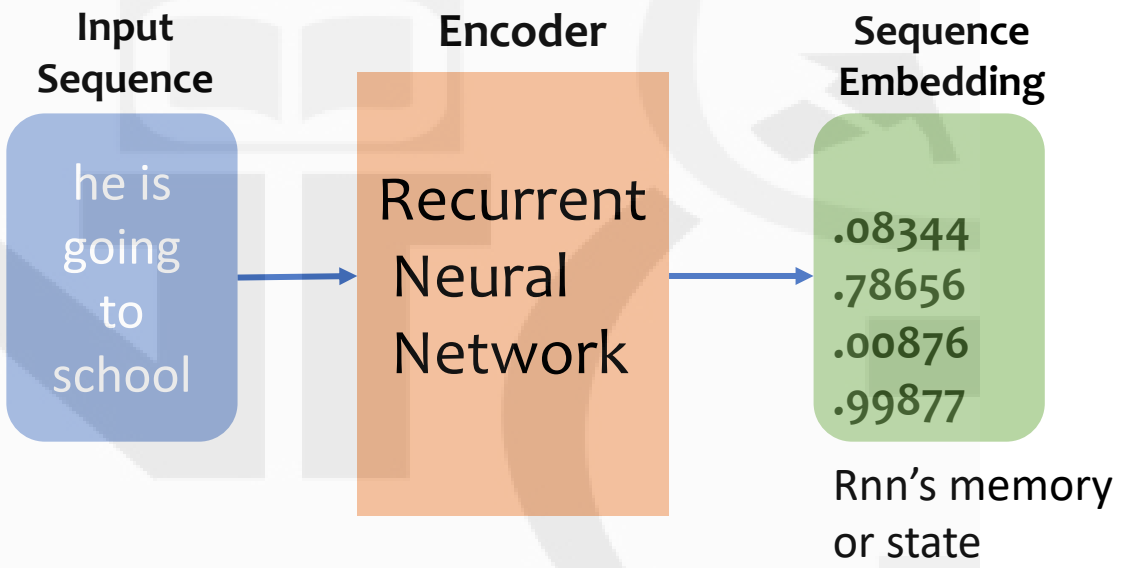
Our input will be a sentence in one language and our model should be able to translate it to another language



Sequence of words in Source language
INPUT

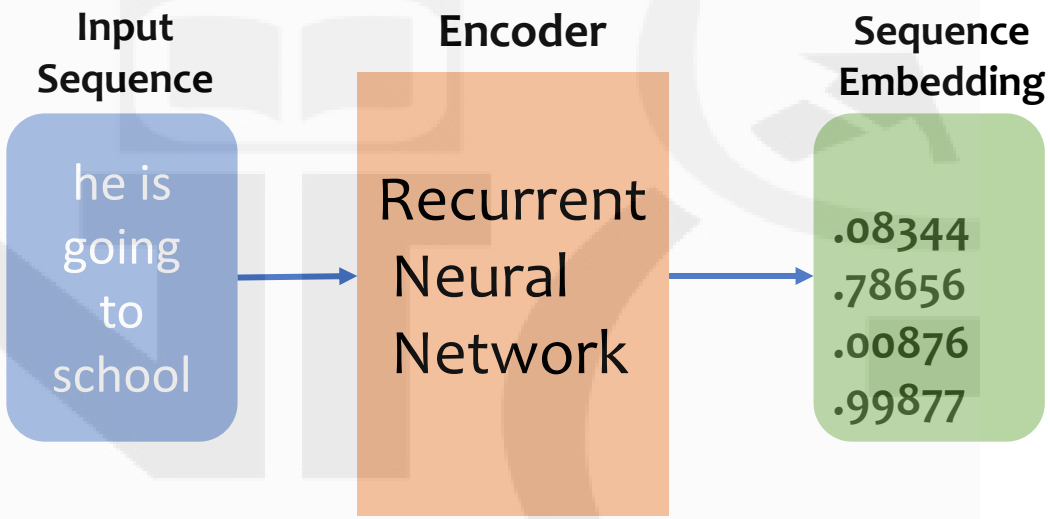
Sequence of words in Target language
OUTPUT

Which model works good with sequences??



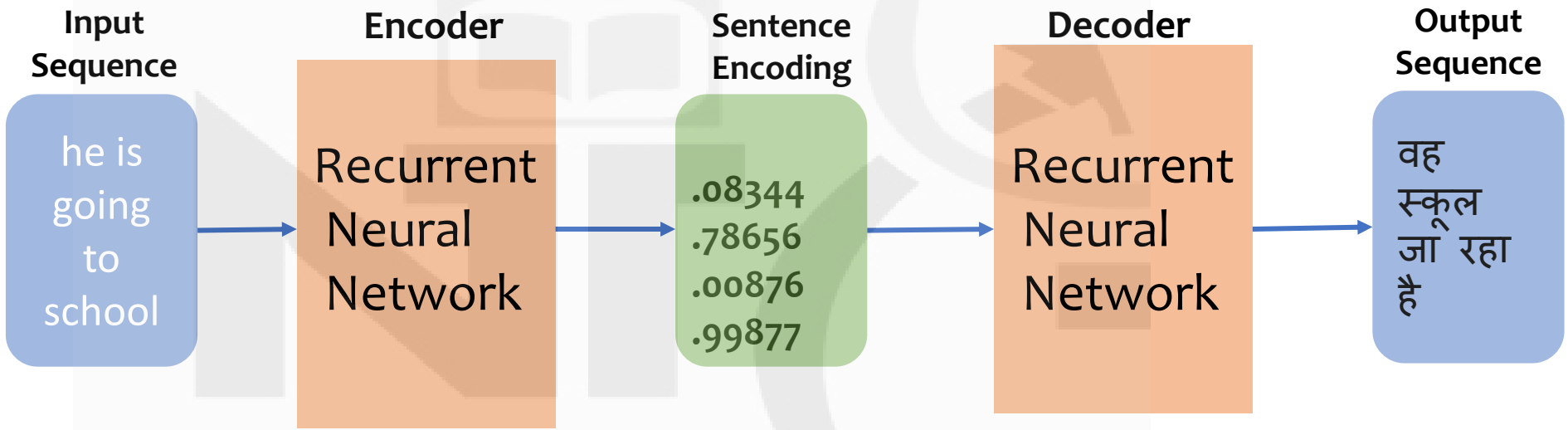
Learning the input sequence

LSTM are very good at understaing a sequence



How do we generate the output sequence from sequence embedding??

Then we will take another RNN convert these embeddings into an output sequence



Sequence to Sequence (Seq2Seq) Model

Seq2Seq Model

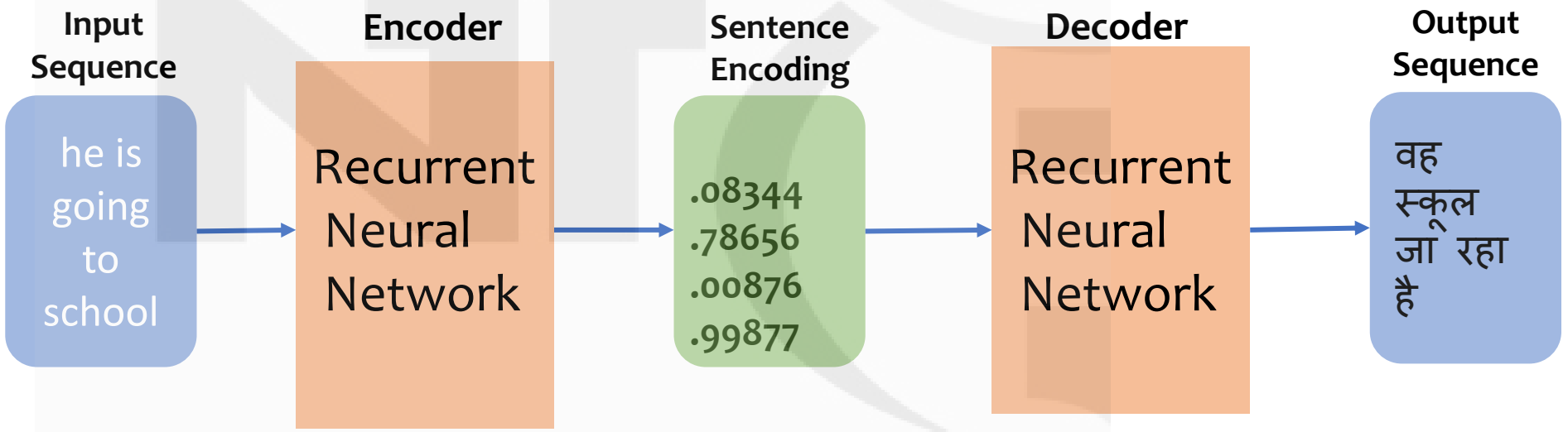
- Google came up with this architecture
- For language translation
- It replaced half a million lines of code with just few hundred lines
- And also, the accuracy was much better replacing human linguistics for translation

No Need to know any rule about human language

Achieved better results than 20 years of work with Statistical Machine Translation

Neural Machine Translation (NMT)

Building Seq2Seq Model in Keras



We will need data first

COLLECT TRANSLATION DATA

https://www.manythings.org/anki/

https://www.manythings.org/anki/

ManyThings.org Reading Sentences More

Tab-delimited Bilingual Sentence Pairs

These are selected sentence pairs from the Tatoeba Project.

Updated: 2024-04-01

- Afrikaans - English afr-eng.zip (918)
- Albanian - English sqi-eng.zip (449)
- Algerian Arabic - English arq-eng.zip (155)
- Arabic - English ara-eng.zip (12523)
- Armenian - English hye-eng.zip (1804)
- Assamese - English asm-eng.zip (4163)
- Azerbaijani - English aze-eng.zip (2191)
- Basque - English eus-eng.zip (683)
- Belarusian - English bel-eng.zip (3938)
- Bengali - English ben-eng.zip (6509)
- Berber - English ber-eng.zip (152163)

Introducing Anki

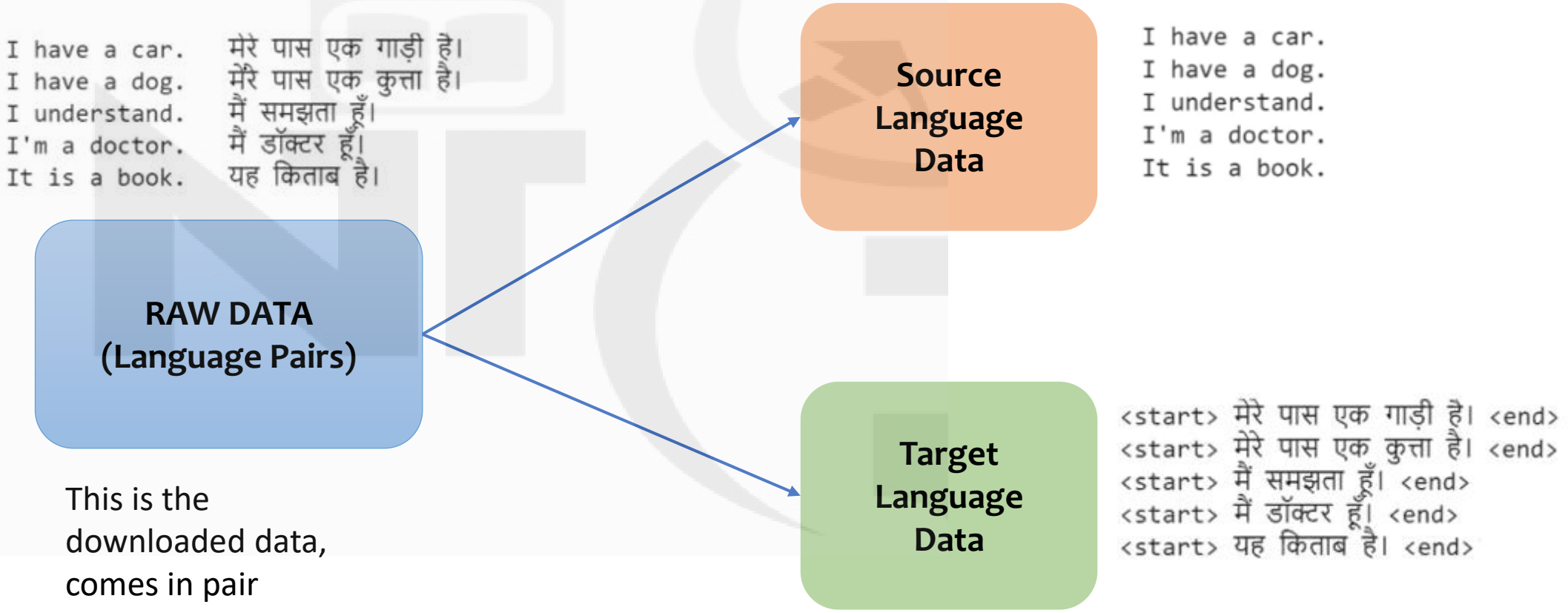
- If you don't already use Anki, vist the website at http://ankisrs.net/ to dov free application for Macintosh, Windows or Linux.

About These Files

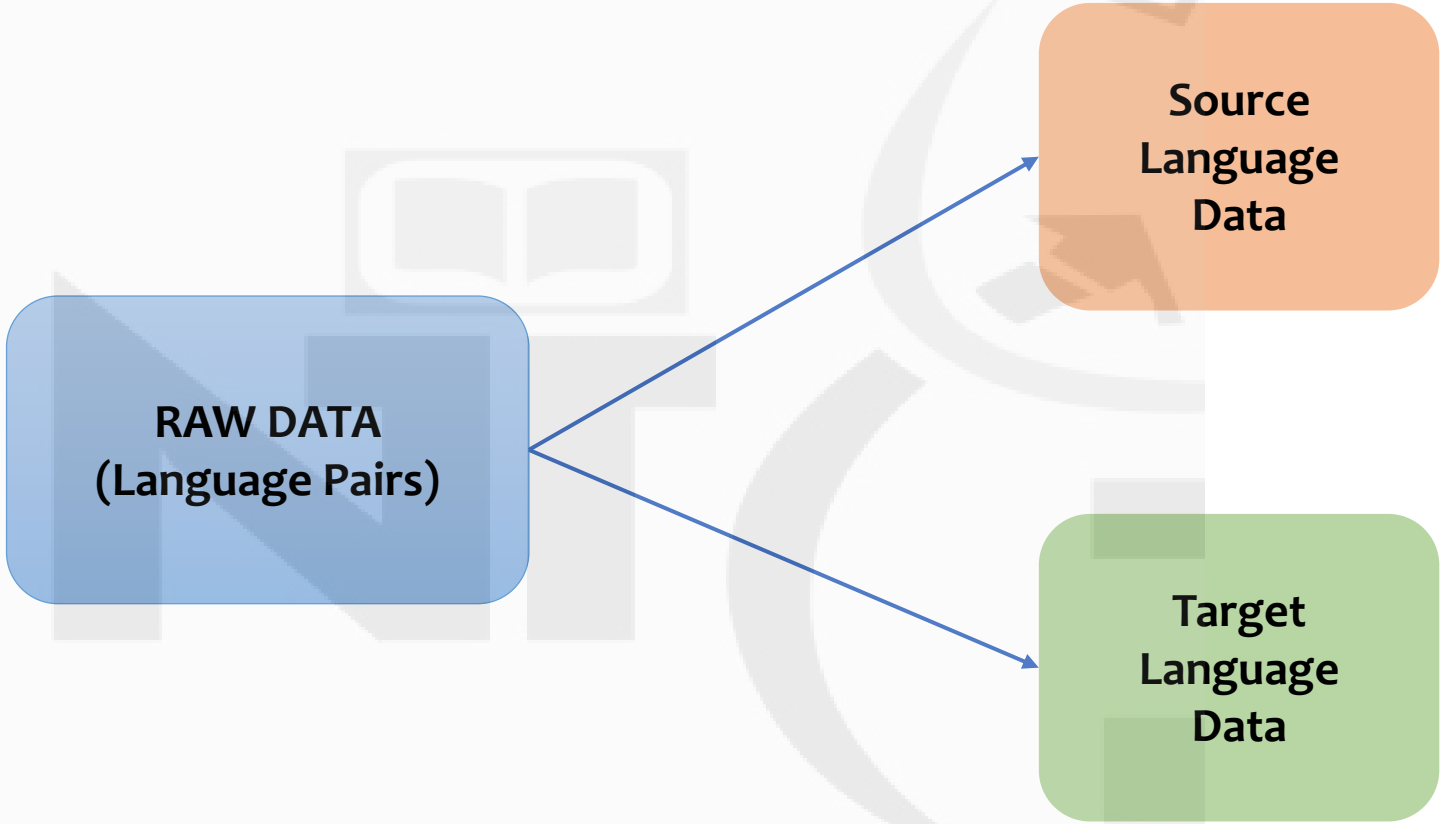
- Any flashcard program that can import tab-delimited text files, such as A can use these files.
- **Warning!** There are errors in the Tatoeba Corpus. (Detailed Warning)
- **In order to minimize the number of errors**, I only used sentences that owned by identified native speakers working on the Tatoeba Project and sentences that I've personally checked and did not reject.
- **Warning!** Please remember that even doing this may not have eliminate errors.

Let’s understand Data Preprocessing

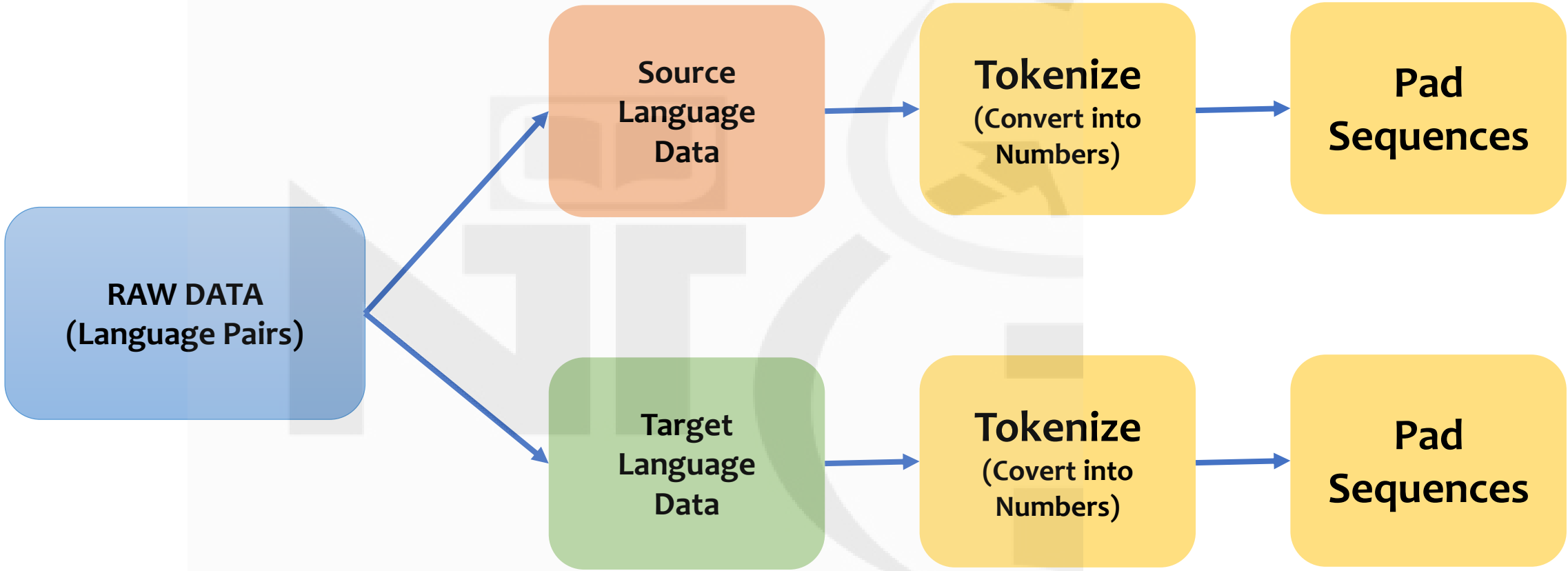
We need to split the downloaded data



This is the downloaded data, comes in pair

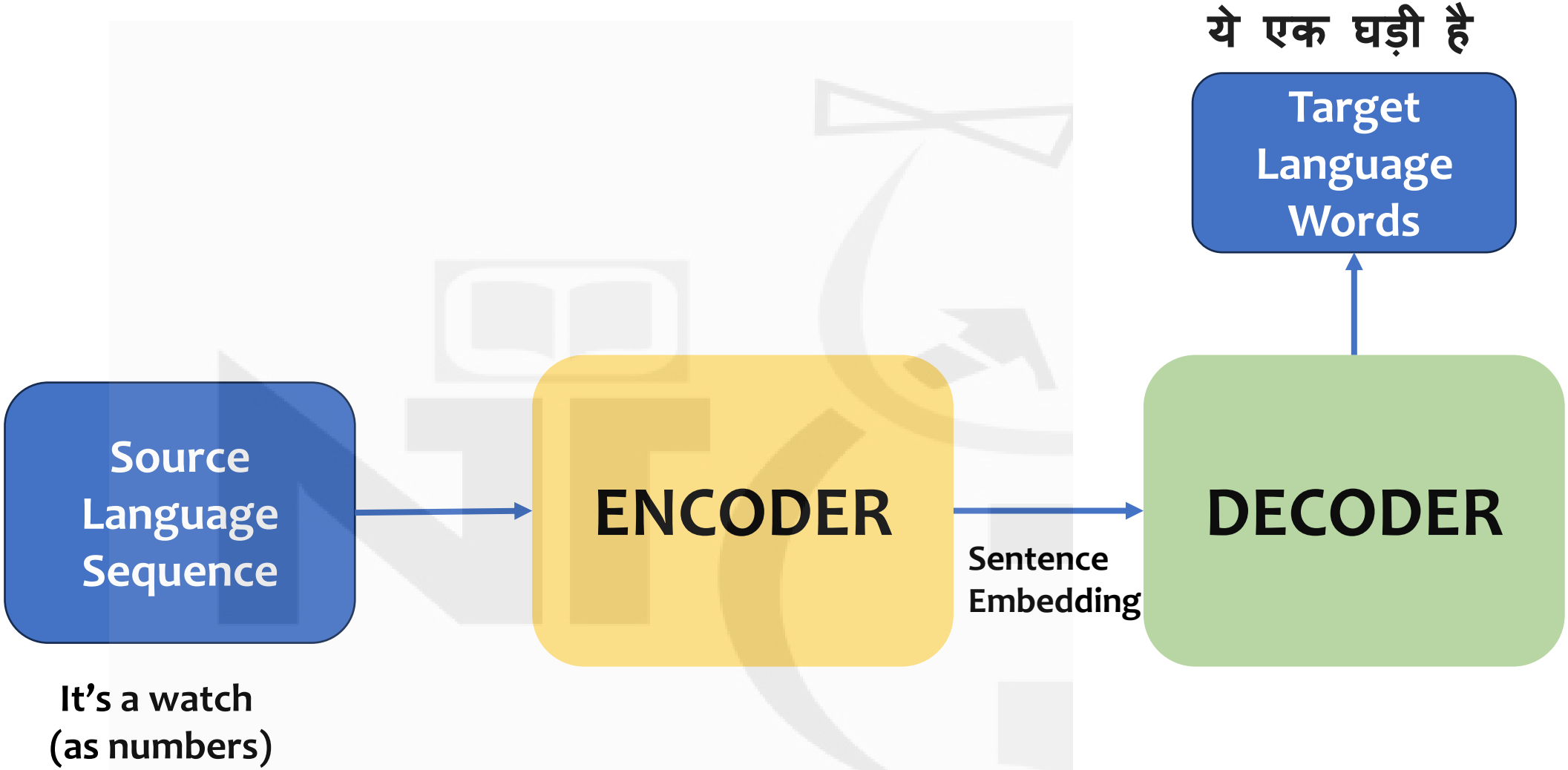


For Target Sequence we will use start and end sequence – useful during predictions

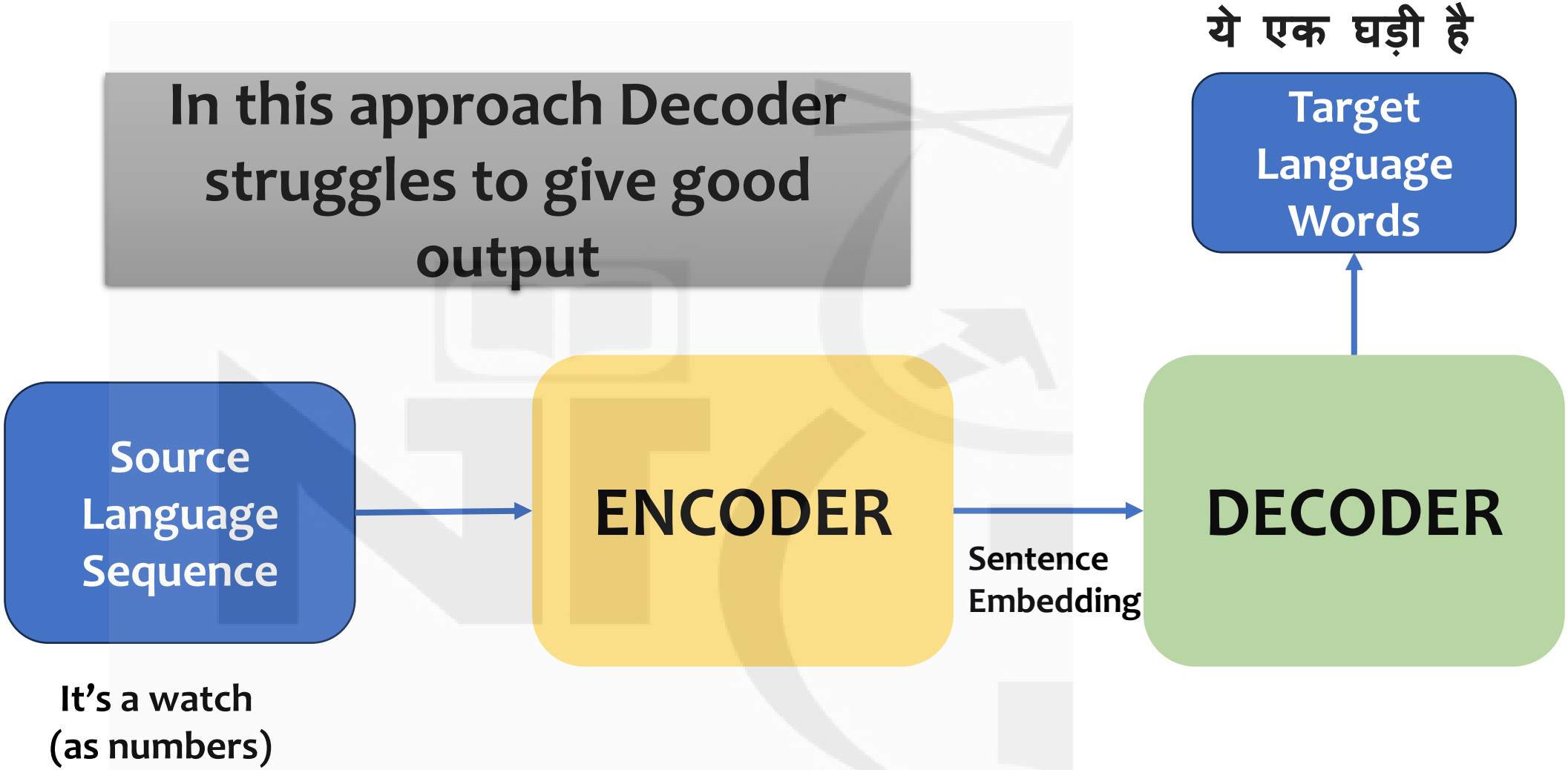


Make all sentences same size!

Building Seq2Seq Model



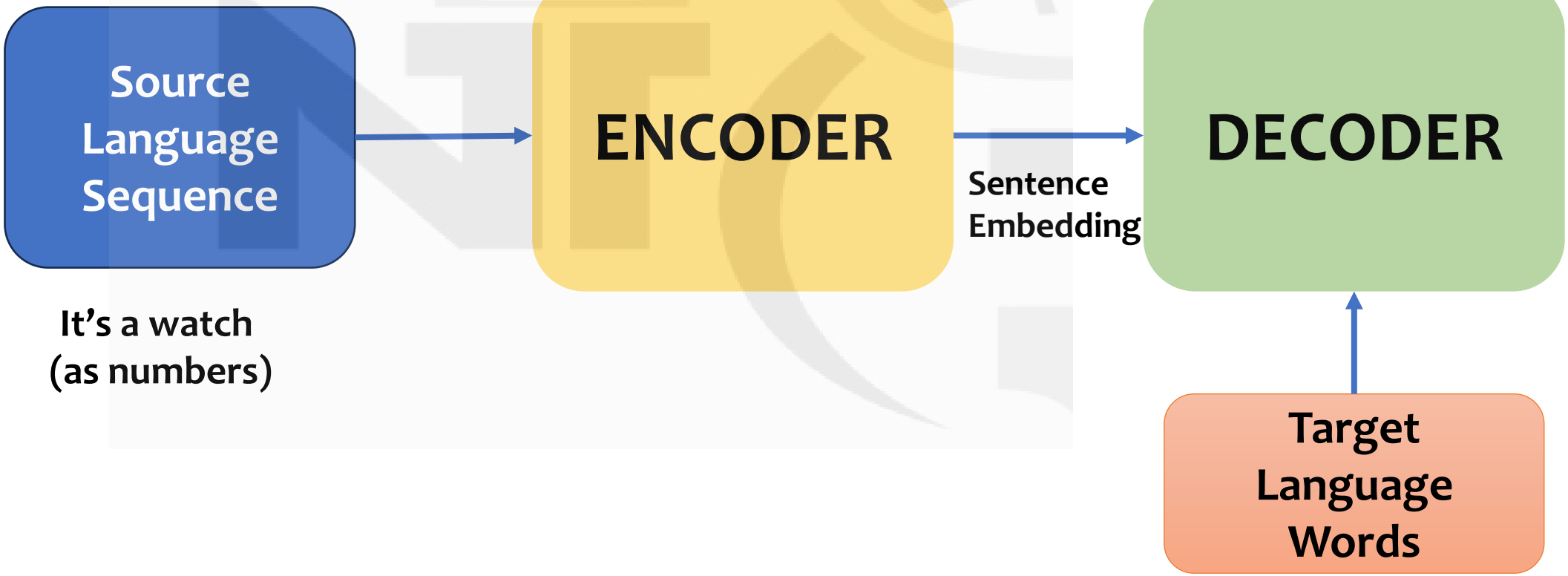
Sequence to Sequence (Seq2Seq) Model



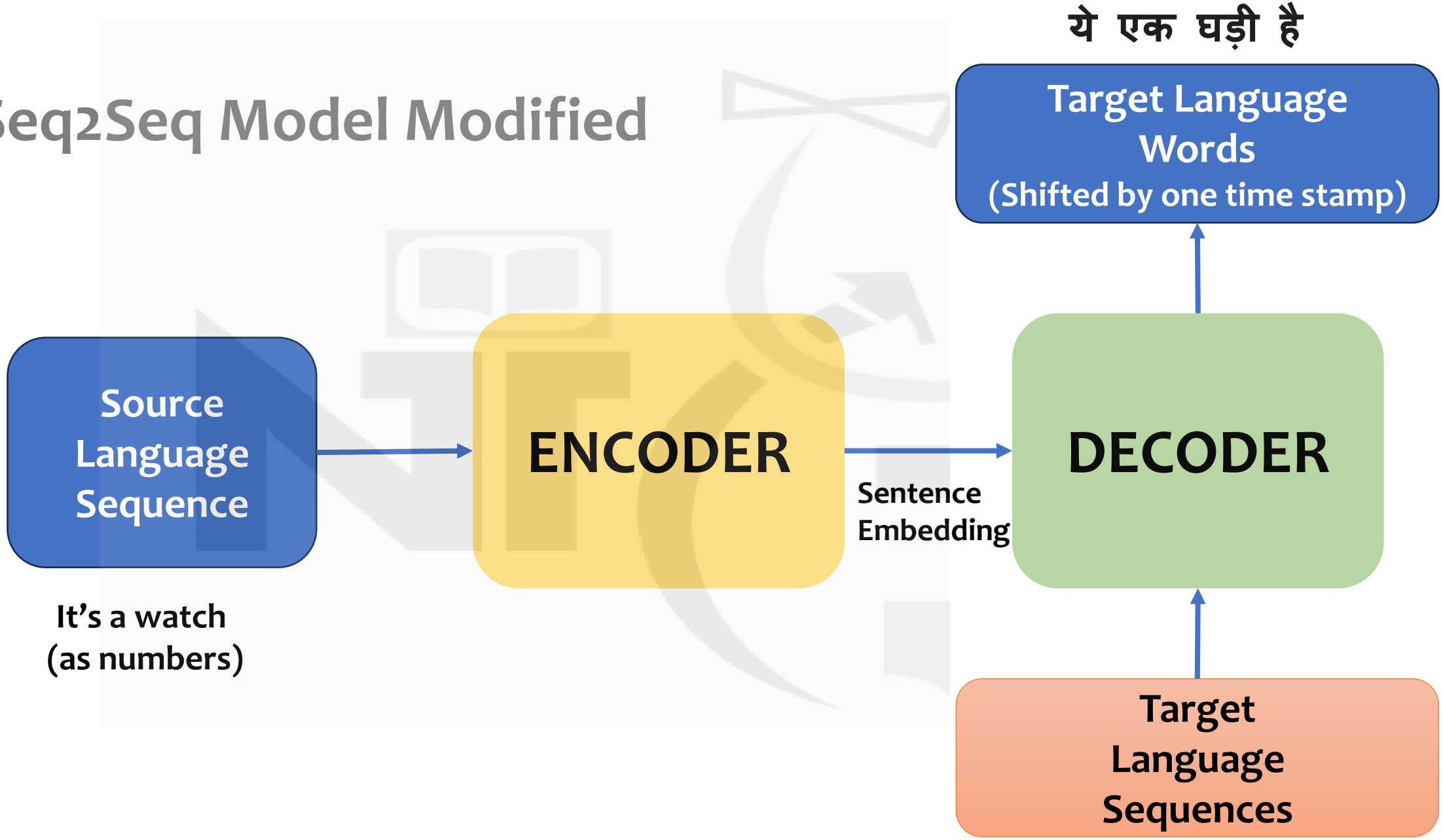
Sequence to Sequence (Seq2Seq) Model

- This approach doesn’t usually work well, Decoder struggles
- To help the decoder we feed the actual hindi (target language) sentence as input
- And then we ask the decoder now give me the hindi (target language) sentence
- If the decoder is taking output hindi sentence as input and outputs the same thing then what is decoder doing?
- But the output of decoder is missing the start tag
- So its not an exact copy of input so there is some learning in decoder

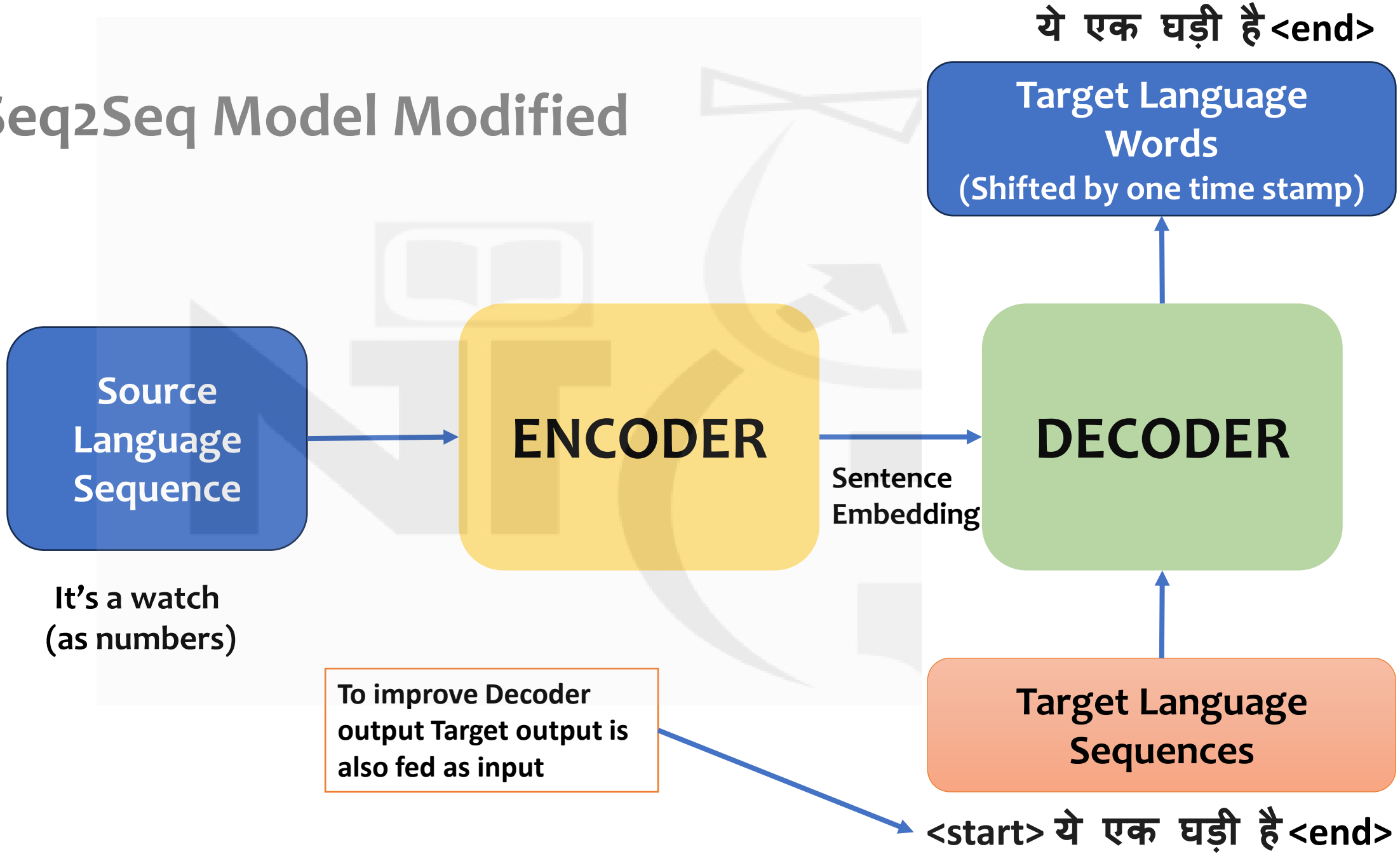
Seq2Seq Model Modified



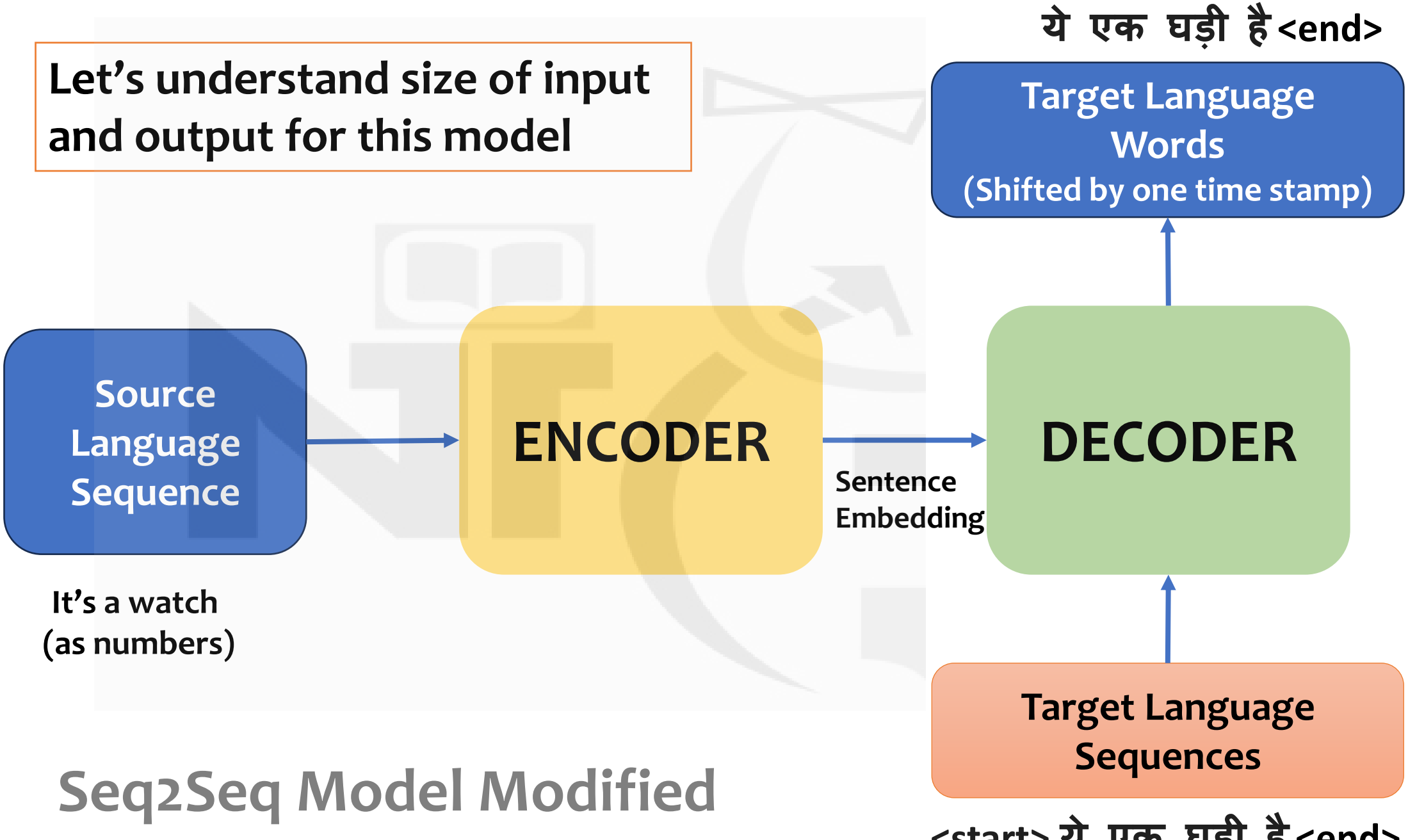
Seq2Seq Model Modified



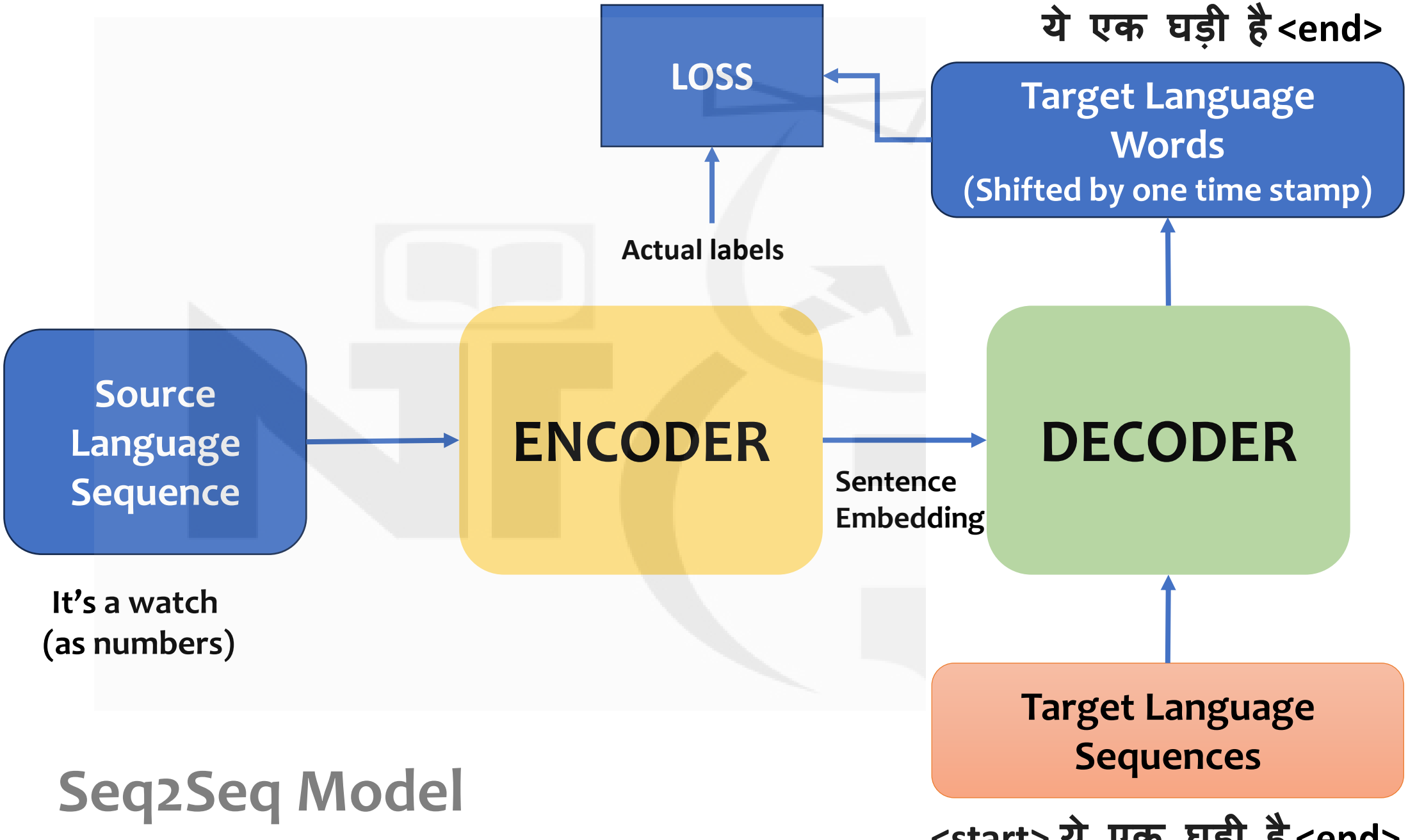
Seq2Seq Model Modified



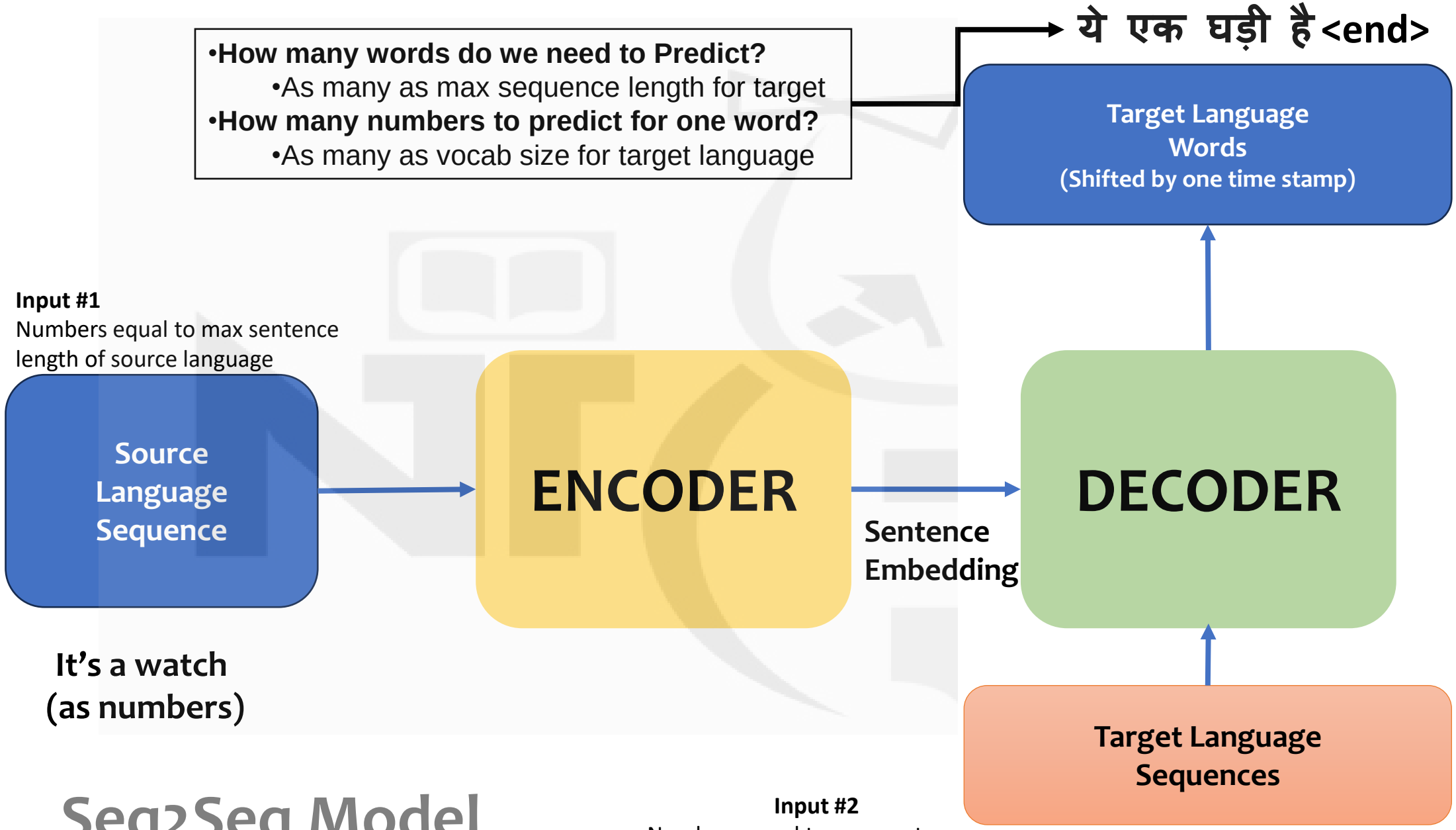
Let's understand size of input and output for this model



Seq2Seq Model Modified

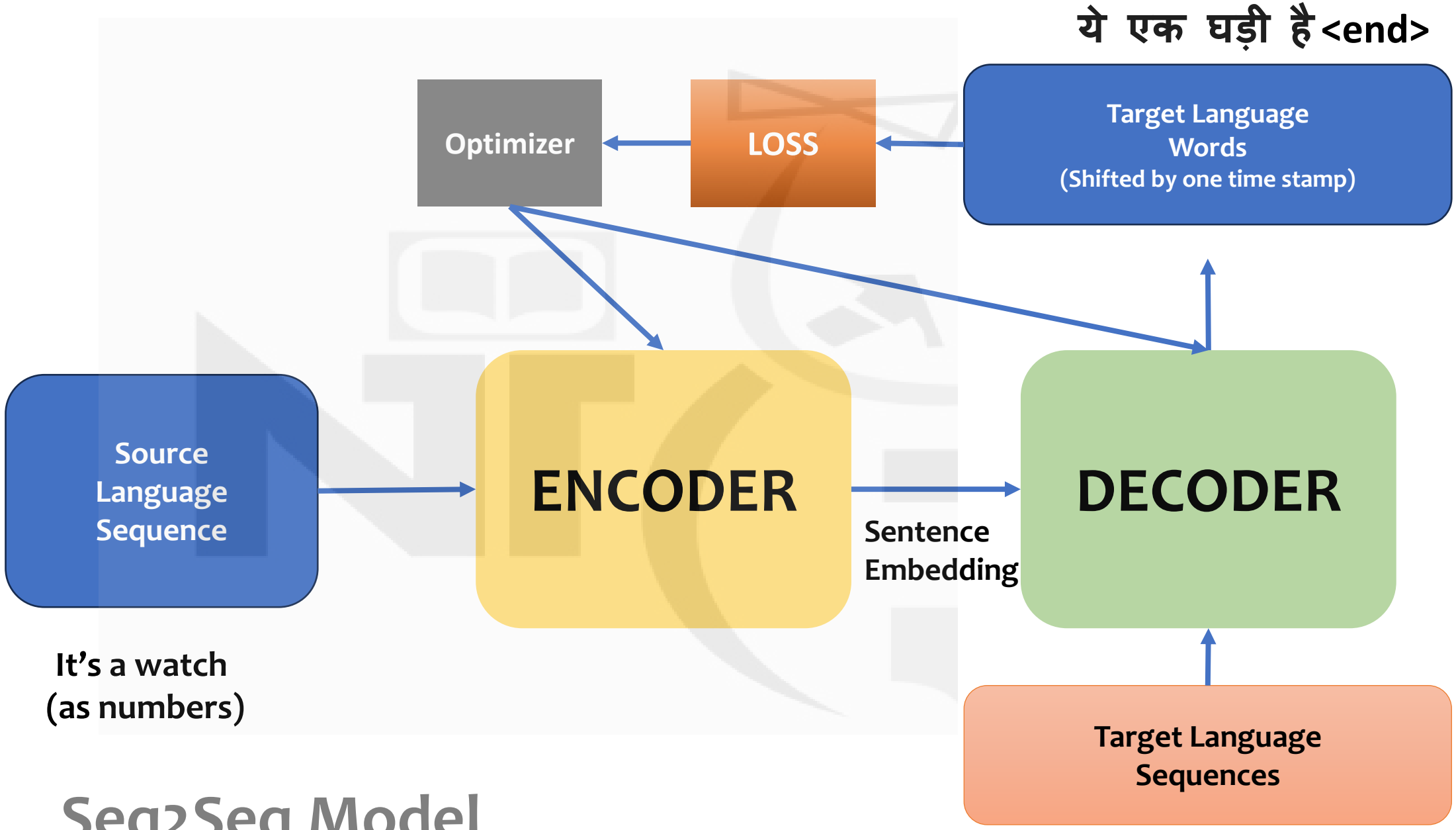


Seq2Seq Model



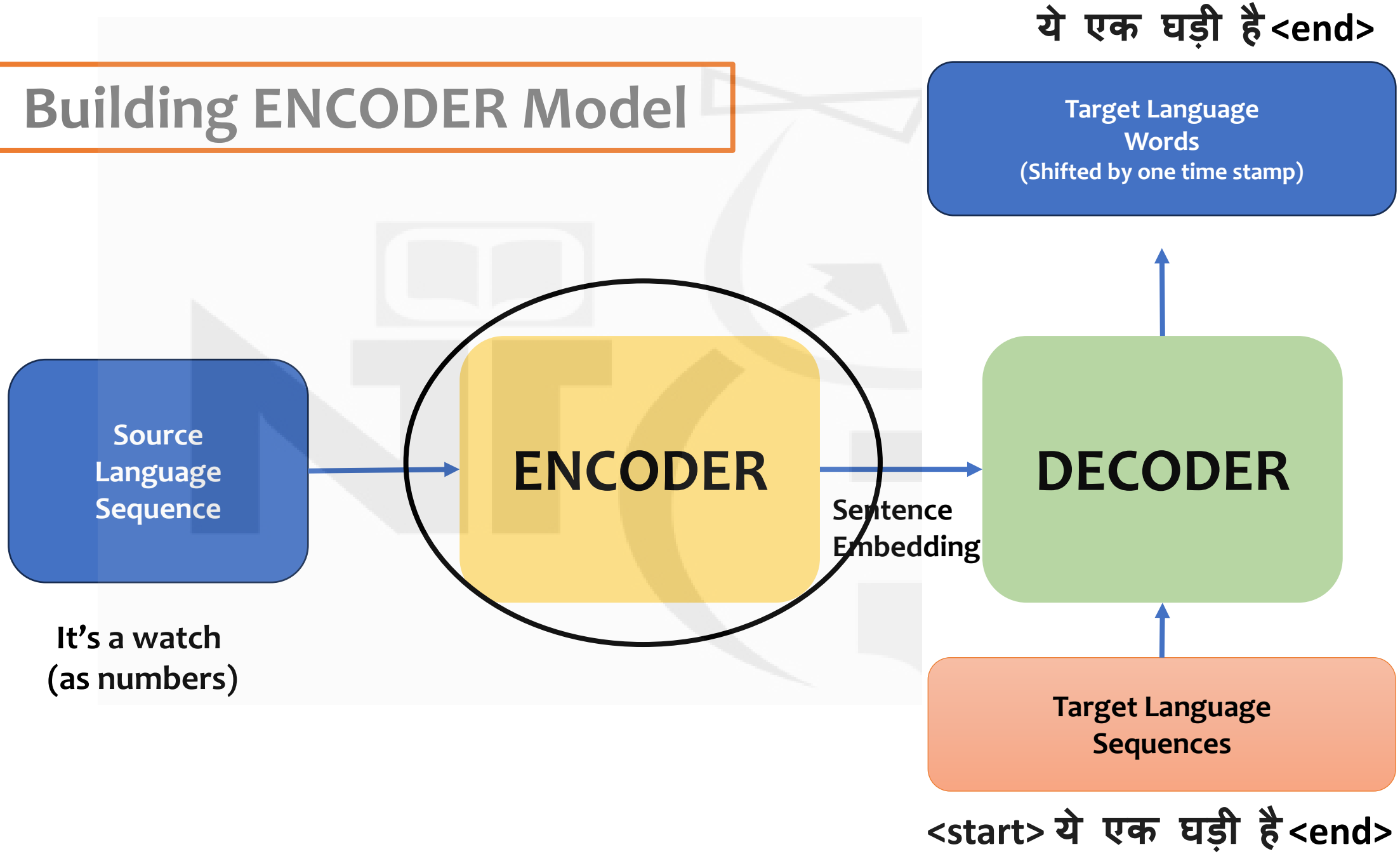
Seq2Seq Model

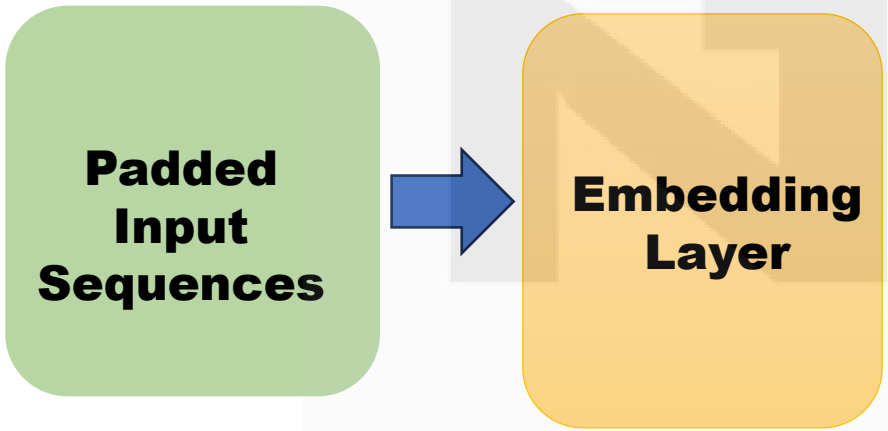
Build the model



Seq2Seq Model

Building ENCODER Model

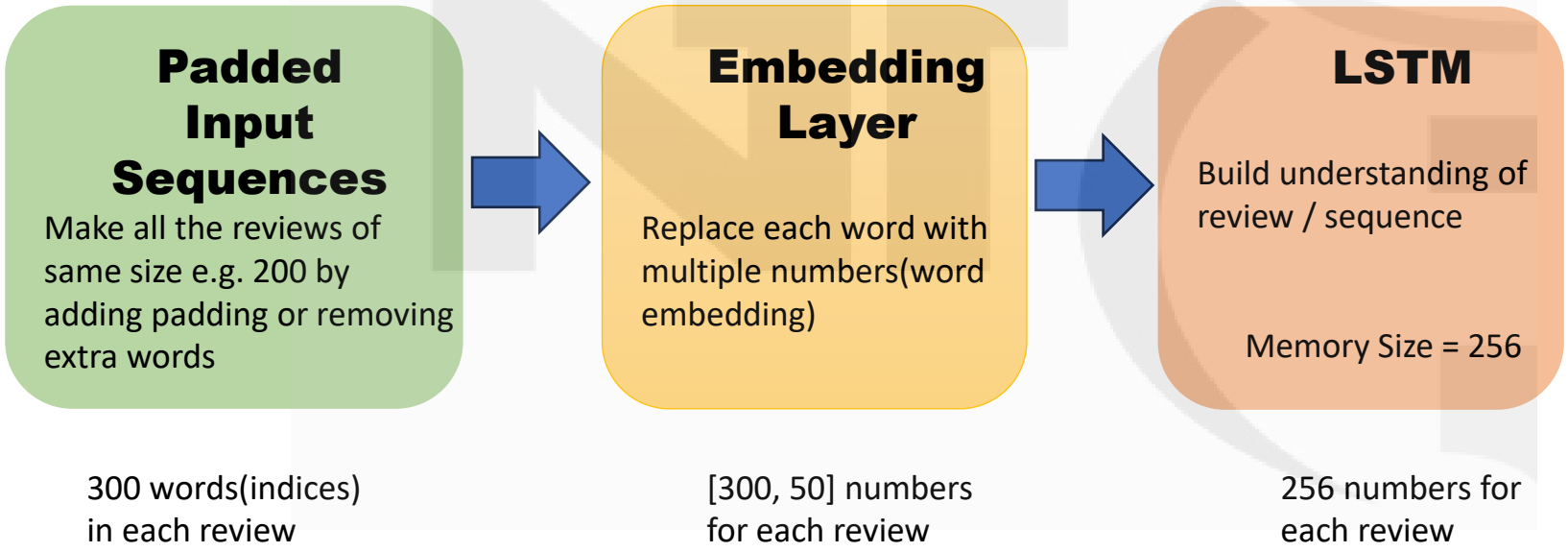




ENCODER

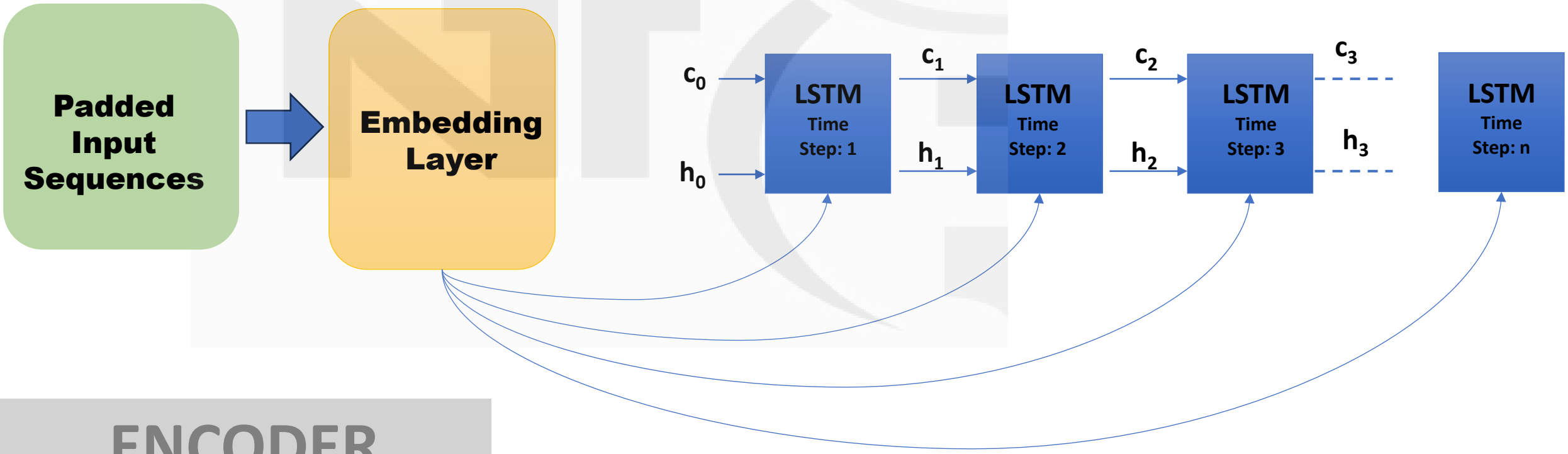
Next?

- We calculate the loss

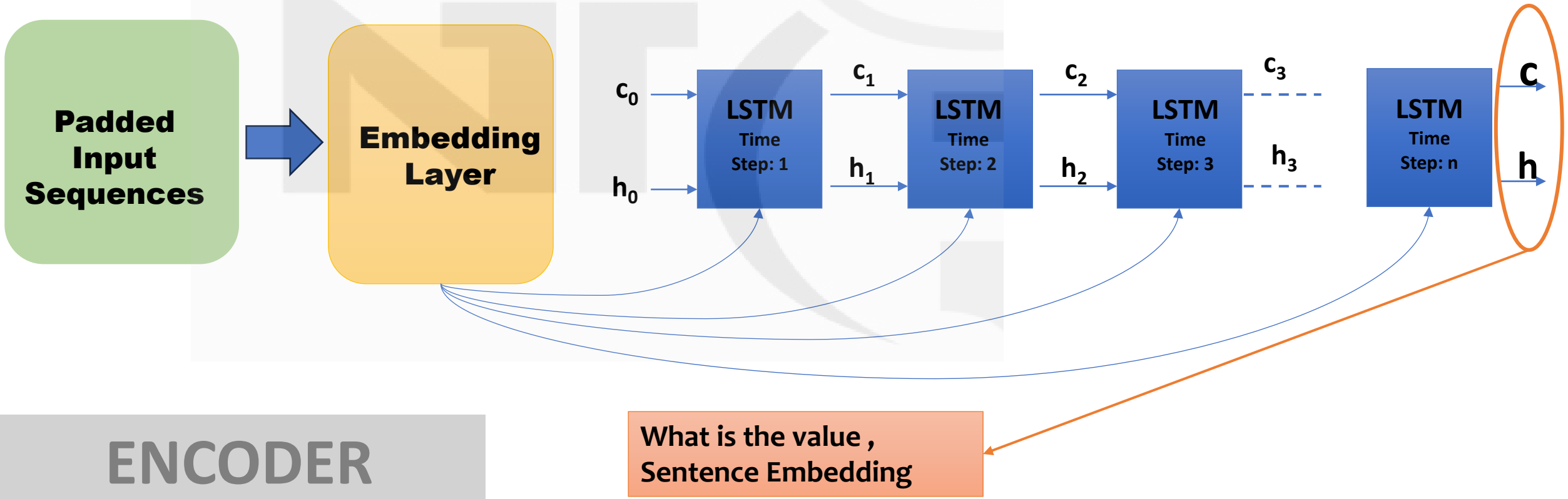


We will use LSTM to understand input sentence

ENCODER


Padded Input Sequences
Embedding Layer
c_0
h_0
LSTM Time Step: 1
c_1
h_1
LSTM Time Step: 2
c_2
h_2
LSTM Time Step: 3
c_3
h_3
LSTM Time Step: n

ENCODER

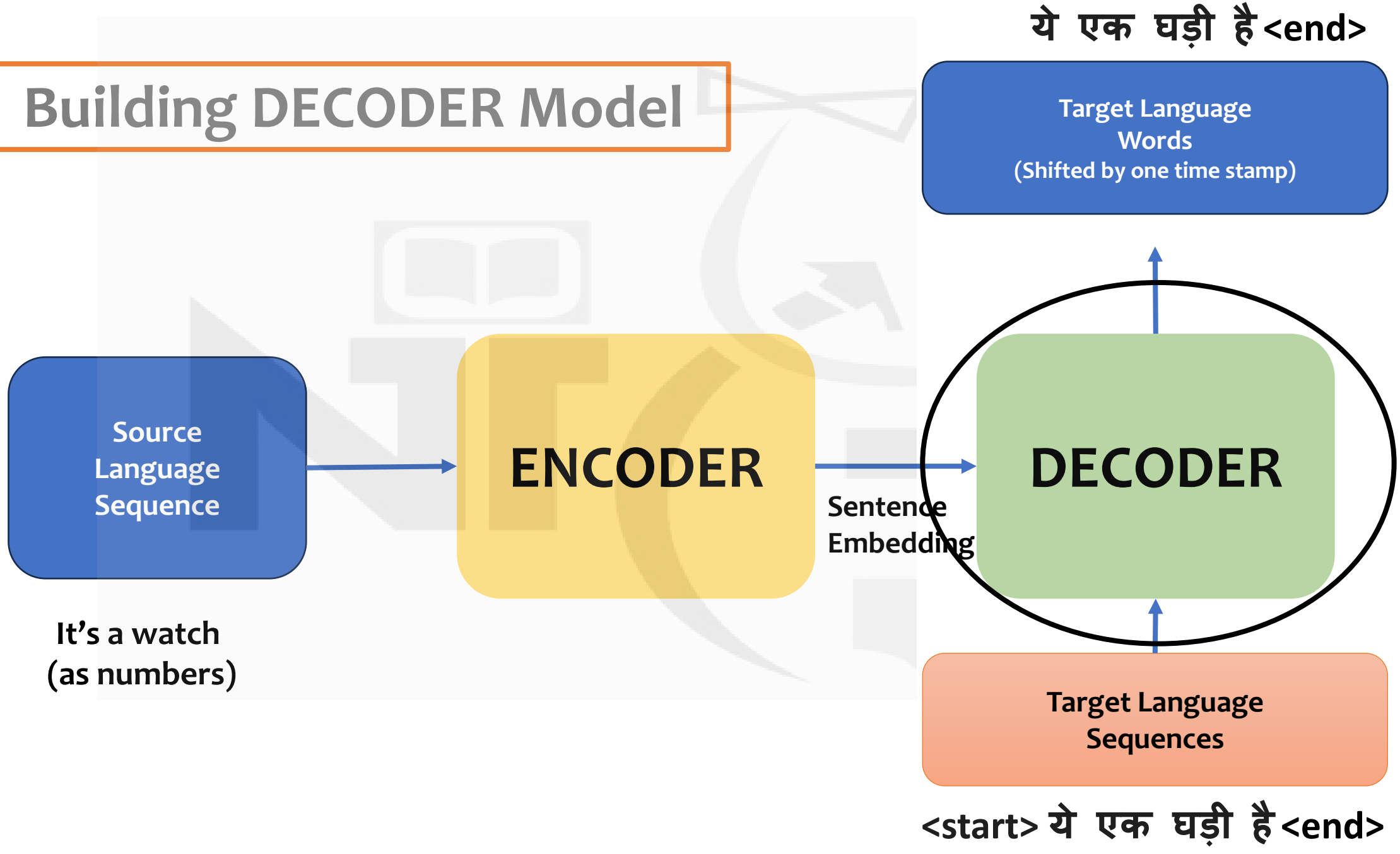


ENCODER

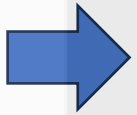
What is the value , Sentence Embedding

Building DECODER

Building DECODER Model



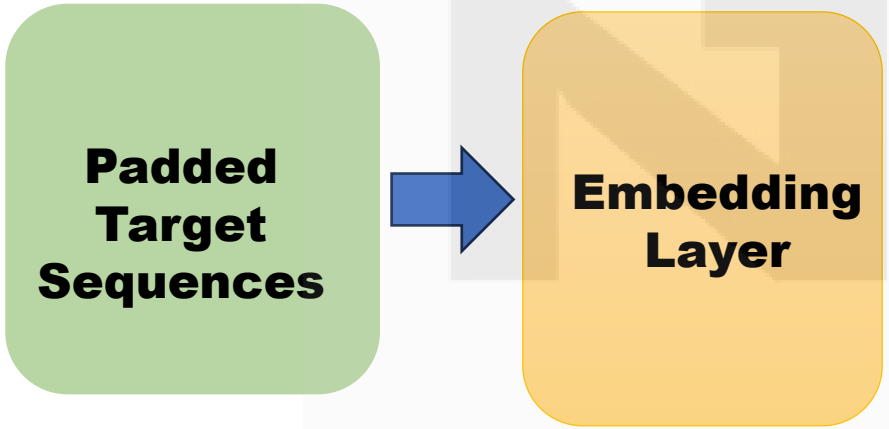
Padded Target Sequences



<start> ये एक घड़ी है <end>

What should be the first layer in DECODER???

DECODER

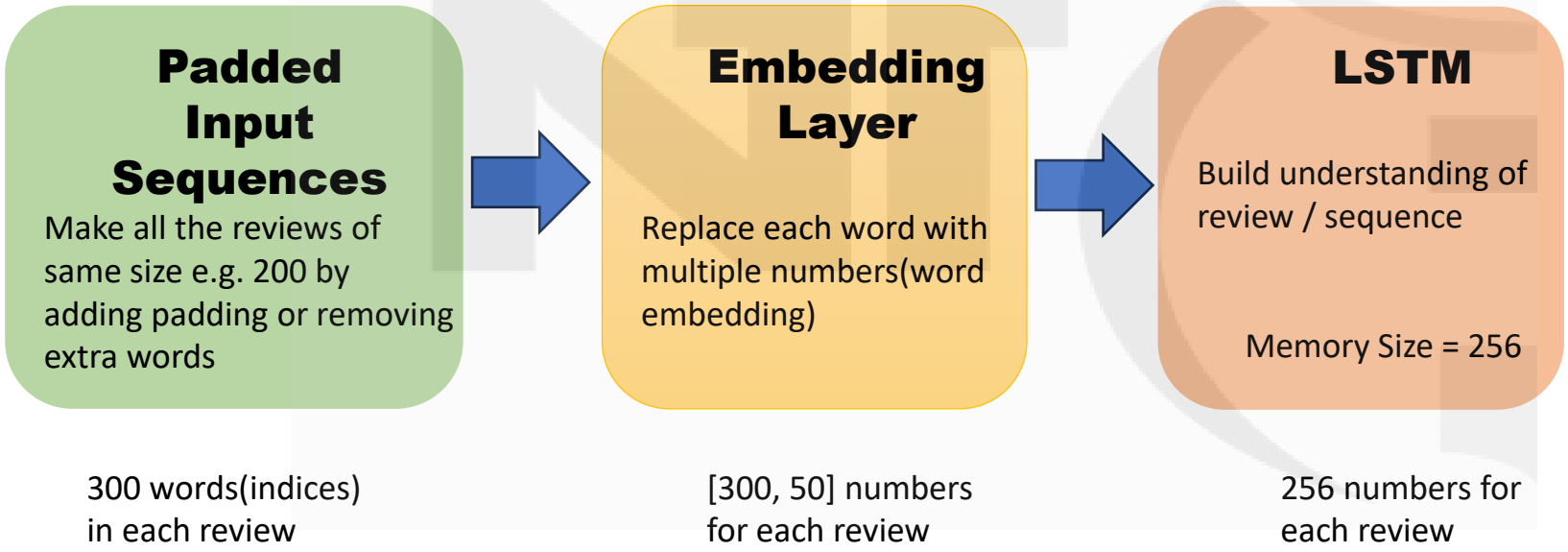


Similar to Encoder, Embedding layer to generate word2Vec embedding

DECODER

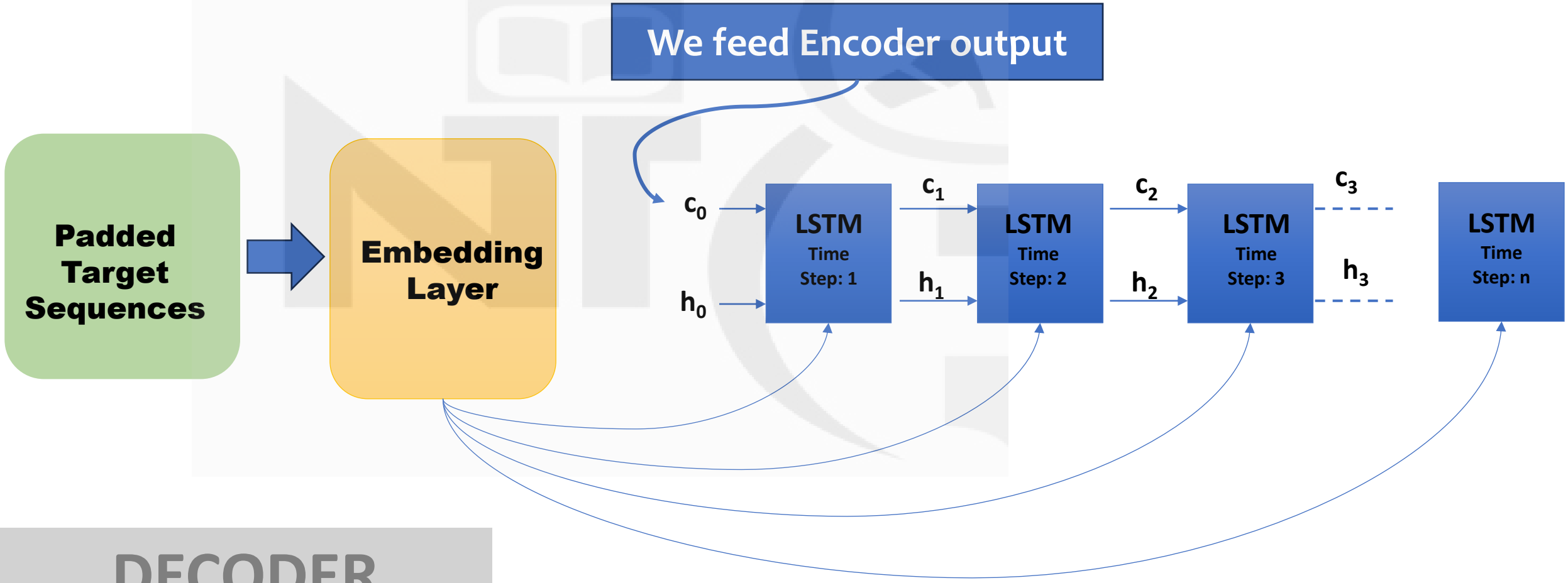
Next?

- We calculate the loss

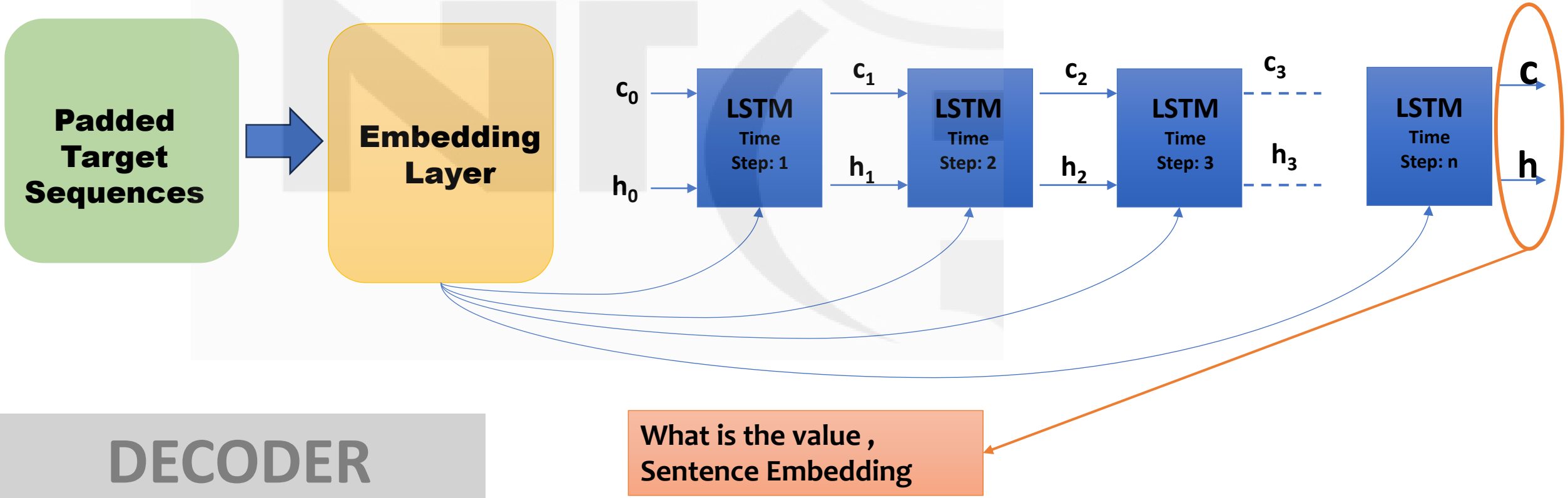


We will use LSTM to understand Target sequence

DECODER

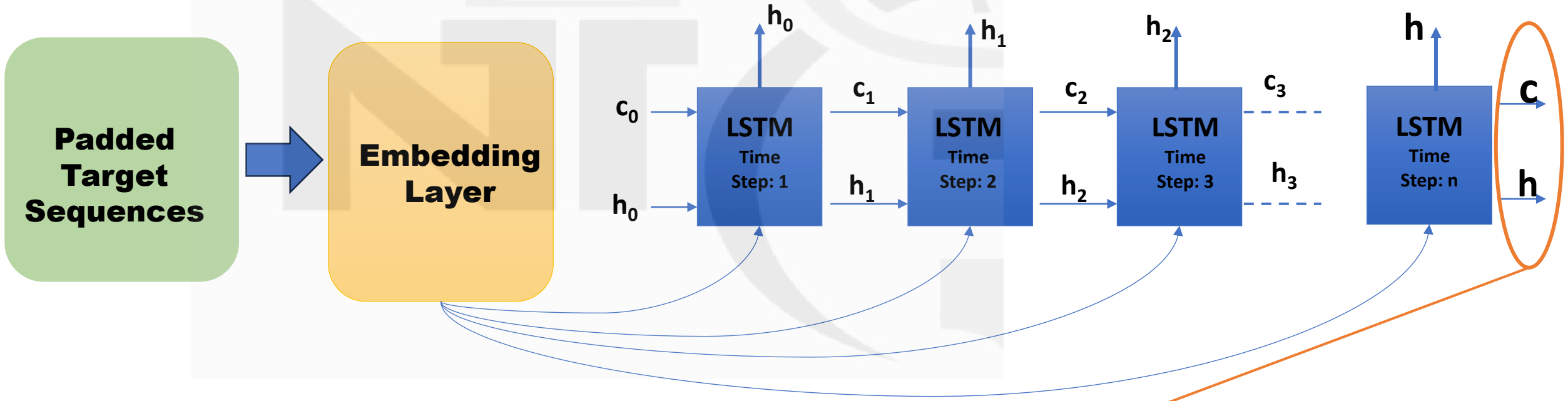

We feed Encoder output
Padded Target Sequences
Embedding Layer
c_0
h_0
LSTM Time Step: 1
c_1
h_1
LSTM Time Step: 2
c_2
h_2
LSTM Time Step: 3
c_3
h_3
LSTM Time Step: n

DECODER


Padded Target Sequences
Embedding Layer
c_0
h_0
LSTM Time Step: 1
c_1
h_1
LSTM Time Step: 2
c_2
h_2
LSTM Time Step: 3
c_3
h_3
LSTM Time Step: n
c
h
What is the value , Sentence Embedding

DECODER

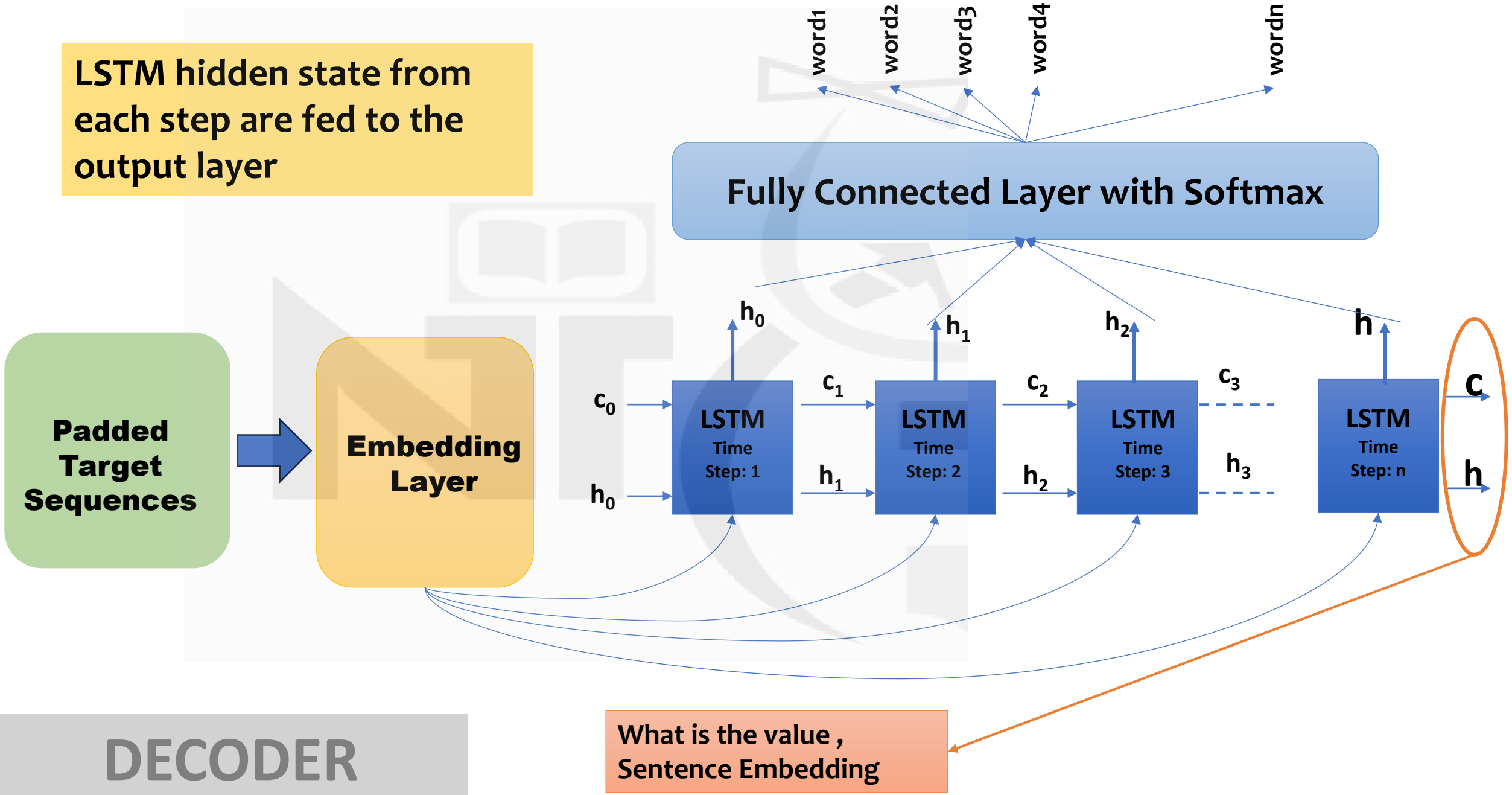
We take LSTM output(hidden state) at each time stamp



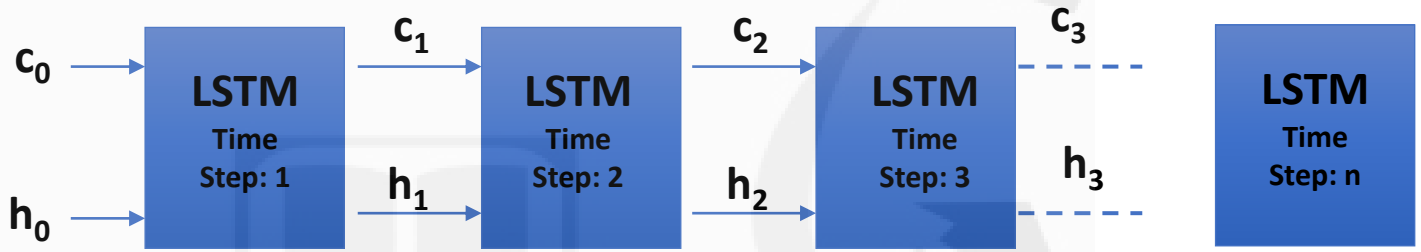
DECODER

What is the value , Sentence Embedding

LSTM hidden state from each step are fed to the output layer

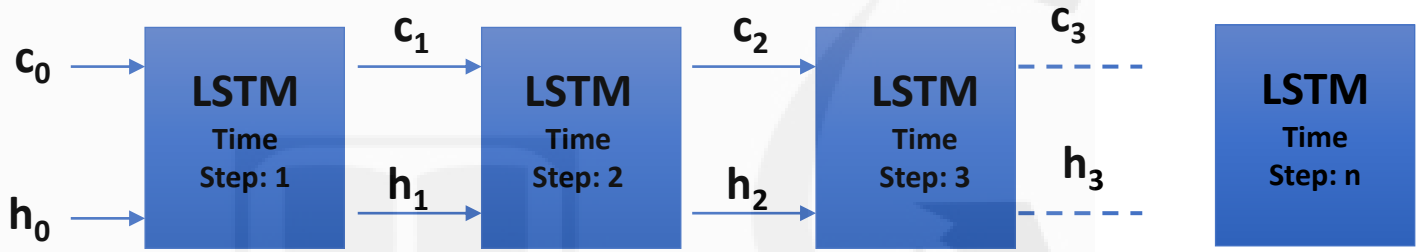


Working with LSTM layer in Keras



output = LSTM(256) <input>

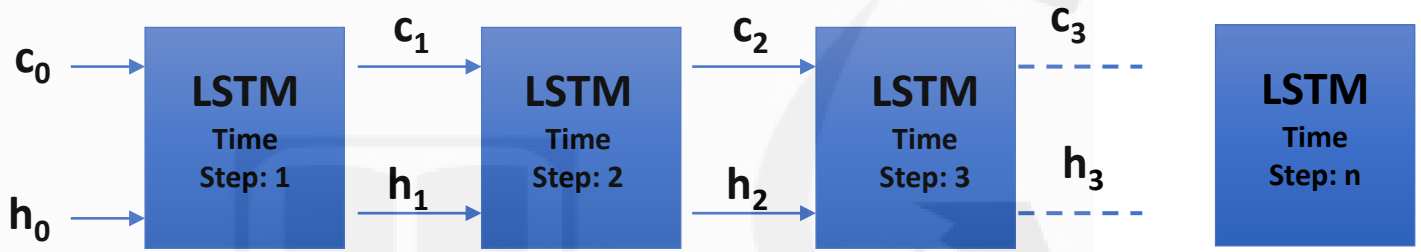
What will the output of LSTM layer?



output = LSTM(256) <input>

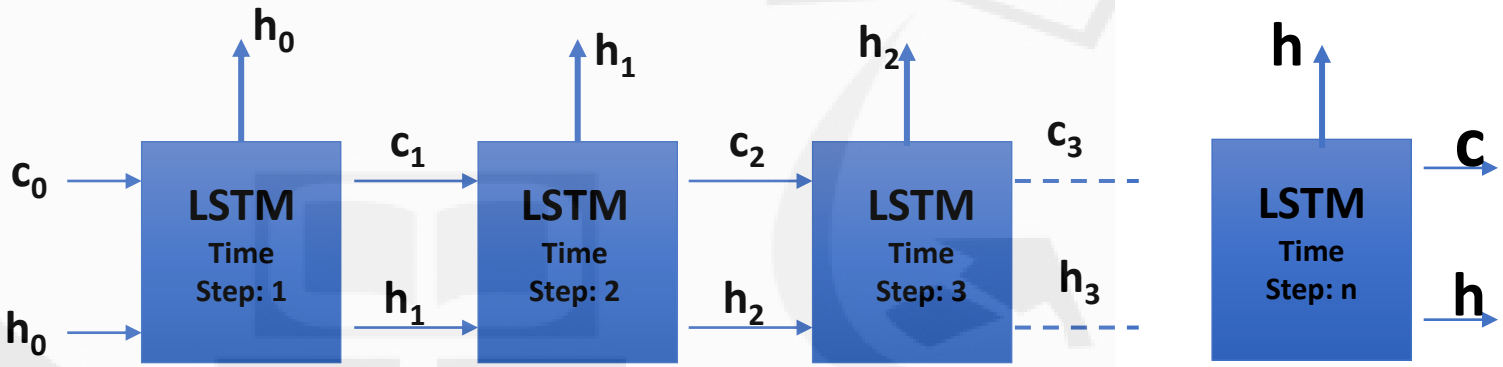
By default, output is hidden_state 'h' of the last step.

How do I get both ‘h’ and ‘c’ of the last step?

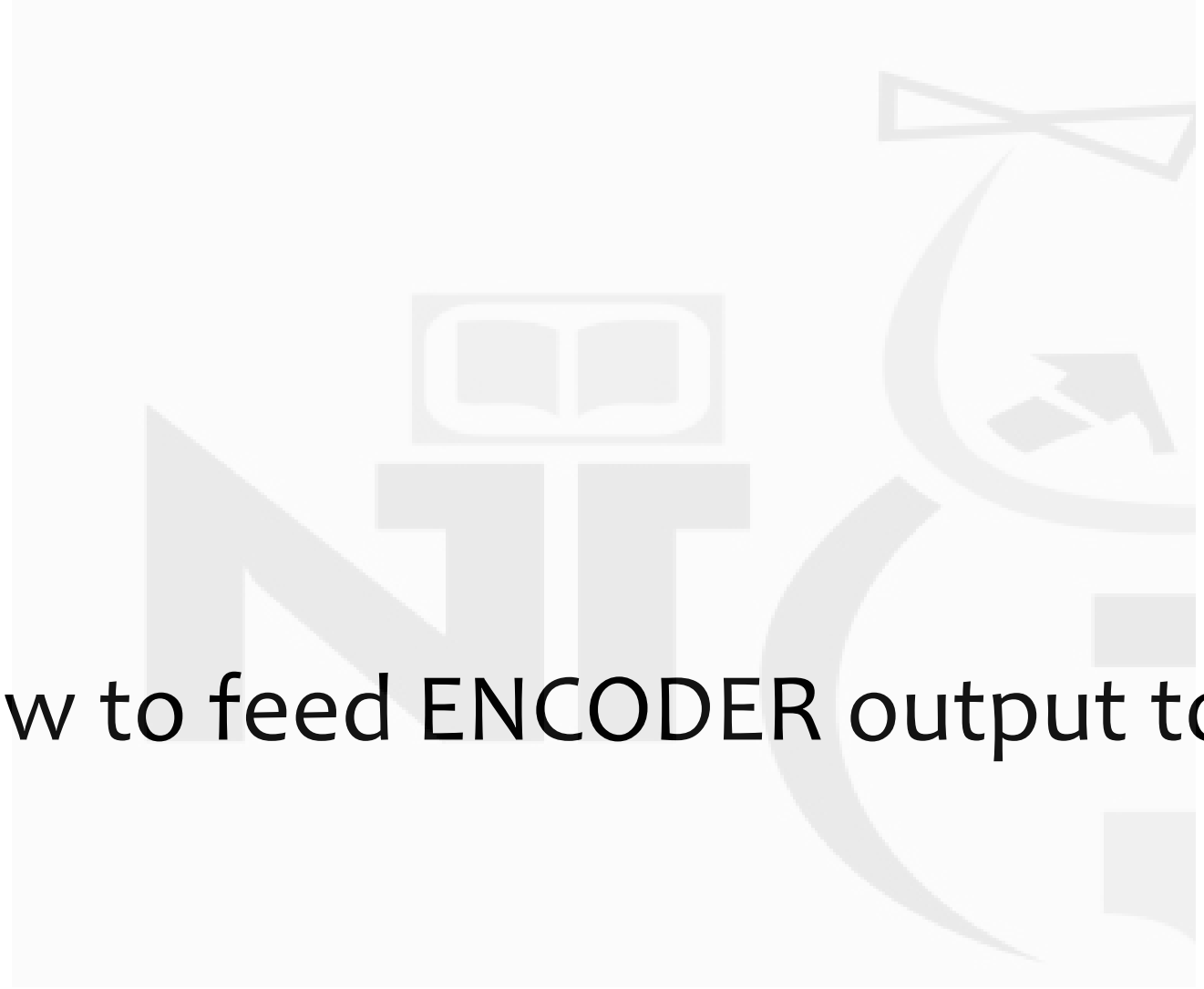



```
output, h, c = LSTM(256, return_state=True) <input>
```

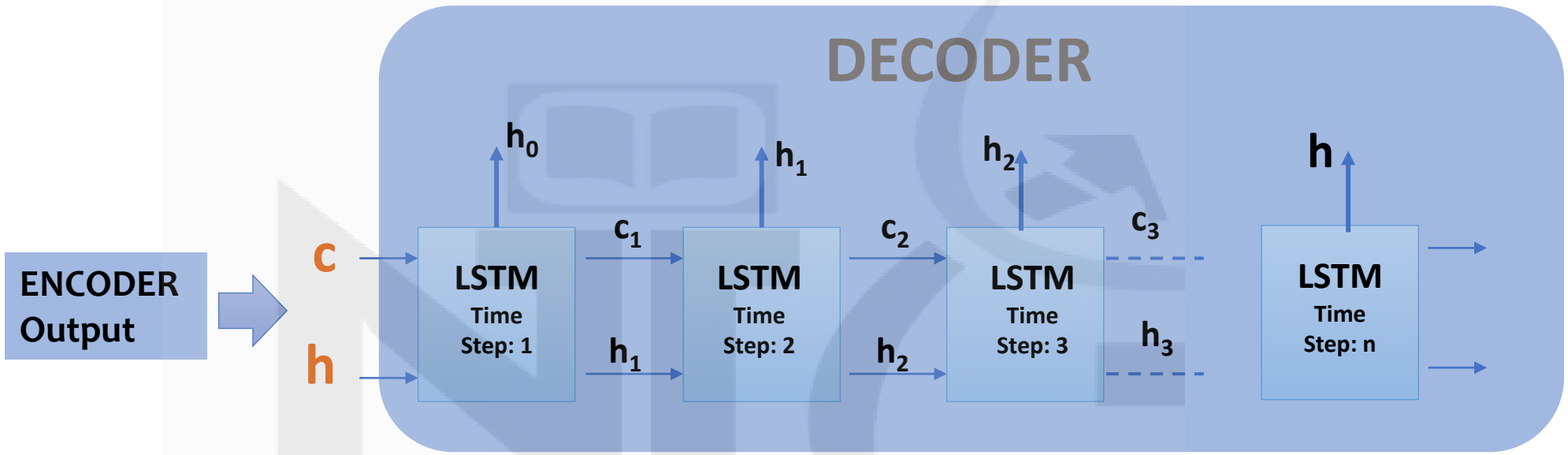
How do I get hidden state ‘h’ of all steps?



Outputs = LSTM(256, return_sequences=True) <input>

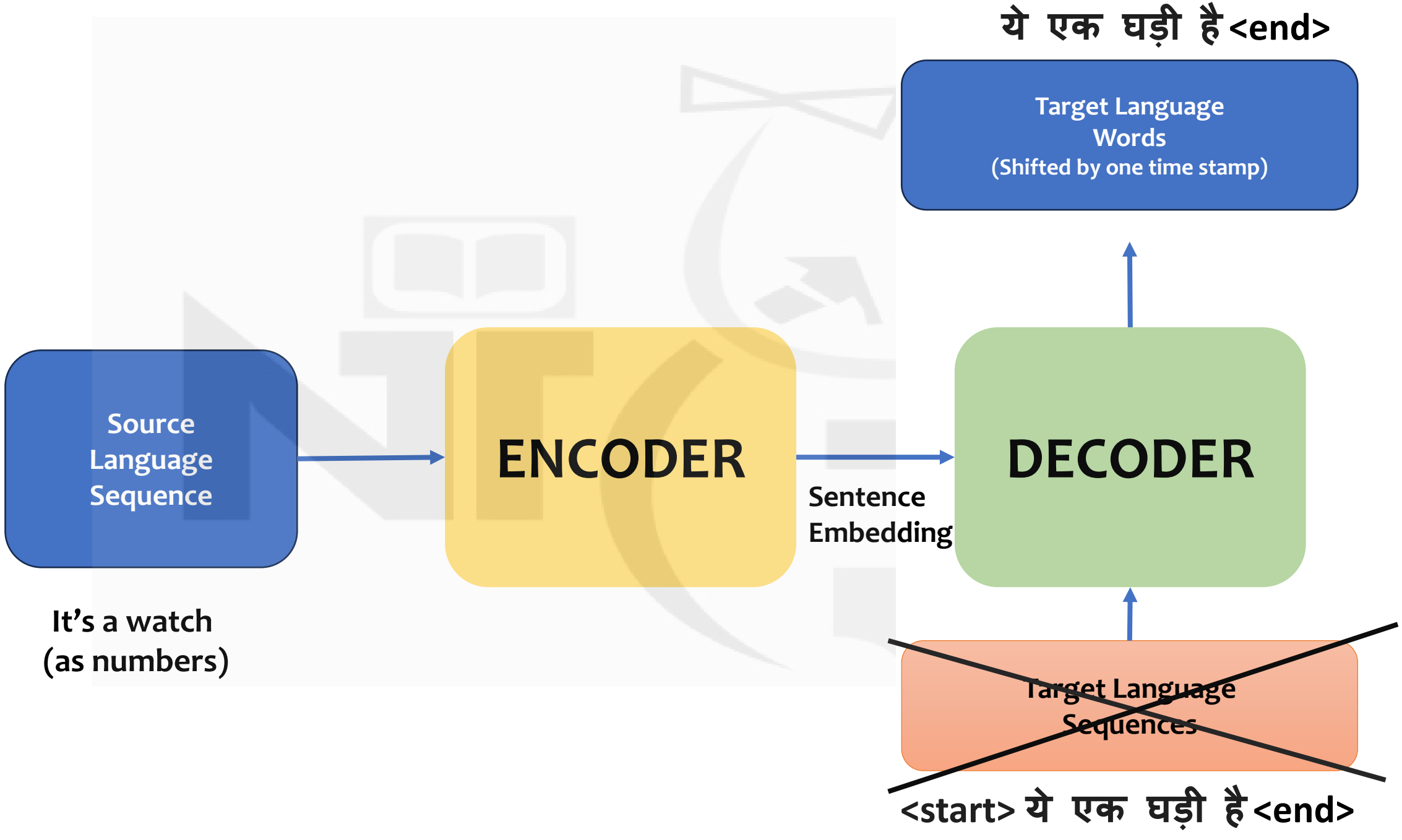


How to feed ENCODER output to DECODER?

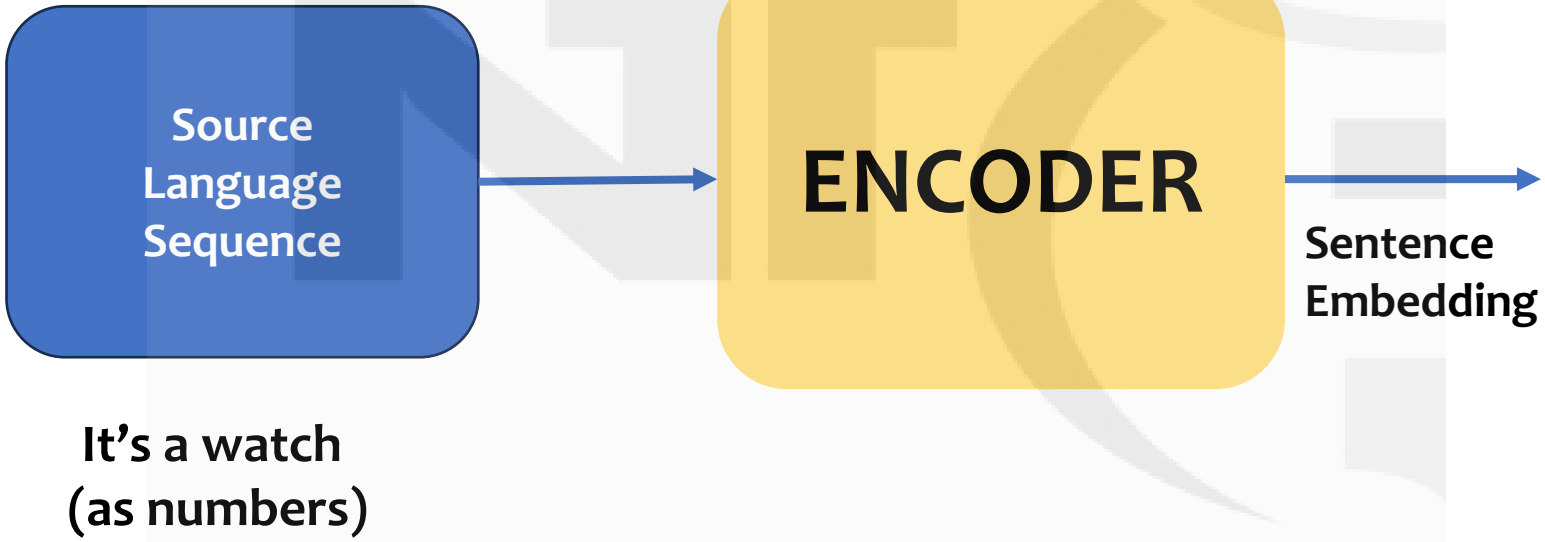



```
model_layer = LSTM(256, initial_state=[h,c]) <input>
```

Building the model for predictor


ये एक घड़ी है <end>
Target Language
Words
(Shifted by one time stamp)
Source
Language
Sequence
ENCODER
Sentence
Embedding
DECODER
It's a watch
(as numbers)
Target Language
Sequences
<start> ये एक घड़ी है <end>

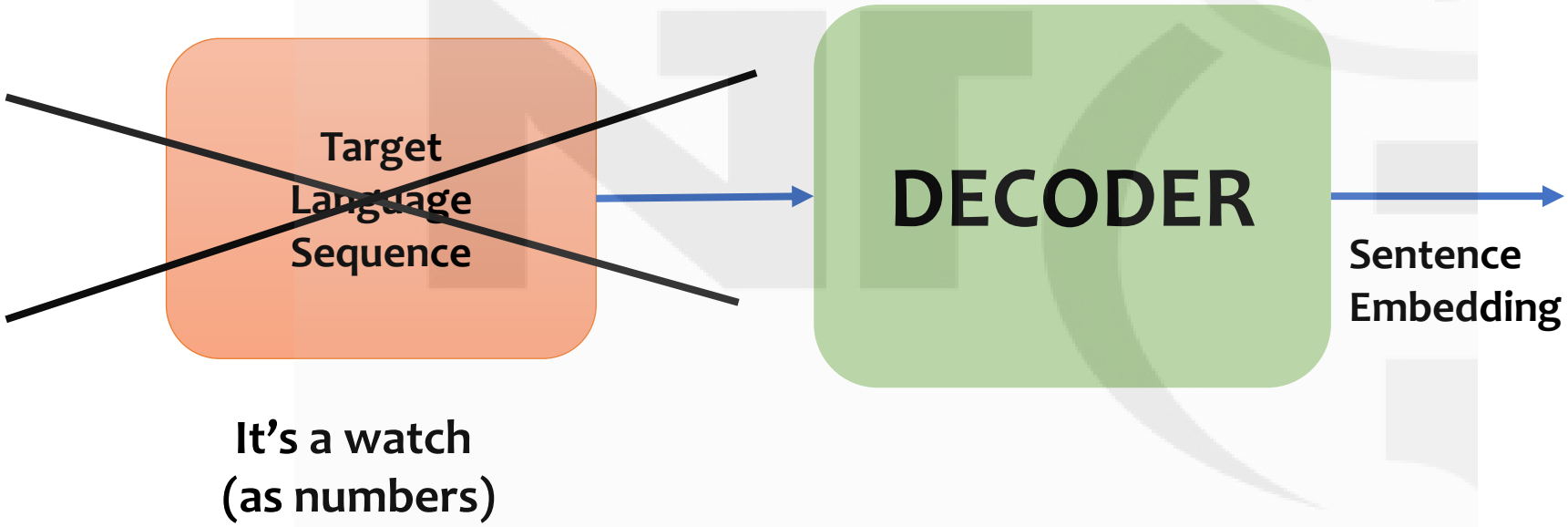
Encoder Model



It's a watch
(as numbers)

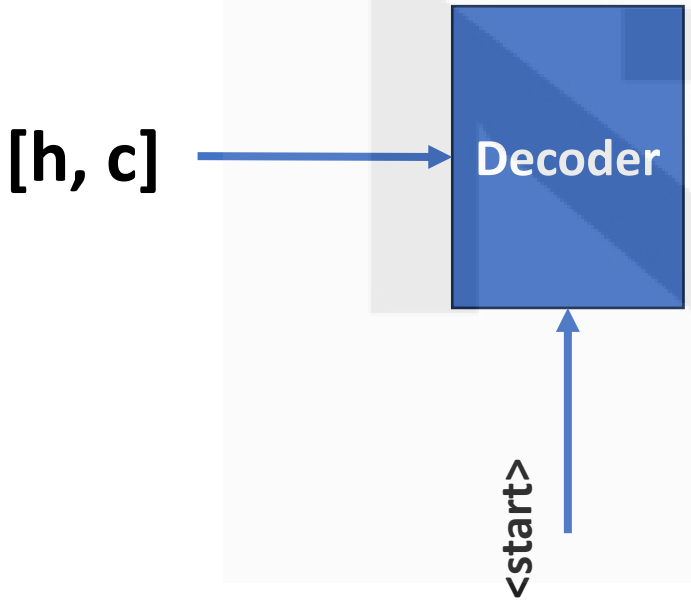
Can we run encoder as it is??

Decoder Model

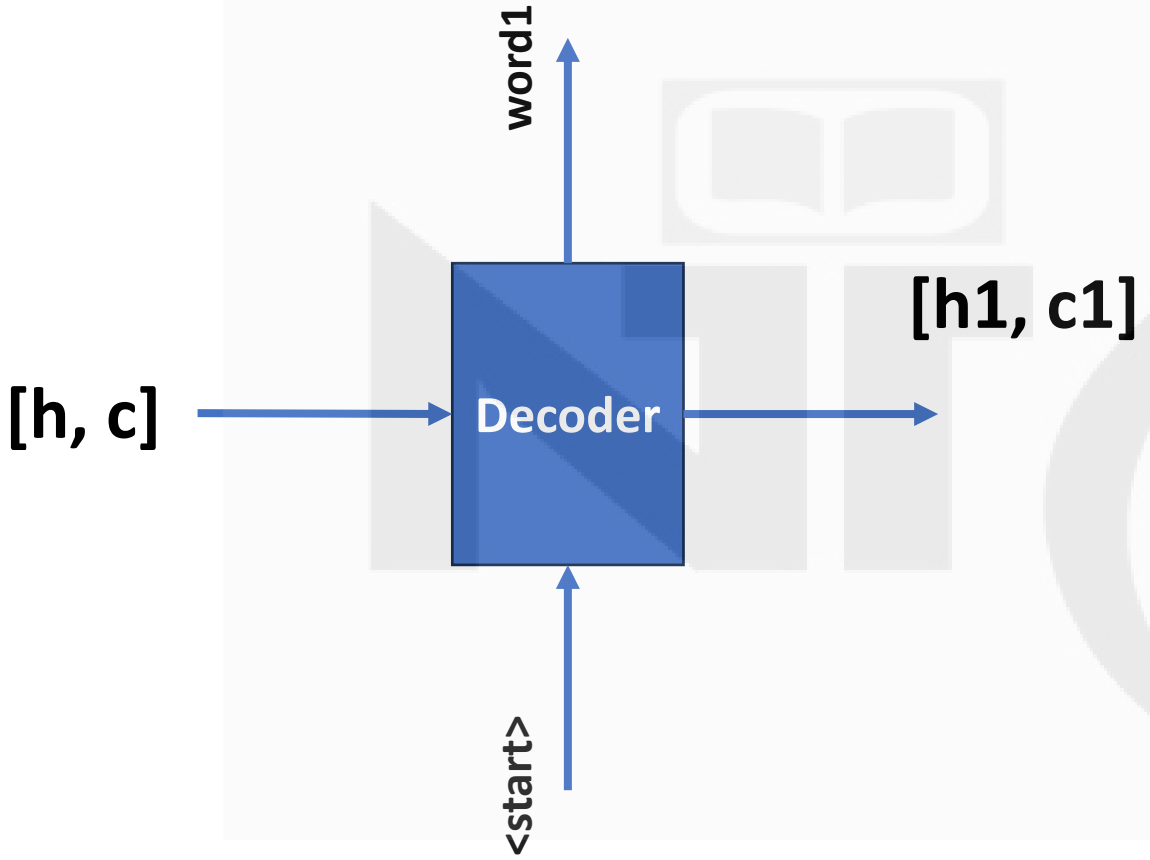


It's a watch
(as numbers)

What is the input to decoder???



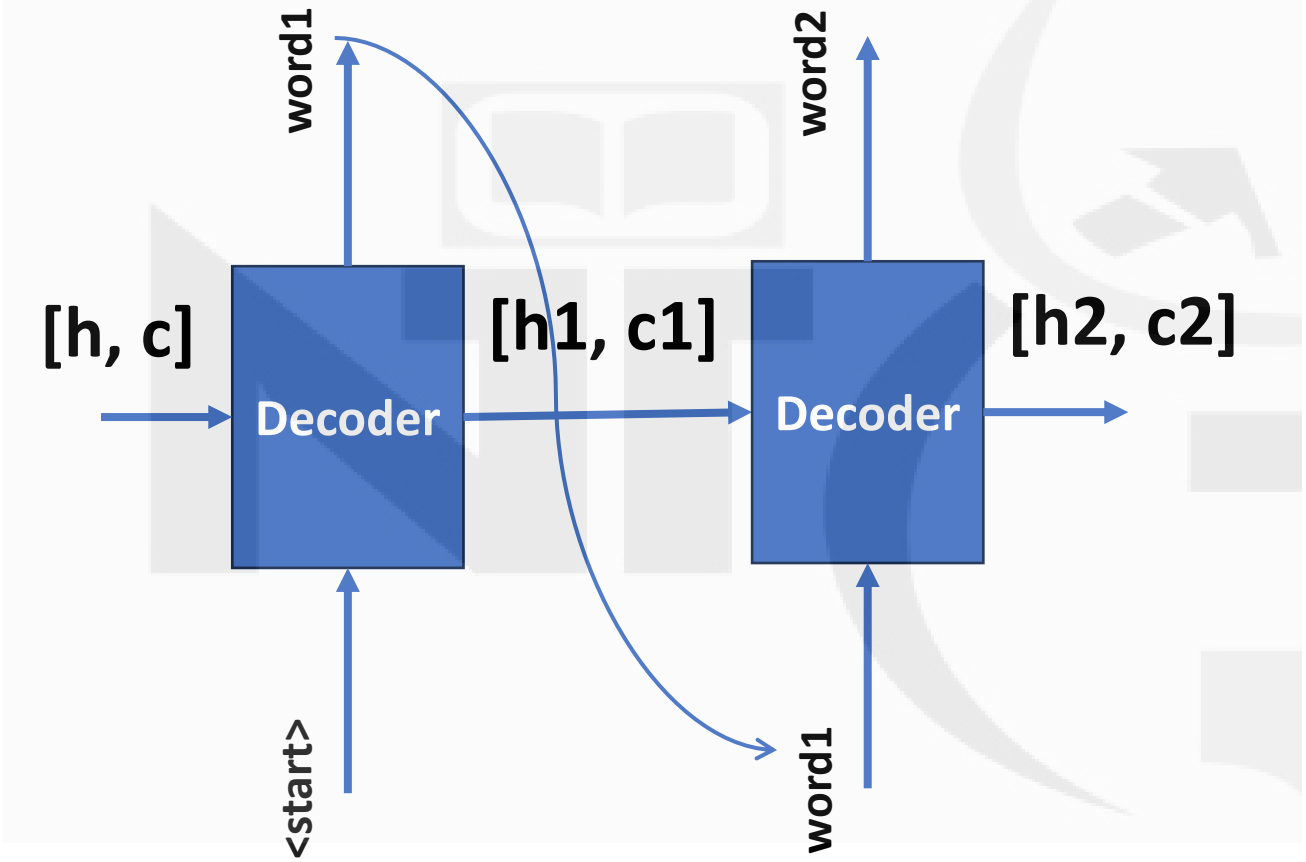
We know the first word always?



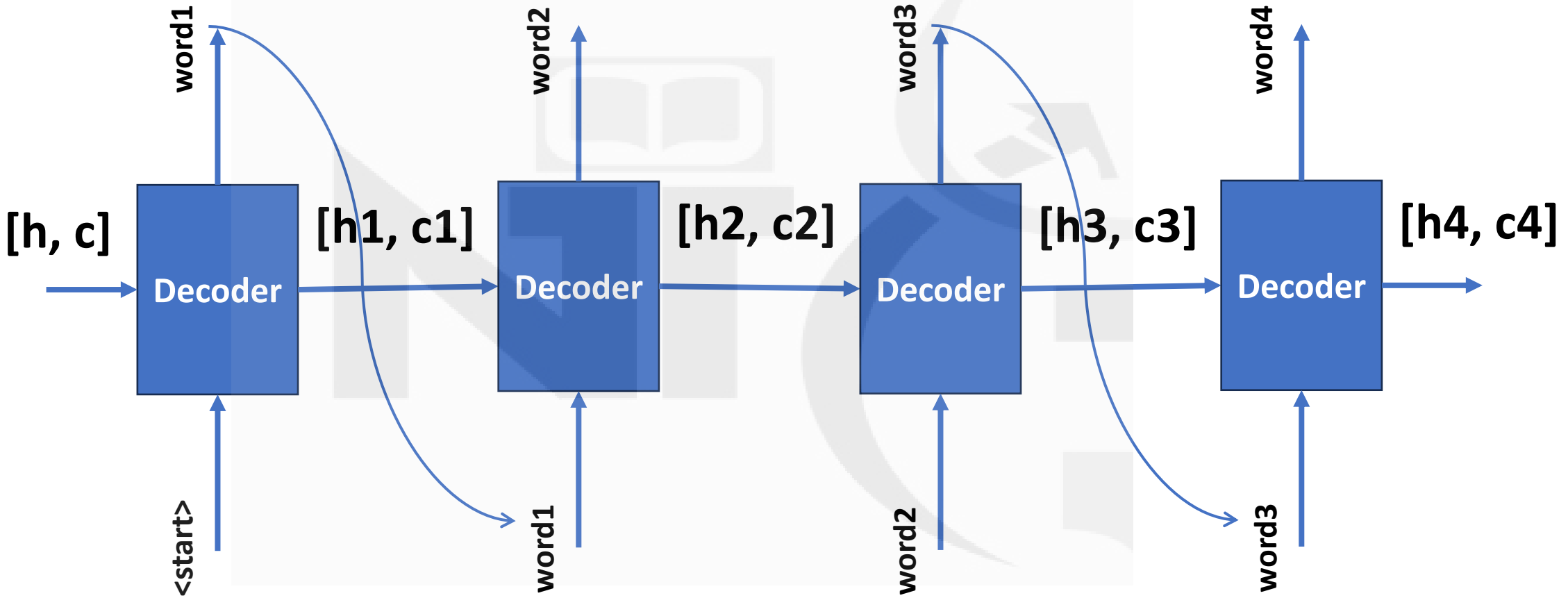
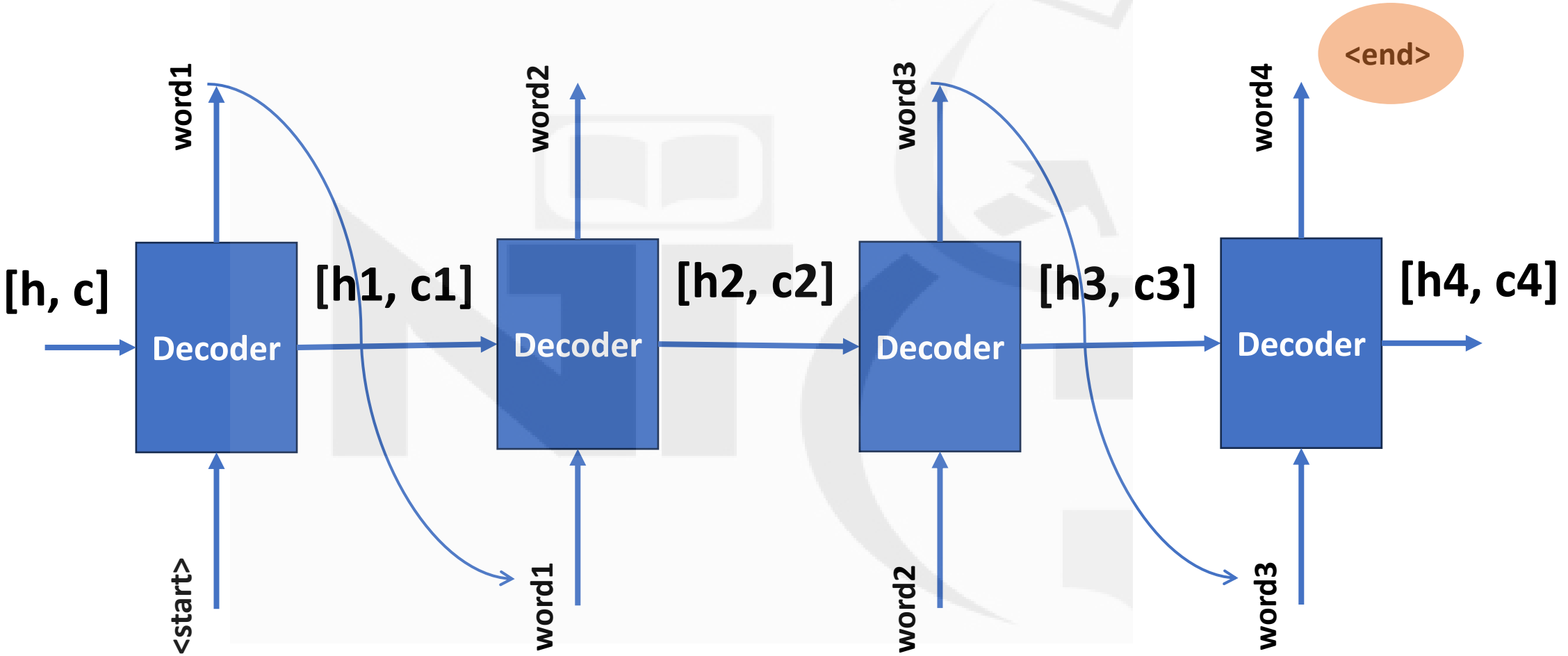
3 outputs:

- Next word
- New h state
- New c state

Run decoder again



And again


word1
word2
word3
word4
[h, c]
Decoder
[h1, c1]
Decoder
[h2, c2]
Decoder
[h3, c3]
Decoder
[h4, c4]
<start>
word1
word2
word3

How many runs of decoders?