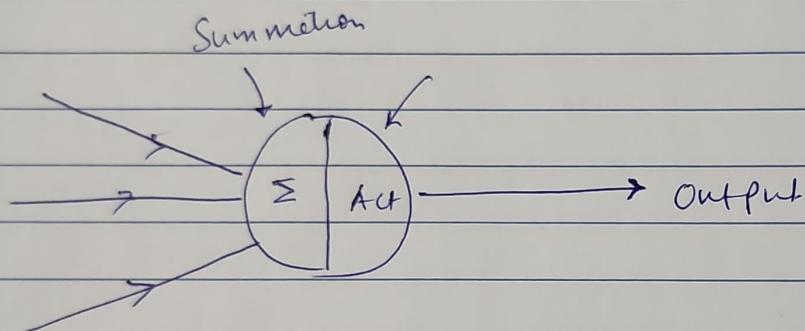


Introduction to LLM Deep Learning

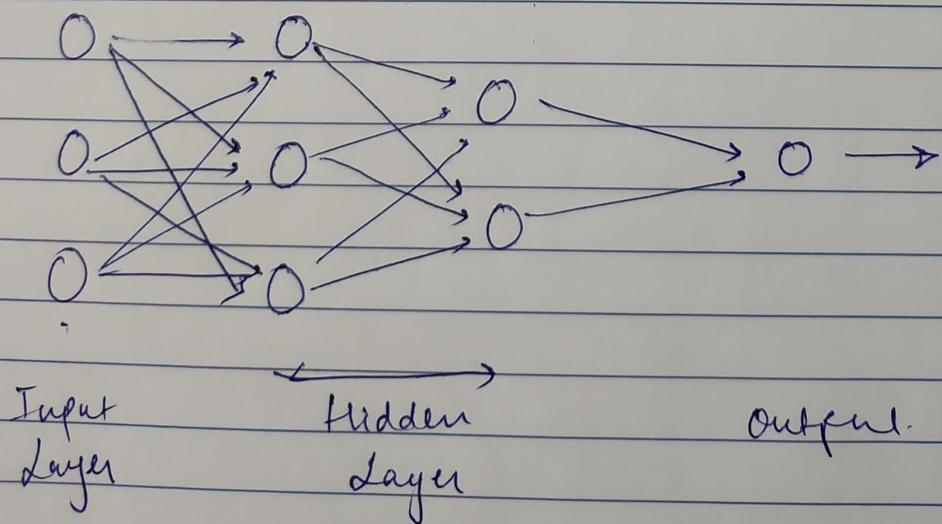
- 1. Artificial Neuron Network ANN
- 2. Convolutional Neuron Network CNN
- 3. RNN
- 4. GAN
- 5. Reinforcement Learning RF

Architecture of Perceptron



Unit of AI \rightarrow perceptron

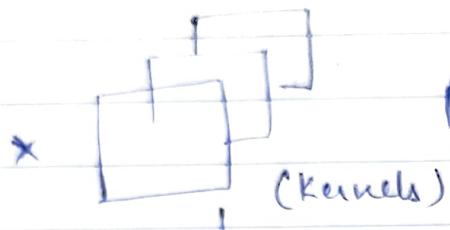
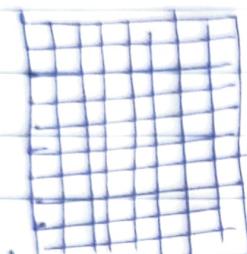
Multi Layer Perceptron "Artificial Neuron Network"



Convolutional Neural Network

Image Related data.

Image is collection of Pixel



↓
pixel is Number

Range of Pixel {0 - 255}

"Convolution"

[RELU] → Activation function

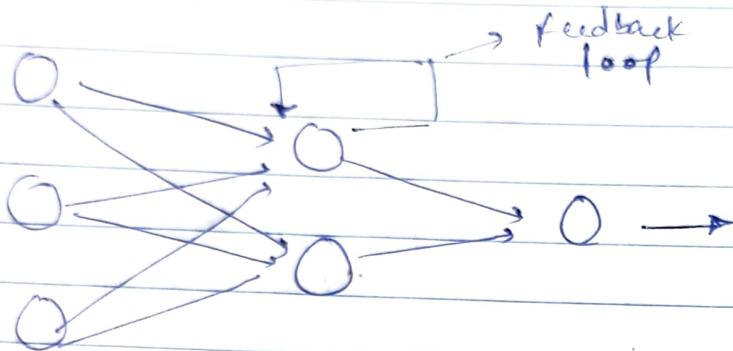
• Pooling] → feature Selection

Flattening → Neural Network

Convert the
entire date in 1 D ARRAY.

RELU \Rightarrow Rectified linear Unit

RNN → Recurrent Neural Network



Input Layer Hidden Layer Output Layer

(Artificial Neuron Network)

In Recurrent Neural Network → you get a Feedback loop.

Process the data based on Timestamp

LSTM → Long Short - Term Memory

* Inspired from RNN

GRU →

RNN, LSTM, GRU
Seq2Seq Mapping

1. one to many Technique
2. Many to one Technique
- (3.) Many to Many Technique

- ↓
1. Synchronize [Same Length]
 2. Asynchronize [Different Length]

Encoder - Decoder (Google
2014. Relecen)
(There were limitation in
Encoder - Decoder)

→ Attention Mechanism

TRANSFORMER (Derived from Attention
Mechanism)

Attention is
All you need

↓
Self - Attention

TRANSFER Learning , fine tuning



L·L·M (Large Language Model)

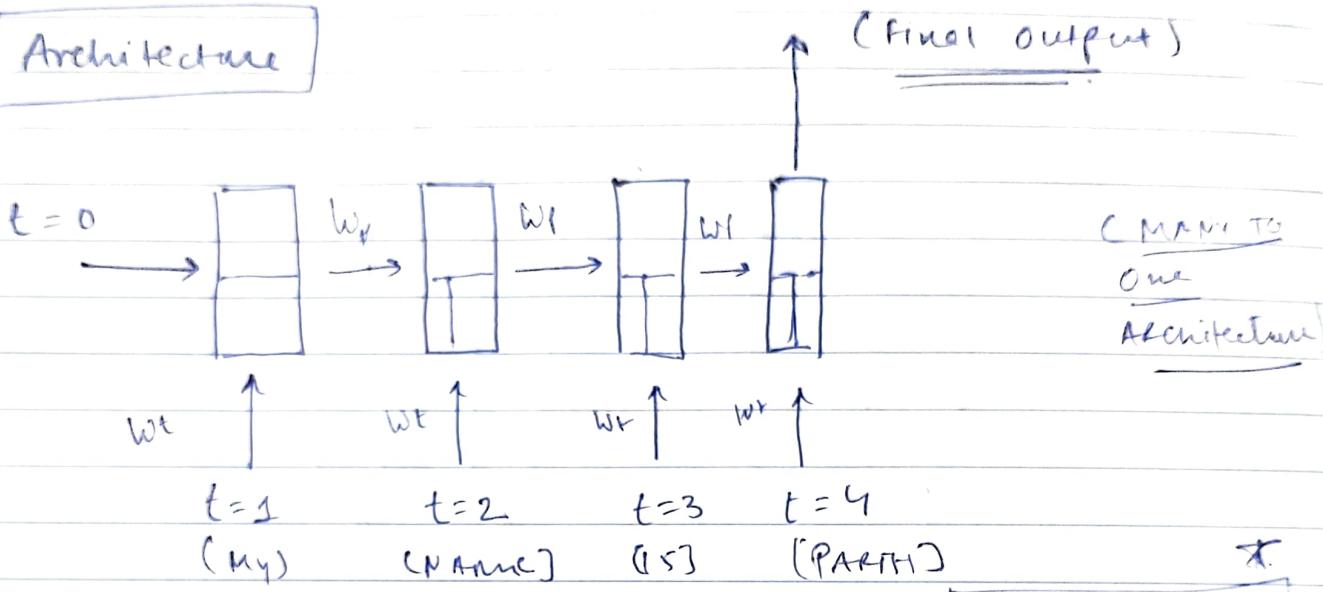


Chat GPT

Doc = My Name is PARTH
 This info is based on previous data
 Text data and also Sequence Data*

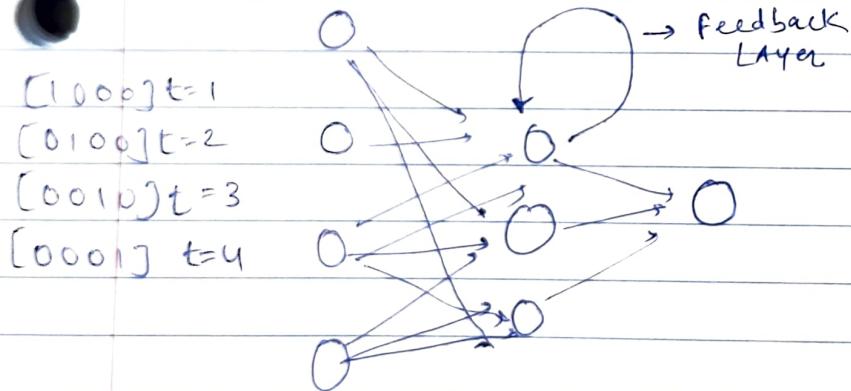
RNN

Architecture



CMANI TO ONE ARCHITECTURE

W_t = Weight
 W_p = Previous Weight



MY NAME IS PARM

$\overline{t=1}$

$\overline{t=2}$

$\overline{t=3}$

$\overline{t=4}$

One-hot Encoding

[1 000]

[0 1 0 0]

[0 0 1 0]

[0 0 0 1]

VVIP

RNN Problem

* Vanishing Gradient [Old was not getting updated]

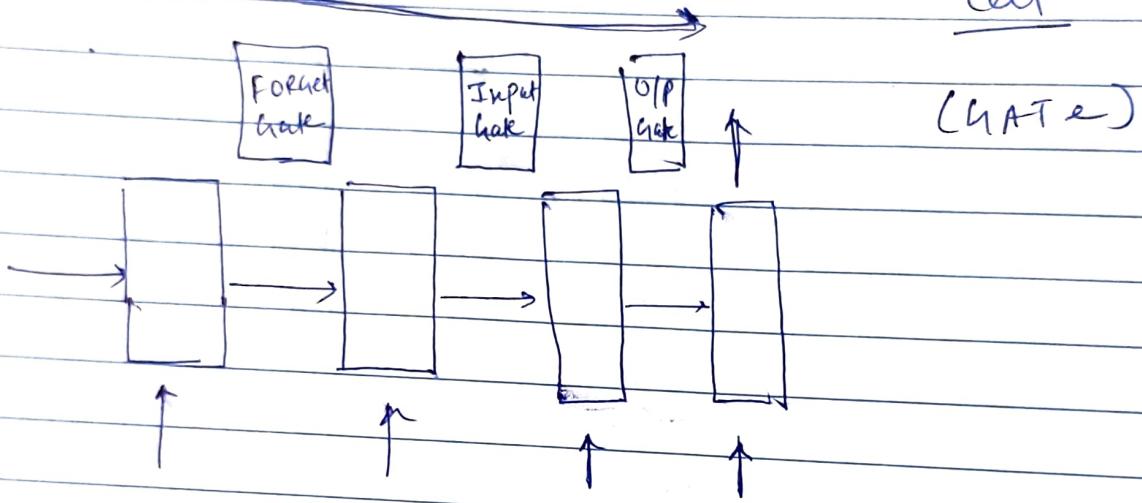
RNN → WAS NOT Able to process long sequence

- NOT able to process long sentence
- Not able to sustain **CONTEXT***

(2)

LSTM → long short term memory

Along with hidden state, concept of Memory Cell



* Memory cell is used for storing or context

Whatever is not required will be removed
in Forget GATE

Gate is Neural Network

GRU Architecture

In GRU there are 2 gate



Reset Gate Update Gate

There is no concept of Memory cell

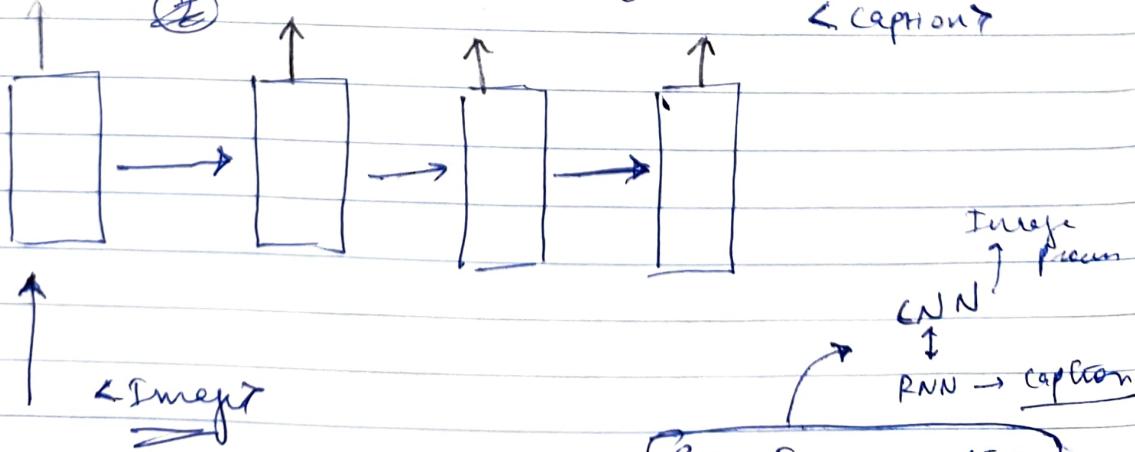
Advantage of this Architecture

→ LSTM too much computational

GRU is implemented on based of LSTM, but architecture is minimized.

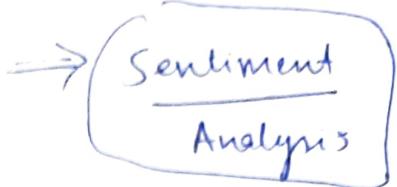
Seq2Seq 3 challenges

① One to Many ↑ = output
②



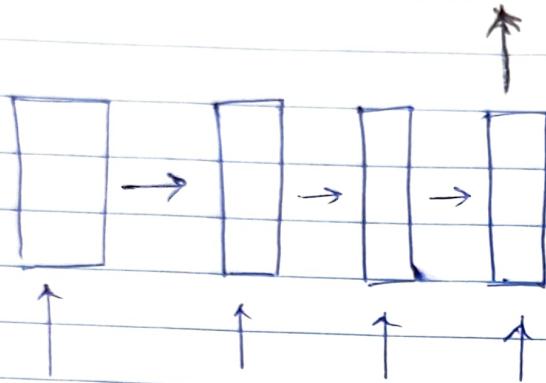
All input is inputted in single shot

{ Many Input → One output }



(2) Many to one

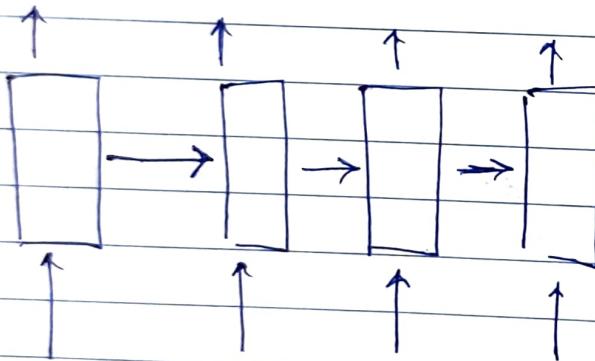
* Sentence processing



[RNN, LSTM,
GRU]

(3) Many to MANY (Same length Architecture) → Synchronize

Many input → MANY output



Ex → Machine TRANSLATION

I AM VERY HAPPY

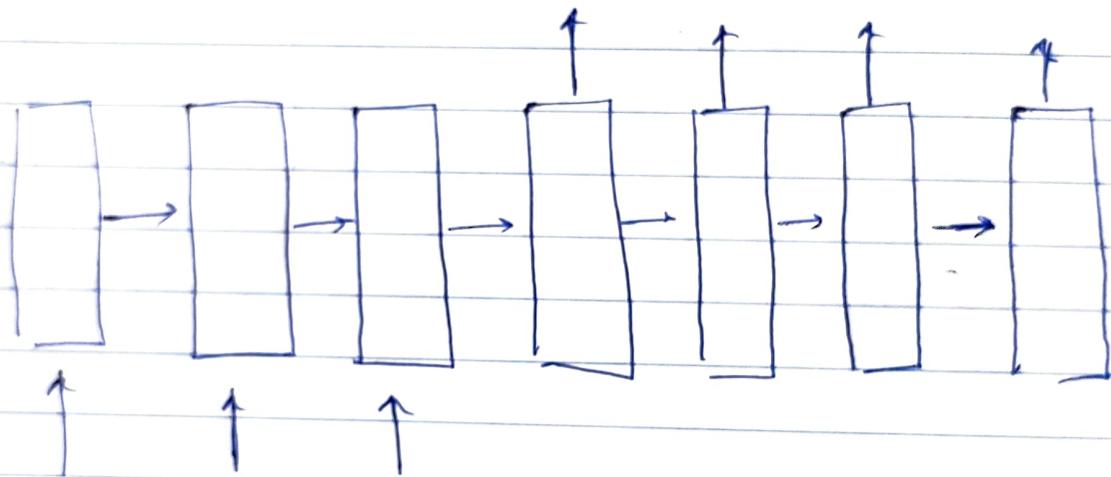
P.O.S Tagging →

NLP

ML (Sync TAN) → Deep learning
(Async TAN)

* Encoder-Decoder or Seq2Seq Architecture but with different context

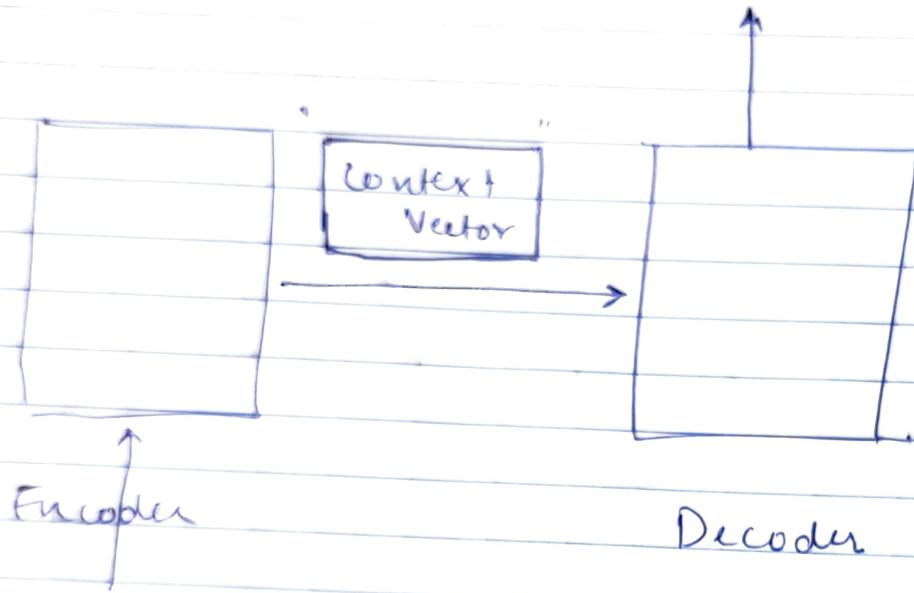
Asynchronous Architecture



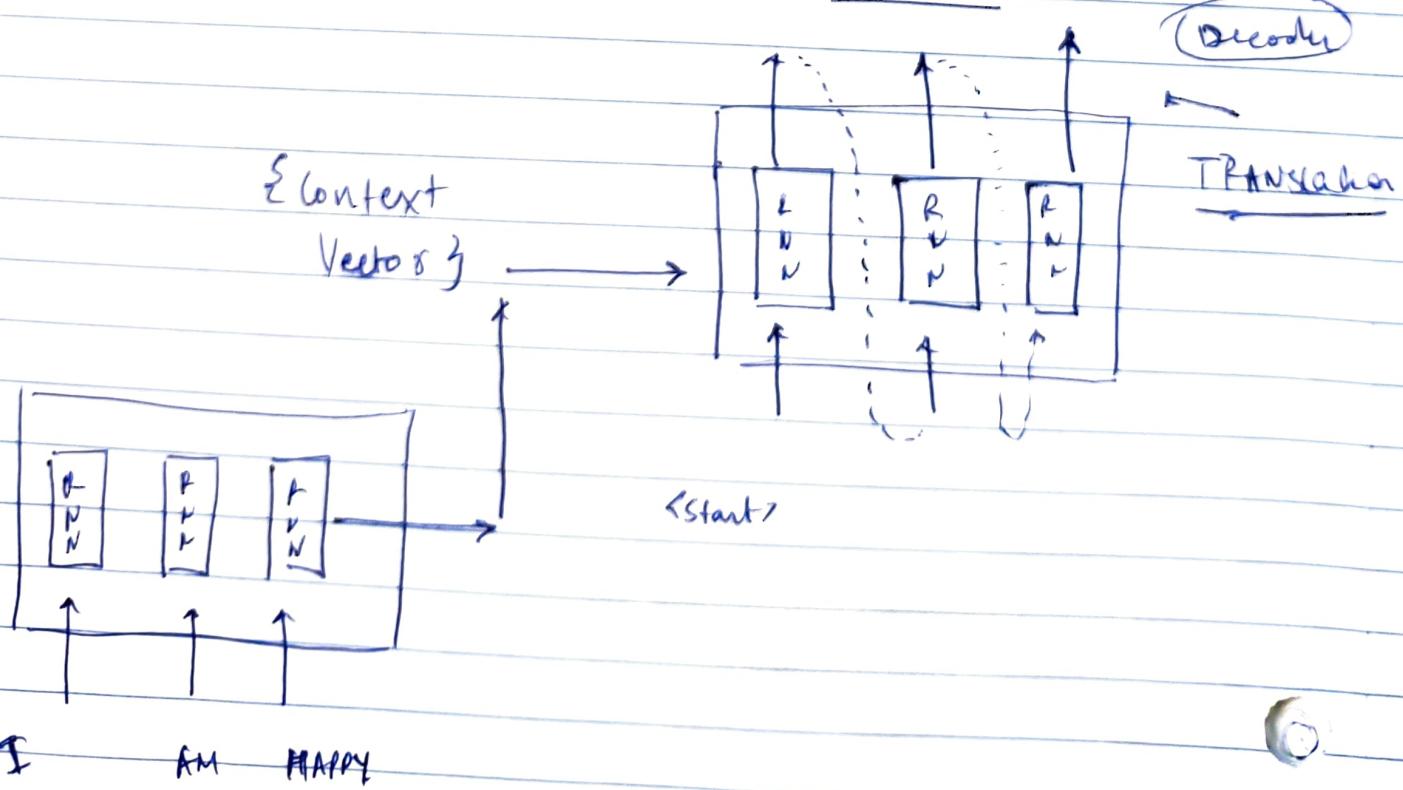
* This kind of problem can be solved by
Encoder-Decoder

Input \rightarrow Seq2Seq Learning
With Neural Network

Encoder Decoder



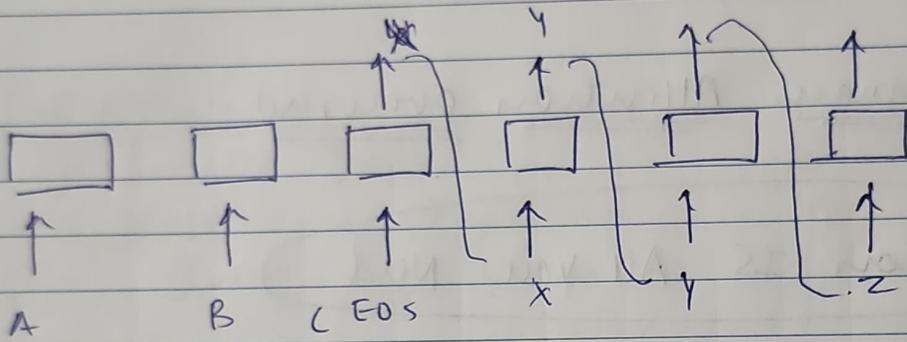
Encoder - Decoder is way of MAPPING



BLEU → Bilingual Evaluation Understudy

If sentence is more than 30 words, in that case Encoder-decoder will fail.

<EOS>



Bleu score started decreasing after 30 words

⇒ Then Google came up with New Idea

"Attention"

Research paper → Neural Machine Translation
for Attention

→ Attention Layer is Also Neural Network

Concept of Attention

Focus on Relative Word.

You → ڈاٹھ ڈھ
are → نیگ
Qual. → سی

* Bahdanau Attention overview

Attention IS All you Need

~~Research paper~~

Problems with Self Attention

(1) Computational complexity

(2) RNN | LSTM | GRU
(Encoder | Decoder with attention)

Self Attention

- (1) Parallel input
- (2) Positional Encoding
- (3) Query, Key, Value

(4) Multi headed Attention

(5) Normalization Layer

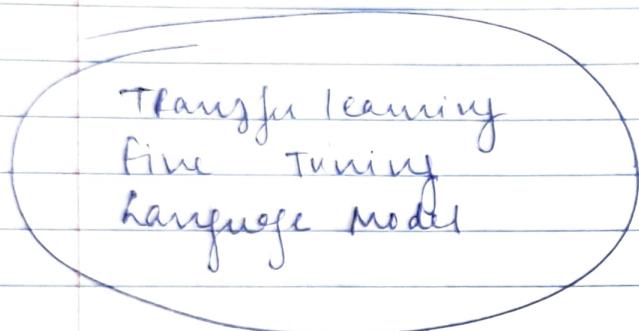
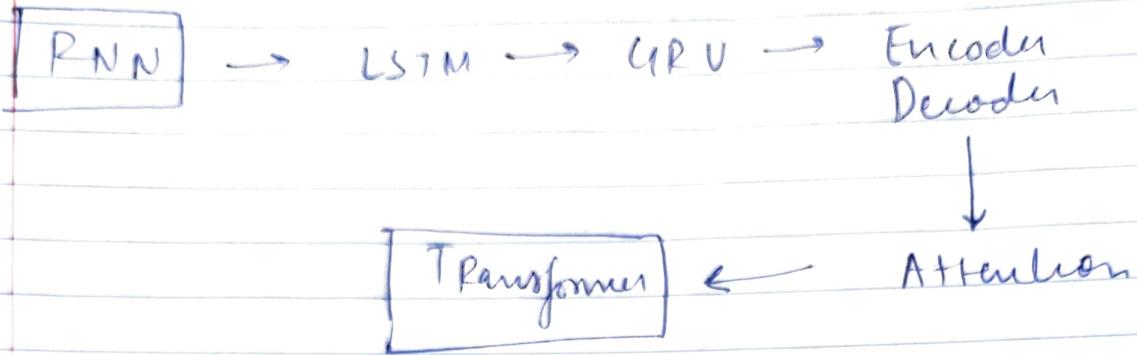
(6) Skip connection

Base
Architect
for LLM

(Seq2Seq)

Encoder - Decoder
Algorithm

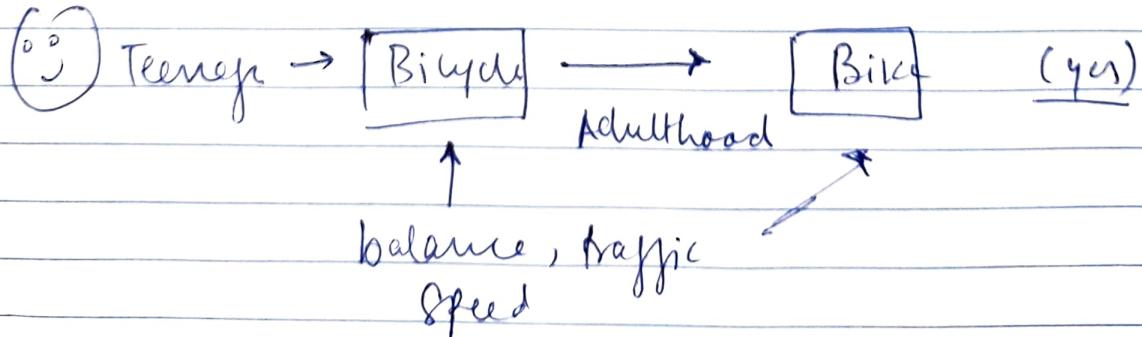
Using this architect it has Reduced
Computational complexity



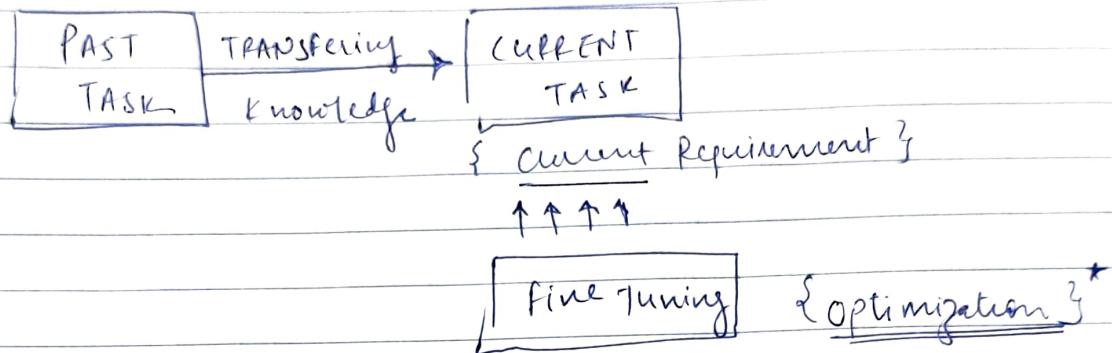
TRANSFER Learning | fine Tuning

Research → Universal Language Model fine
Paper Tuning for Text classification

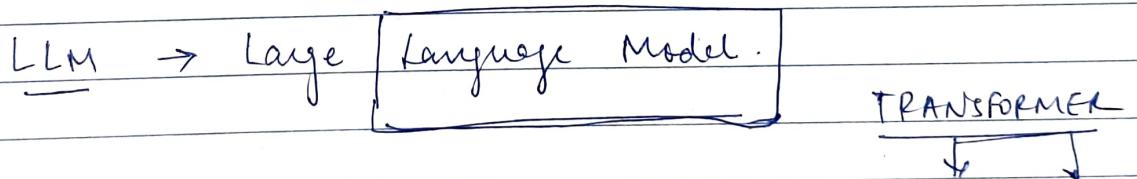
Transfer learning ⇒ Transferring the learning from
Previous task to Next
Task



Transfer learning $\Rightarrow \{$ learning from Past Experience $\}$



- No need to Train data from scratch
- But add data / learning from Requirement perspective is called fine tuning



Requirement for Transformer

(BERT) (GPT)

- ① Huge data Required to Train Transformer
- ② Good hardware
- ③ lots of Time

Google, facebook, MSFT \rightarrow Trained own model.

Language model

→ Huge amount of data.

TASK

- ① Text Summarization
- ② P-O-S / N-E-X-T
- ③ Sentiment Analysis
- ④ Text Generation

1 model for all the task?

Answer :- Yes → Ryan Research paper

"Universal Language Model
Fine Tuning for Text
Classification"

Unsupervised
ML

- ←
1. Language modelling
 2. Supervised fine-tuning

I AM PARTH. I AM DATA scientist.

—
predict
the next word

By doing this prediction, I taught
my model ⇒ ~~to~~ Language

BERT is trained on this
principle

Open AI - Chat GPT - 3rd layer

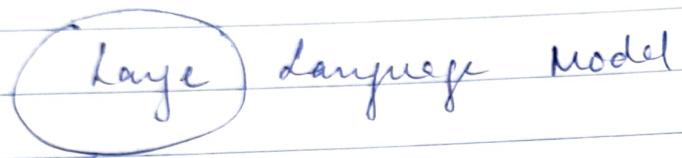
(RLHF) ~~BERT~~ → Reinforcement Learning
from human feedback

Language Understanding
↑ by Next Word

1. Unsupervised learning
2. Supervised fine tuning

[Language Model]
[Train your model on
specific Task]

3. RLHF



language data.

Chat API ⇒ 45 TB of DATA

* Research : BERT Pre Training of Deep
Paper Bi-directional Transformer for
language Understanding

