NLP

1. RNN
2. LSTM
3. GRU
4. encoder & Decoder
5. Attention
6. transformer
7. GPT & BERT (Language modeling) → ULMFIT

Embedding
text cleaning & text preprocessing

(gpt) openai → ↓↓↓ hugging APIs

Projects
1. Bert
2. Google Pegus

BERT, GPT, ULMFIT

Projects

Transformer
1. Encoder
2. Decoder

→ Attention all you need

Progression of NLP in chronological order:

1. RNN
2. LSTM
3. GRU

→ { Encoder & Decoder with attention } → Transformer

Transformer → E & D ⇒ Self attention

If in the interview we are asked about how are we going to identify which transformer we would be requiring. Then tell them based on divisions mentioned alongside?

There are many LLM models. But the ones mentioned here are the milestones which even people are using in industries in Production

Transformer

Only encoder based transformer

Both Encoder and Decoder based transformer

Only Decoder based transformer

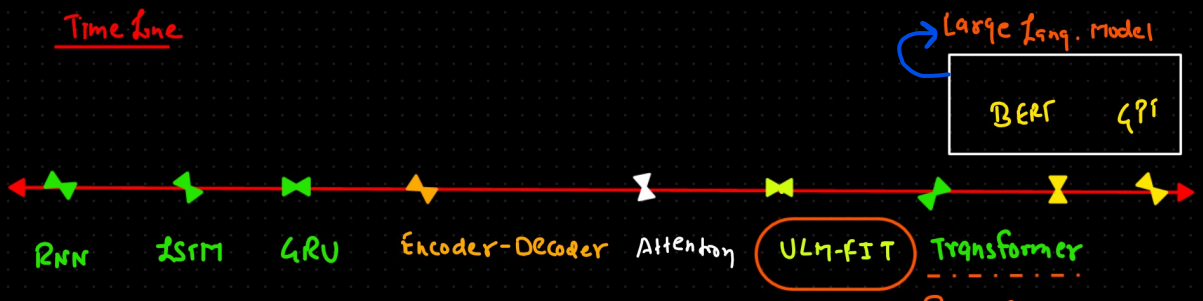
Encoder
1. BERT
2. RoBerta
3. XLM → XLM-R
4. ALBERT
5. DeBERTa

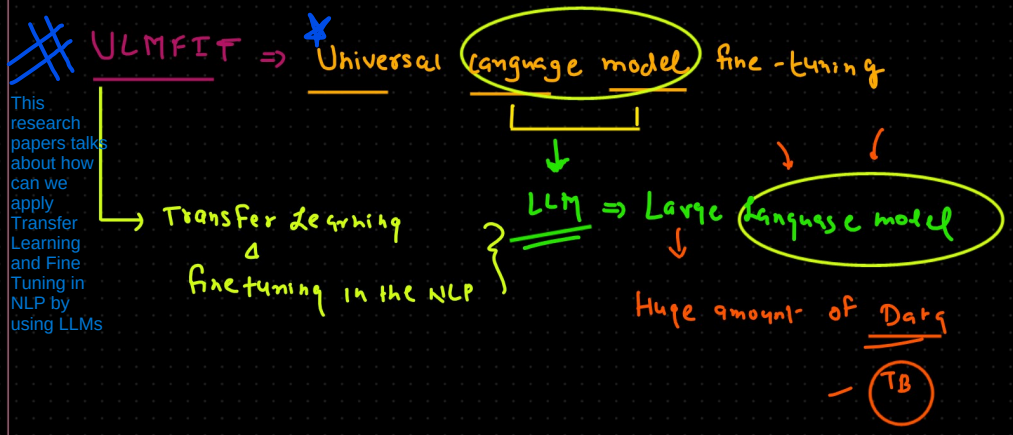
Encoder + Decoder
1. T5 (Flant T5)
2. BART
3. M2M-100
4. BigBird

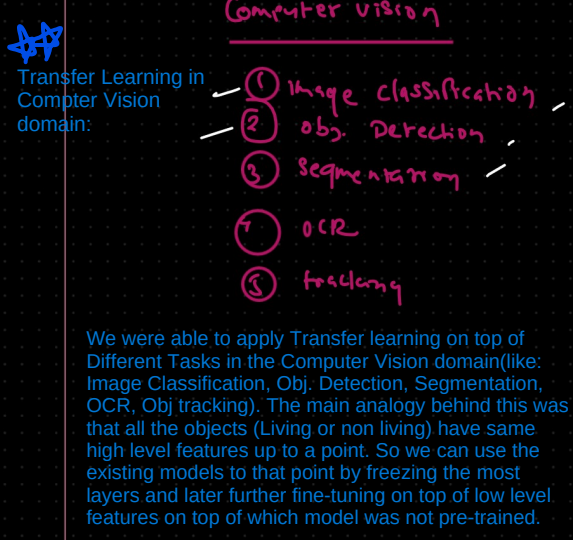
Decoder
1. GPT
2. GPT-2 → CTRL
3. GPT-3, 3.5, 4
4. GPT-Neo, GPT-J

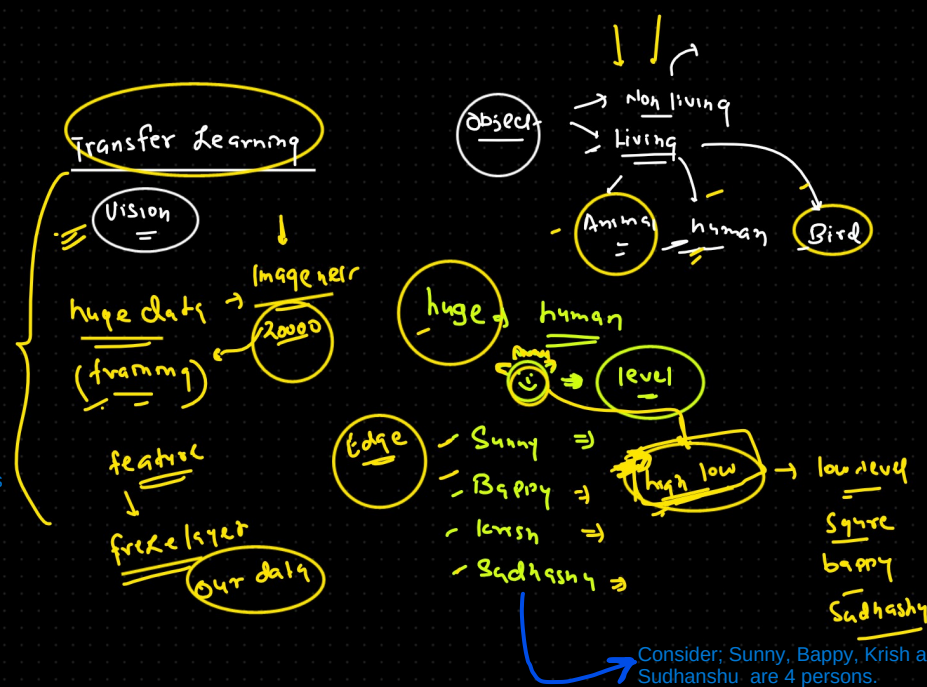
⑥ ELECTRA ⑤ Megatron (Megatron)

Chronological Progression in NLP domain

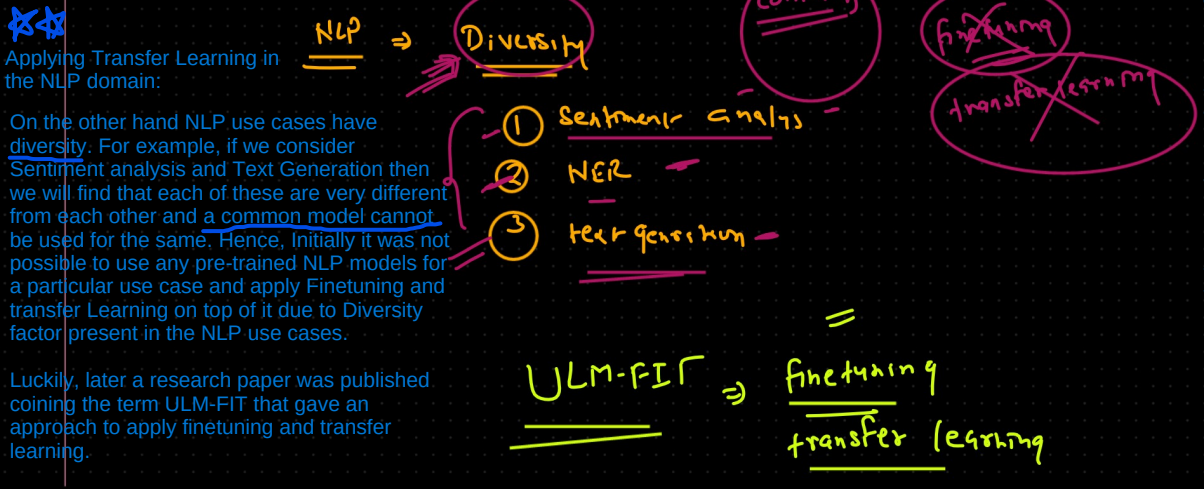


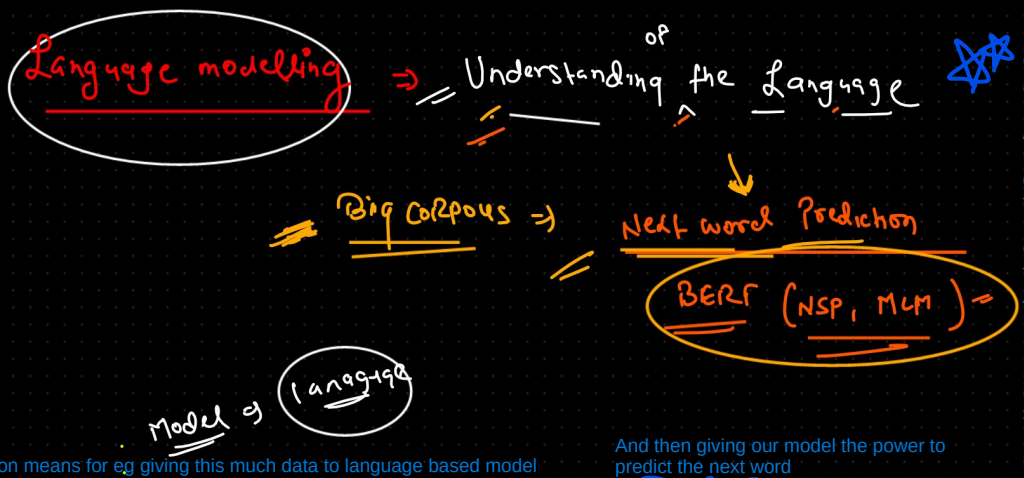






Consider; Sunny, Bappy, Krish and Sudhanshu are 4 persons.
So here high level feature can be overall edges detection as a human which will be same for all the human entity. But when we wish to establish individual distinction then we have to consider low level features like face type(Square, round, oval etc,), nose type etc. In other words here we can use a common model to identify human entity and train the existing model on top of pictures of Sunny, Bappy, Krish and Sudhanshu to make a model capable to identify these persons differently. Here we were able achieve this using a single model since there is very less diversity or high level features are common mostly.





Q: How ULM-FIT was able to apply transfer learning in NLP use cases?
Ans: Using Language modeling which meaning getting the understanding of the language.

Q: How the understanding of the language is obtained?
Ans: Using next word Prediction approach. For Ex; Understanding of the Language using BERT LLM is obtained using NSP(Next sentence prediction) and MLM(Masked language modeling)

Next word prediction means for eg giving this much data to language based model

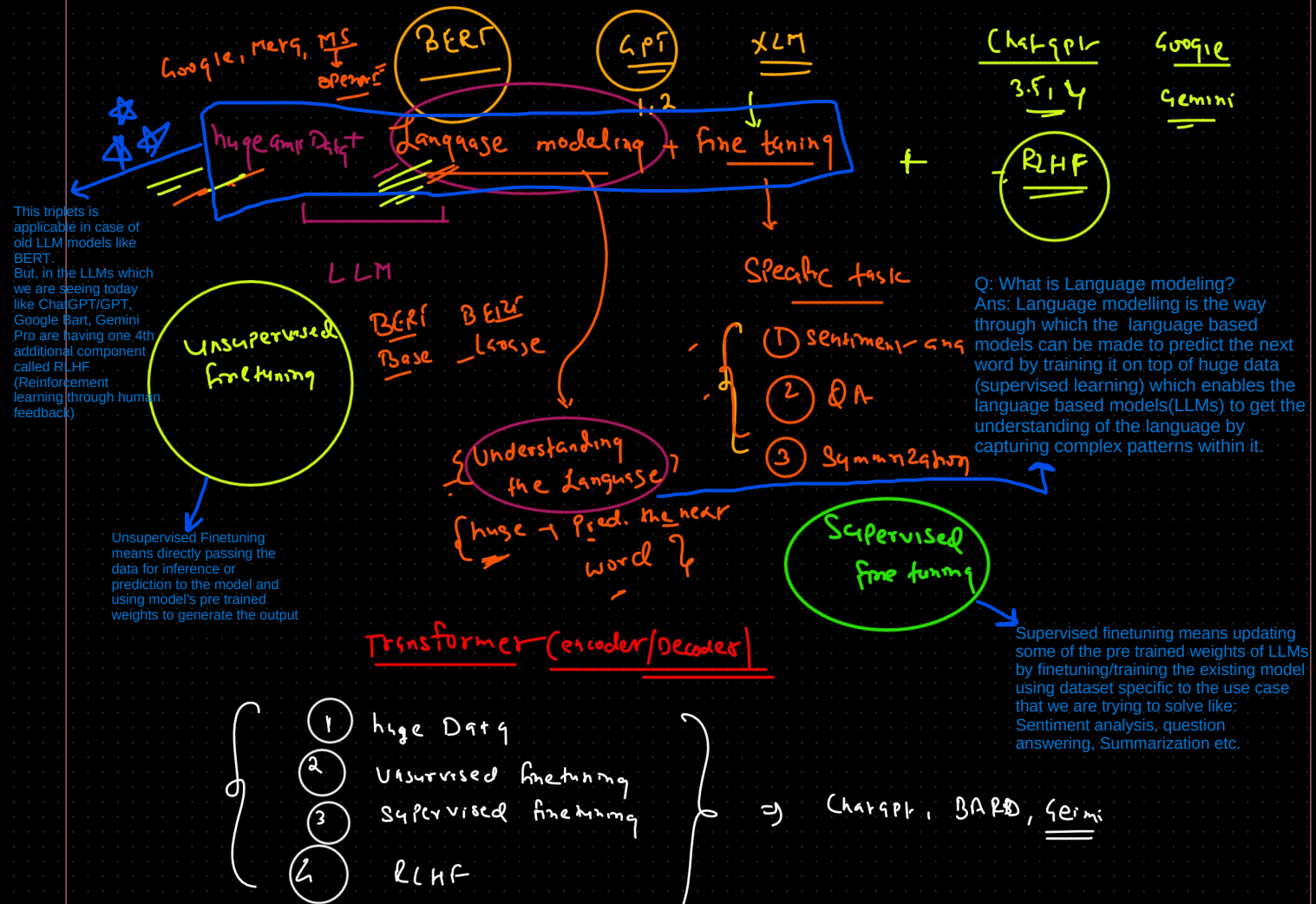
And then giving our model the power to predict the next word

Tesla, Inc. (/ˈtɛslə/ TESS-lə or /ˈtɛzlə/ TEZ-lə[a]) is an American multinational automotive and clean energy company headquartered in Austin, Texas, which designs and manufactures electric vehicles (cars and trucks), stationary battery energy storage devices from home to grid-scale, solar panels and solar shingles, and related products and services. Its subsidiary Tesla Energy develops and is a major installer of photovoltaic systems in the United States and is one of the largest global suppliers of battery energy storage systems with 6.5 gigawatt-hours (GWh) installed in 2022.

Tesla is one of the world's most valuable companies and, as of 2023, is the world's most valuable automaker. In 2022, the company led the battery electric vehicle market, with 18% share.

Tesla was incorporated in July 2003 by Martin Eberhard and Marc Tarpenning as Tesla Motors. The company's name is a tribute to inventor and electrical engineer Nikola Tesla. In February 2004, via a $6.5 million investment, Elon Musk became the company's largest shareholder. He became CEO in 2008. Tesla's announced mission is to create products which help "accelerate the world's transition to sustainable energy."

The company began production of its first car model, the Roadster sports car, in 2008. This was followed by the Model S sedan in 2012, the Model X SUV in 2015, the Model 3 sedan in 2017, the Model Y crossover in 2020, the Tesla Semi truck in 2022 and the Cybertruck light-duty pickup truck in 2023.[7] The Model 3 is the all-time bestselling plug-in electric car worldwide, and in June 2021 became the first electric car to sell 1 million units globally.[8]



This triplets is applicable in case of old LLM models like BERT.
But, in the LLMs which we are seeing today like ChatGPT/GPT, Google Bart, Gemini Pro are having one 4th additional component called RLHF (Reinforcement learning through human feedback)

Q: What is Language modeling?
Ans: Language modelling is the way through which the language based models can be made to predict the next word by training it on top of huge data (supervised learning) which enables the language based models(LLMs) to get the understanding of the language by capturing complex patterns within it.

Unsupervised Finetuning means directly passing the data for inference or prediction to the model and using model's pre trained weights to generate the output

Supervised finetuning means updating some of the pre trained weights of LLMs by finetuning/training the existing model using dataset specific to the use case that we are trying to solve like: Sentiment analysis, question answering, Summarization etc.

Decoding ULMFIT research paper: **ULMFIT** → Transfer Learning & fine tuning (2018 → Published year)

text classification → Here transfer learning and Fine tuning is done on top of text classification use case

1. Model Architecture
   1. RNN
   2. LSTM

Instead of using transformers this used the traditional RNN and LSTM to build the language model which can be enabled for Transfer learning and Fine tuning

2. Fine tuning Methodology
   1. General Language model Pre training
      - huge data & Langage understanding

In this language modeling is done meaning enabling language model to predict next word by training it using huge data. Also called Pre-training and the language model obtained from this is called pre-trained model

   2. Target task Language model Fine tuning
      - text classification

As name suggest here fine tuning is performed on top of target task or use case where a pre-trained model is intended to make predictions. In ULMFIT it is text classification use case.

Please note that:
- Through General Language model Pre Training we are enabling the language model for Transfer Learning.
- Whereas, through Target task language model fine tuning we are enabling the language model for Fine Tuning.

Similarly, decoding the BERT research paper: **BERT** ⇒

1. Model

Unlike ULMFIT BERT is build using Transformer(Encoder Only Transformer)

Transformer → Encoder

- BERT BASE → 12 Encoder
- BERT Large → 24 Encoder

These are the two variants of BERT model

2. Fine tuning methodology

Language modeling is used for Pre-training or building the Pre-trained BERT model.

1. Language Modeling ⇒ Pre-training ⇒ (Predicting the next word in sentence)
   - = Understanding of the Language

According to the research paper Target task language model fine tuning is done on top of Sentiment analysis use case.

2. target-task ⇒ Sentiment classification

Some common terms related to the BERT research paper:

1. BERT ⇒ Bi-Directional Encoder Representation from transformer
   - Bi-Directional → Both Direction
2. Transformer → Self attention, skip connection, Norm, NN ⇒ Encoder (Typo)
3. Pre-training ⇒ init Phase → large Corpus → Understanding [next word Pred]
4. Fine-tuning ⇒ Task specific Data ⇒ train
5. MLM ⇒ Mask Language modeling
6. NSP ⇒ Next sent Pred
7. Contextual embedding ⇒ position Embed
8. CLS Token ⇒ init token ⇒ classification
9. Sep token ⇒ Segmentation

If time permits read about the ULMFIT, GPT, BERT research papers from Jay Alammar's blog

Difference b/w BERT and GPT language models:

Feature	BERT	GPT
Training obj.	MLM (Masked Language modelling) Predicts the masked based on context Masked word here represents the next word which we are intend to predict	Autoregressive Language Model: Predicting the next word based on the preceding context.
Architecture	Transformer ⇒ Encoder with Bi-Directional context understanding	Transformer → Decoder unidirection context genera-tion.
Context-understanding	Bi-directional context	Unidirectional context left-to right ⇒ next word
Usecase	text classification NER QA	Text generation story completion Code generation
Pre-training	{Huge amt} → Powerful	huge variety and amt of Data ⇒ 175B

Fine tuning

{Commonly fine-tuned for specific task}

it can be fine-tune

Bert → generation

Token masking

Uses mask lang. model

masking of the token

Pred

Here masking of tokens (Input) is done and the same is used to draw prediction based on the context.

Does not perform token masking (Autoregressive)

Doesn't require masking instead it is a autoregressive model meaning it can generate the next word by using the context of the preceding words.

Model Size

BERT. BASE

BERT Large

GPT-1
GPT-2
GPT-3, 3.5, 3.5 turbo
GPT-4
GPT-4V

Mathematics ⇒ transformer → encoder / decoder

10% of 100 → 10 ⇒ MASK → Predict

Understanding the mask language modelling concept used in the BERT:

Context

{I am Sunny I am a [MASK] who love to work with ML, DL, NLP etc. I know the Big data. I know how to Define Data Pipeline.

Predict

So, this masking is applied during pre-training the model.
During pre-training 10-15% of a corpus or single sentence is masked(Hidden). Then model is enabled to perform the prediction of the masked words by using the context captured from the preceding and succeeding words.

MLM ⇒ Masked Language modelling

BERT

{Understanding the Language}

Ques: Say for example your manager asked you to implement question and answering use case. Also, you are provided with very less or limited resources. So which variant of BERT will you use or how are you going to approach this problem statement?

Ans: Well there are different variants of BERT like: ALBERT(A Lite BERT), RoBERTa, ELECTRA, DistilBERT, TinyBERT. As we know BERT language based model can be used for solving the mentioned usecase - Question and Answering. So, firstly we are going to read all the research papers describing the different variants of BERT. Then we are going to see which one consists of less parameters also we are going to see the Evaluation or Benchmark evaluation which will basically help us in understanding that how well the chosen variant will work on top our use case.
One can also google about the different BERT variants, read the different blogs and then take a reasonable decision.