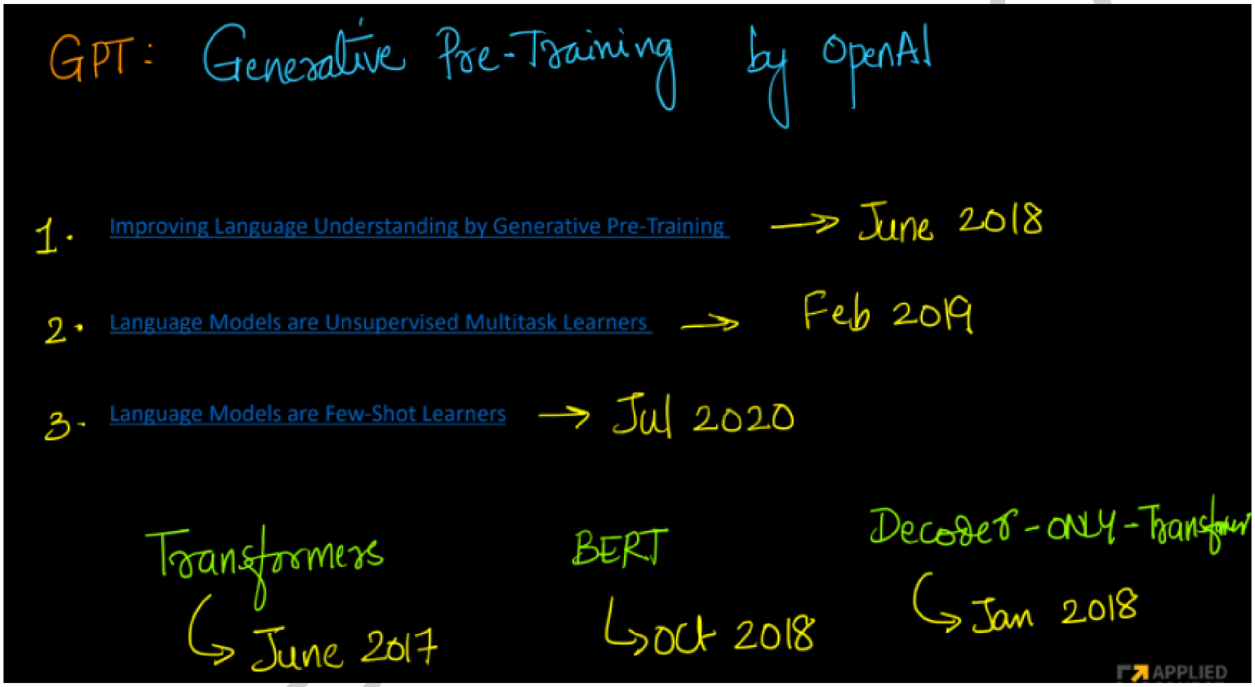
GPT-1, 2 and GPT-3 Models

GPT Models:

The focus is mainly on understanding the internals of GPT models 1,2 & 3. The fundamental prerequisite for this concept is to have knowledge of Transformers.

GPT stands for Generative Pre-training.



GPT: Generative Pre-Training by OpenAI

1. Improving Language Understanding by Generative Pre-Training → June 2018
2. Language Models are Unsupervised Multitask Learners → Feb 2019
3. Language Models are Few-Shot Learners → Jul 2020

Transformers ↳ June 2017

BERT ↳ Oct 2018

Decoder-ONLY-Transformer ↳ Jan 2018

Refer to the first Research paper for GPT-1 and second paper for GPT-2 and the third paper for GPT-3

1. [Improving Language Understanding by Generative Pre-Training]
2. [Language Models are Unsupervised Multitask Learners]
3. [Language Models are Few-Shot Learners]

All these research papers are from Open AI.

Transformers concept is invented in 2017 and this concept is extended to BERT in October of 2018 after the GPT-1 is published. In January of Decoder-only-Transformer paper is published.

The below diagram is from the actual Transformer paper.

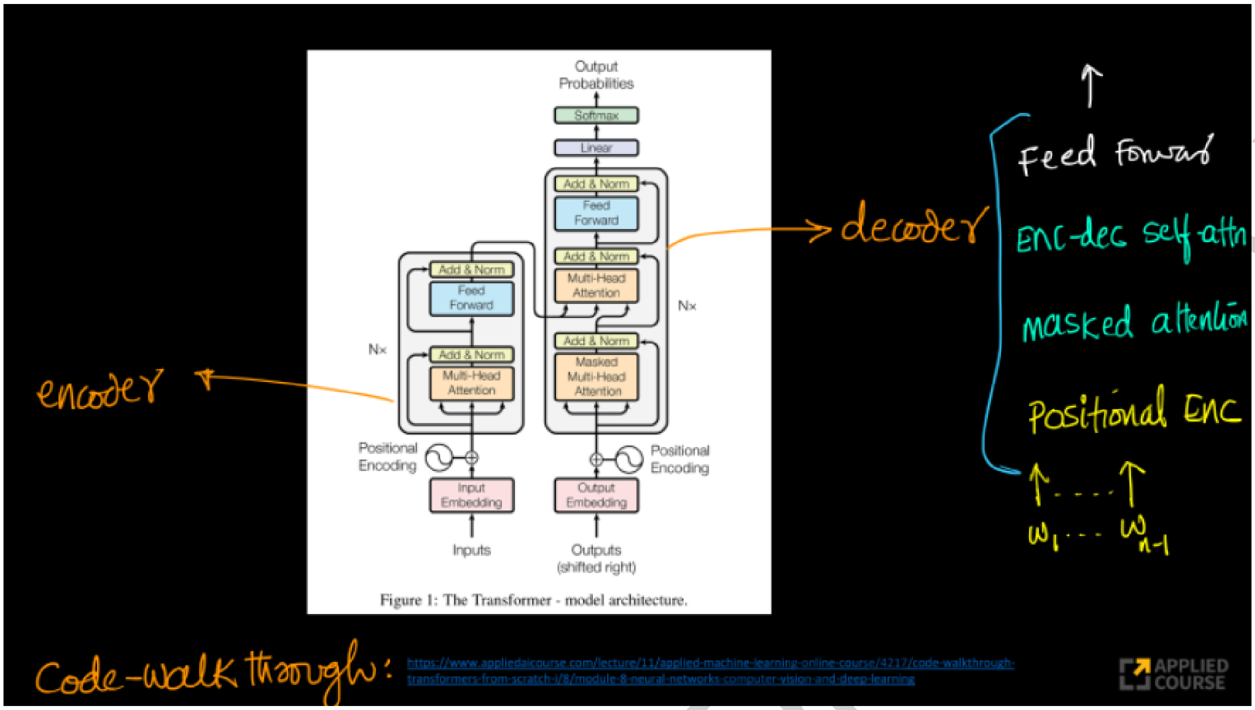


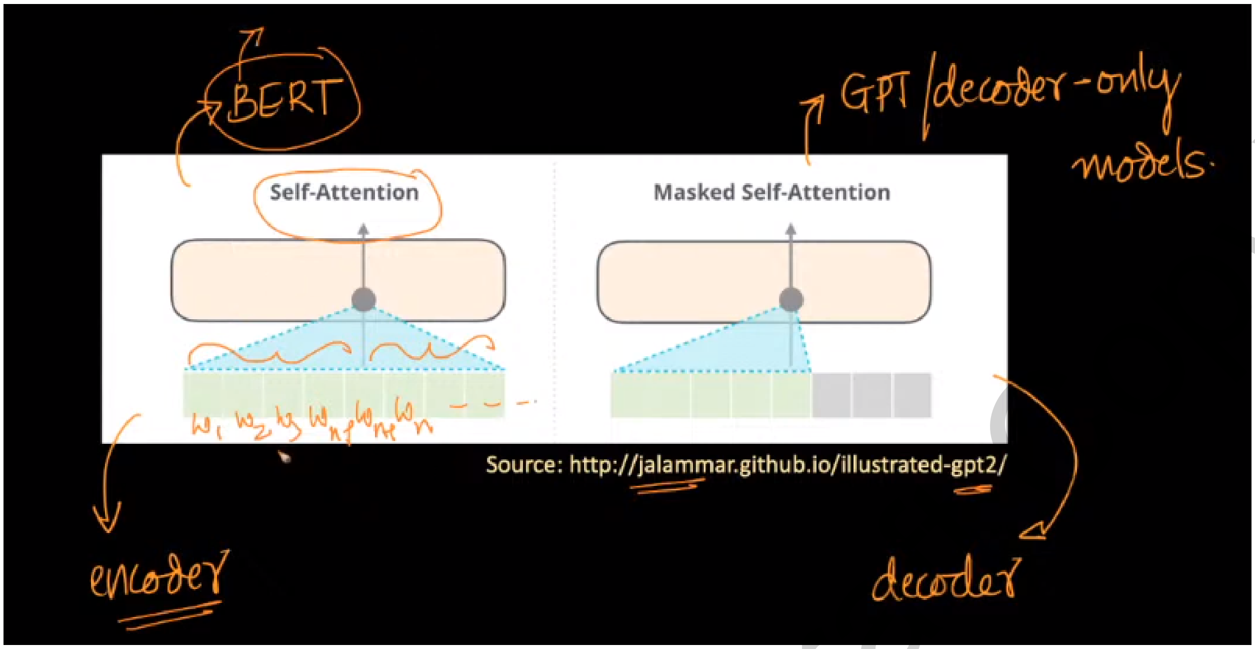
Figure 1: The Transformer - model architecture.

Code-walk through: https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/4217/code-walkthrough-transformers-from-scratch-i/8/module-8-neural-networks-computer-vision-and-deep-learning

This has been implemented from scratch in this Live session.

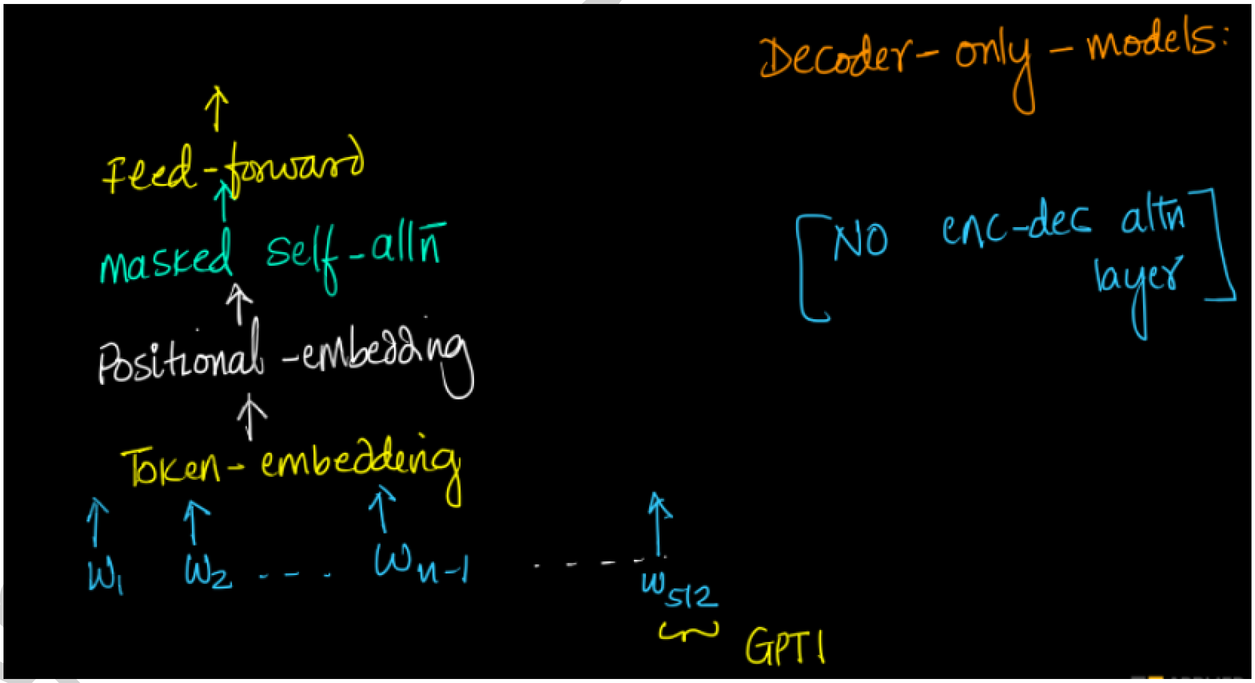
GPT-1:

In typical BERT [Encoder only stack], using Self Attention, we can look for past and also future words.
But using masked Self-attention, we can only look till the present word. All the future words are masked.



Source: http://jalammar.github.io/illustrated-gpt2/

BERT uses Self-Attention and GPT/decoder-only-Models use masked self-attention, so, it works as time step based model. So there is no concept of timesteps in the BERT as all the words are visible already. The Basic decoder-only models work as follows.



We have words w1,w2, w3,..wn and we have token embedding, then positional embedding, then masked self-attention, feed-forward.

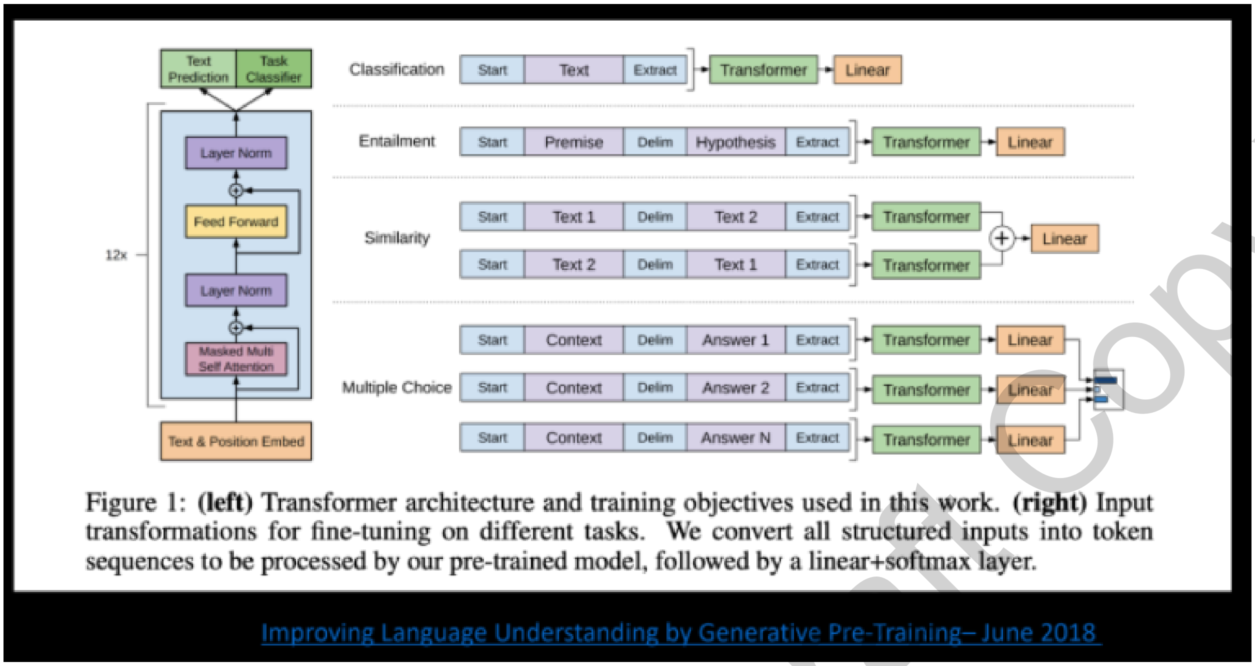


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

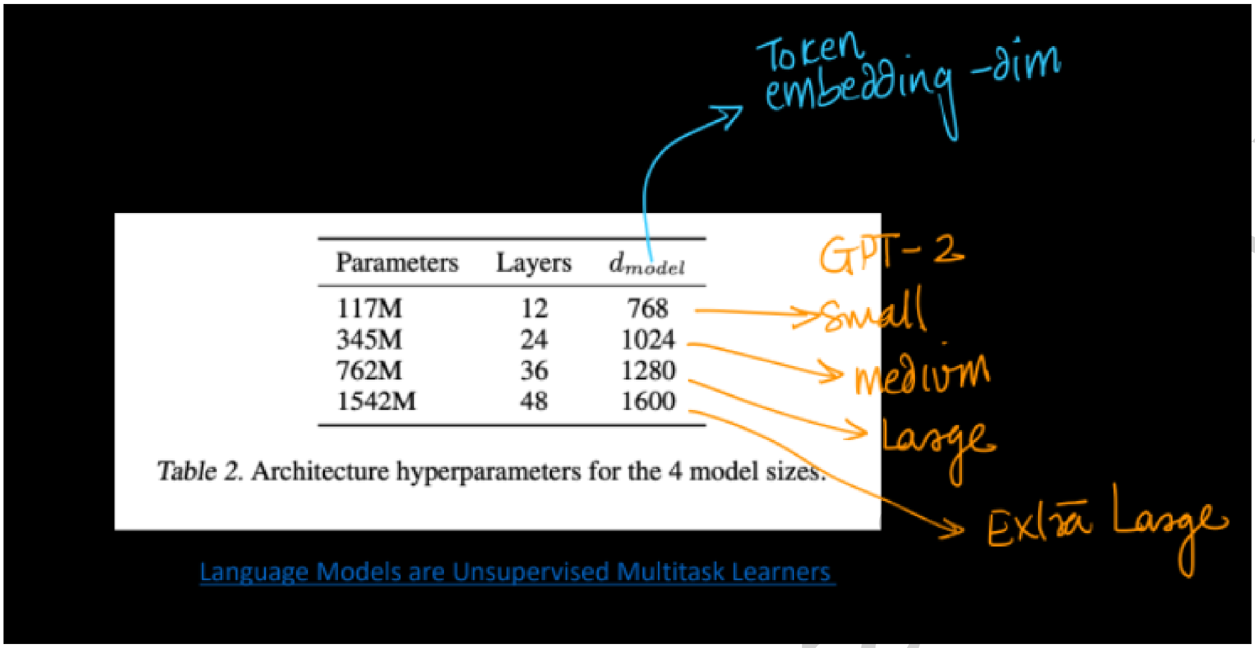
Improving Language Understanding by Generative Pre-Training– June 2018

For GPT-1 models, the max input length number is 512. And there are 12 decoders. The output from the layer Norm will be a vector of some length and can be used for different applications/tasks
For example, If the task is Classification, pass <start>text<Extract> into the transformer of 12 decoders and extract the vector. Build a simple linear classifier model on top of it.

GPT-2

From an Architectural standpoint, GPT-2 is exactly the same as GPT-1 except for very minor changes.
Here are some changes:

1. The max input length is 1024. So that, longer sentences can be used
2. Vocabulary size is increased to 50,257 words.
3. The Layer normalization is added at the input of each sub-block.
4. Larger batch sizes are used.



Parameters	Layers	d_{model}
117M	12	768
345M	24	1024
762M	36	1280
1542M	48	1600

Table 2. Architecture hyperparameters for the 4 model sizes.

Language Models are Unsupervised Multitask Learners

GPT-2 was a combination of 4 models. Small, Medium, Large, Extra-large. The GPT-2 has used 80million documents of common crawl data and Wikipedia data. This is ten times more data than GPT-1. And ten times more parameters.

Zero-shot Learning:

Usually, If we are training for Q/A:

You can pre-train on the next word prediction task and fine-tune the model and then we pass a text to the model for prediction.

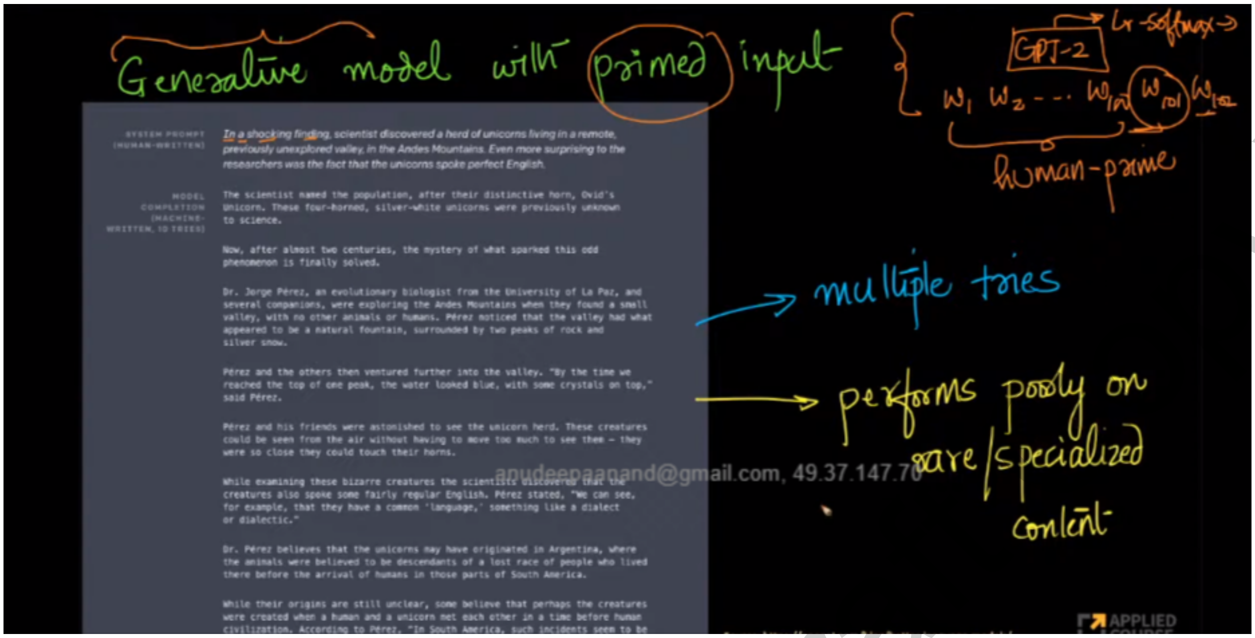
But in Zero-shot learning, we do not finetune the model.

Example of Generative model:

We are pretraining the GPT-2 models such that given a series of words, it predicts the next word.

This is called the Generative model with primed input. Primed Input because humans are priming the model with some input and then asking for predictions.

But it is observed that the 50% of outputs given from the model are sensible for general topics and poor on specialized topics.


Generative model with primed input
GPT-2
Lr-softmax →
$w_1\ w_2 \ldots w_{100}\ w_{101}\ w_{102}$
human-prime
multiple tries
performs poorly on rare/specialized content
anudeepaanand@gmail.com, 49.37.147.70
APPLIED

SYSTEM PROMPT (HUMAN-WRITTEN)

In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

MODEL COMPLETION (MACHINE-WRITTEN, 10 TRIES)

The scientist named the population, after their distinctive horn, Ovid's Unicorn. These four-horned, silver-white unicorns were previously unknown to science.

Now, after almost two centuries, the mystery of what sparked this odd phenomenon is finally solved.

Dr. Jorge Pérez, an evolutionary biologist from the University of La Paz, and several companions, were exploring the Andes Mountains when they found a small valley, with no other animals or humans. Pérez noticed that the valley had what appeared to be a natural fountain, surrounded by two peaks of rock and silver snow.

Pérez and the others then ventured further into the valley. "By the time we reached the top of one peak, the water looked blue, with some crystals on top," said Pérez.

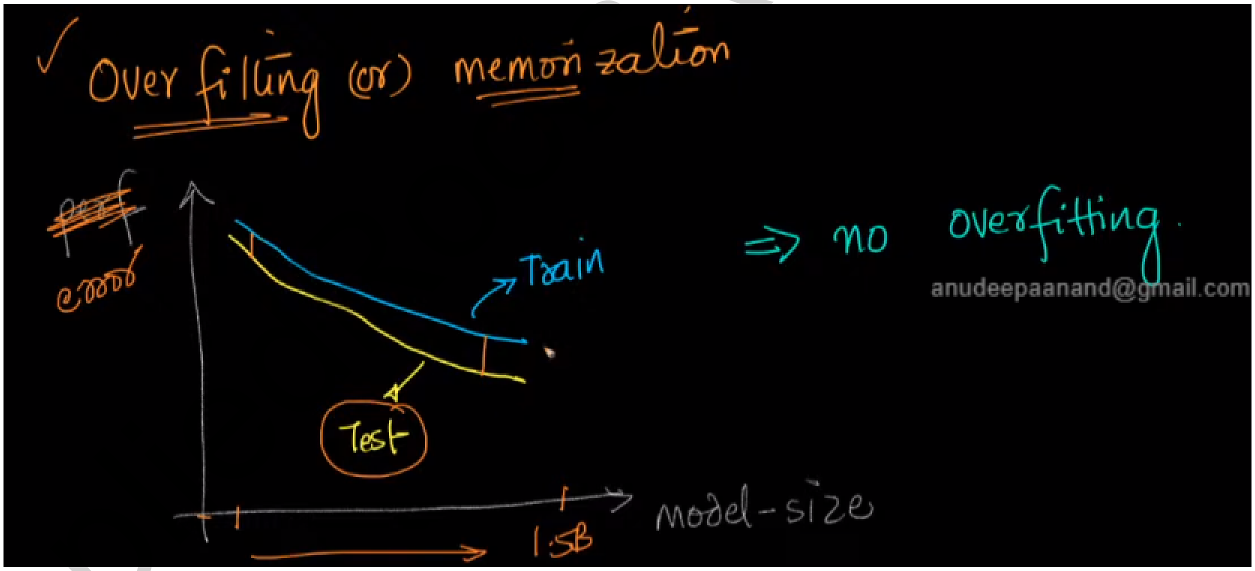
Pérez and his friends were astonished to see the unicorn herd. These creatures could be seen from the air without having to move too much to see them – they were so close they could touch their horns.

While examining these bizarre creatures the scientists discovered that the creatures also spoke some fairly regular English. Pérez stated, "We can see, for example, that they have a common 'language,' something like a dialect or dialectic."

Dr. Pérez believes that the unicorns may have originated in Argentina, where the animals were believed to be descendants of a lost race of people who lived there before the arrival of humans in those parts of South America.

While their origins are still unclear, some believe that perhaps the creatures were created when a human and a unicorn met each other in a time before human civilization. According to Pérez, "In South America, such incidents seem to be

The graph shows the perplexity with respect to the model size.
As the model size increases, the training data makes fewer errors means the error decreases. Similarly, the test error also decreases.

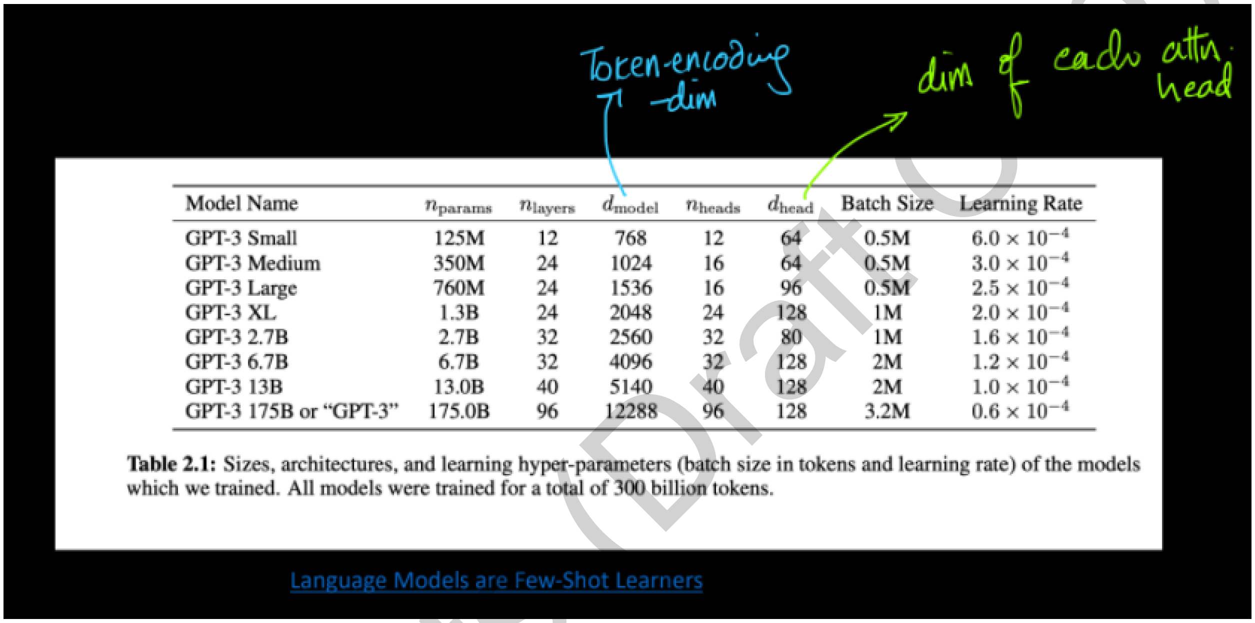


So, the key takeaway from this paper can be decoder-only models can work for a huge amounts of data without overfitting.

GPT-3:

GPT-2 is not very different from GPT-2.
The difference being:

1. The largest GPT-3 model has 150B parameters [100 times more than GPT-2].

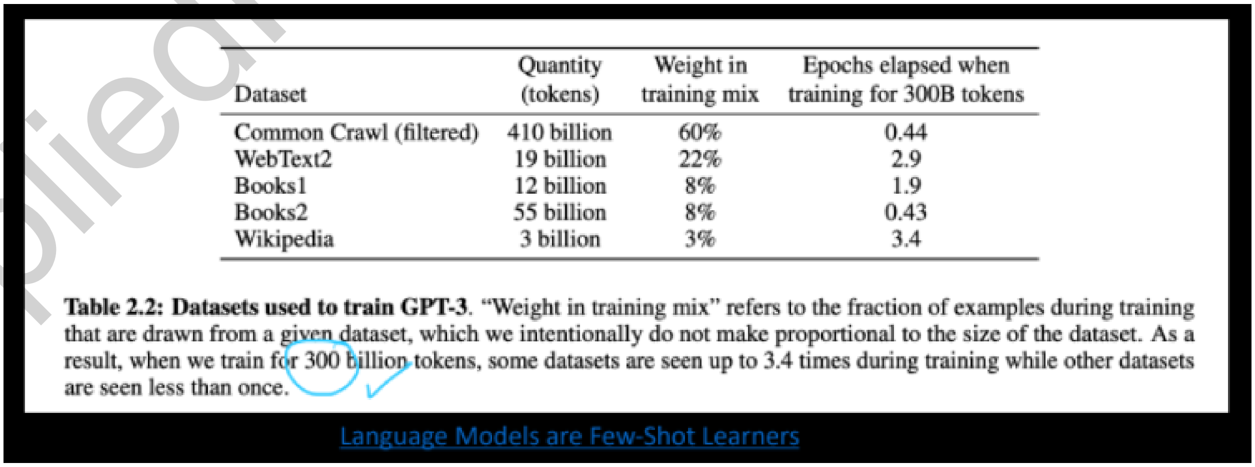

Token-encoding -dim
dim of each attn. head

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

Table 2.1: Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

Language Models are Few-Shot Learners

2. The largest dataset has been used. 300B tokens of text have been used for training. Common crawl data of 3 years has been used which weights about 60% of total data. And some specified books nd Wikipedia data has also been used.



Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

Table 2.2: Datasets used to train GPT-3. "Weight in training mix" refers to the fraction of examples during training that are drawn from a given dataset, which we intentionally do not make proportional to the size of the dataset. As a result, when we train for 300 billion tokens, some datasets are seen up to 3.4 times during training while other datasets are seen less than once.

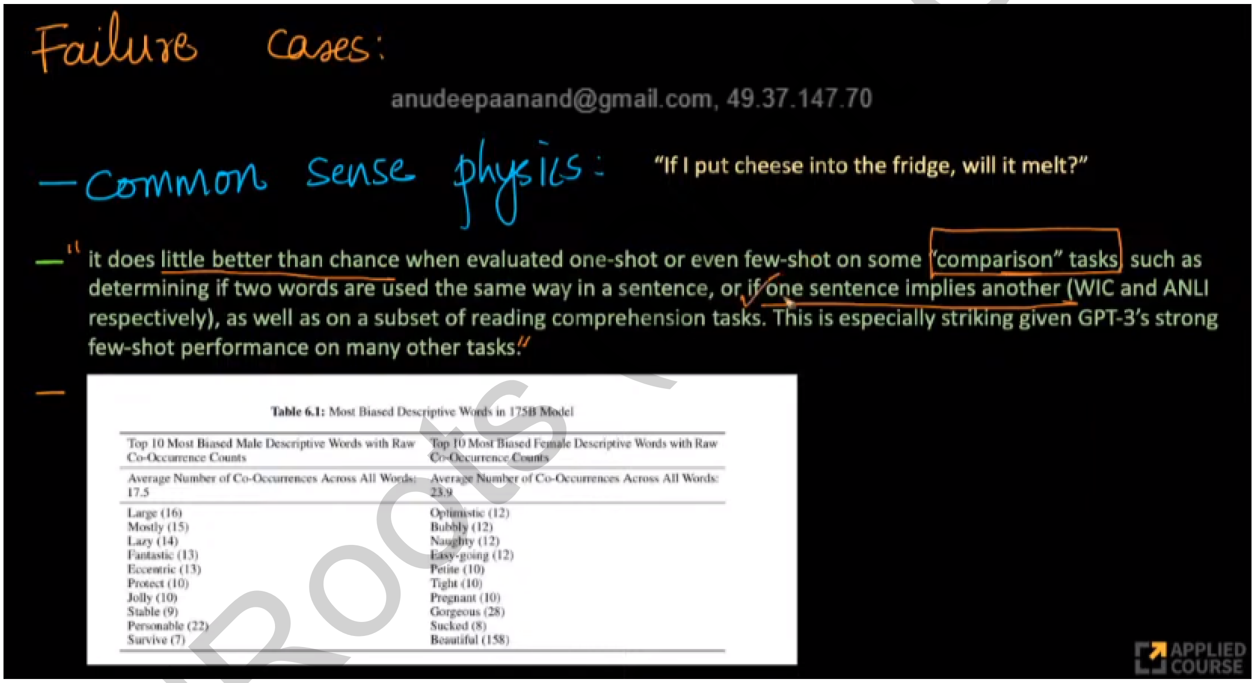
Language Models are Few-Shot Learners

3. The one innovative addition is they have to use Sparse Attention [sqrt(N) positions instead of N]

Imagine if we have N-words, All words have been used in self-Attention which creates computational complexity.
Sparse Attention says very often we don't need all the words as input, just a subset is enough. This can be implemented in multiple ways.

Failure Cases:

1. It fails for common sense physics. That is the model is failing for reasoning questions.



Failure Cases:

— Common sense physics: "If I put cheese into the fridge, will it melt?"

— "it does little better than chance when evaluated one-shot or even few-shot on some "comparison" tasks, such as determining if two words are used the same way in a sentence, or if one sentence implies another (WIC and ANLI respectively), as well as on a subset of reading comprehension tasks. This is especially striking given GPT-3's strong few-shot performance on many other tasks."

—

Table 6.1: Most Biased Descriptive Words in 175B Model

Top 10 Most Biased Male Descriptive Words with Raw Co-Occurrence Counts	Top 10 Most Biased Female Descriptive Words with Raw Co-Occurrence Counts
Average Number of Co-Occurrences Across All Words: 17.5	Average Number of Co-Occurrences Across All Words: 23.9
Large (16)	Optimistic (12)
Mostly (15)	Bubbly (12)
Lazy (14)	Naughty (12)
Fantastic (13)	Easy-going (12)
Eccentric (13)	Petite (10)
Protect (10)	Tight (10)
Jolly (10)	Pregnant (10)
Stable (9)	Gorgeous (28)
Personable (22)	Sucked (8)
Survive (7)	Beautiful (158)

2. They are lots of bias problems. The model is turning out as biased towards a particular gender, race, or something.
3. The training and Inference are extremely costly.

One way to solve the Inference problem? There is an idea called model distillation. Let's assume a large model which is already trained. Given input and we get the output yi. Now create a new data set "D_L "and then give it as input for prediction.
This smaller model when trained carefully, gives correct mappings.

These days, many are using GPT models for generative purposes. That is when we give input, the model has to generate the next word.