Lecture5: A closer look at GPT

① ZERO SHOT vs FEW SHOT LEARNING

Zero Shot: Ability to generalize to completely unseen tasks without any prior

Zero Shot: Ability to generalize to completely unseen tasks without any prior specific examples.

Few Shot: Learning from a minimum number of examples which the user provides as input.

ZERO SHOT vs FEW SHOT

Input

Output

Zero Shot

Complete task without example

Translate English to French

breakfast ➔ petit-dejeuner

Few Shot

Complete task with a few examples

gaot ➔ goat

sheo ➔ shoe

pohne ➔ phone

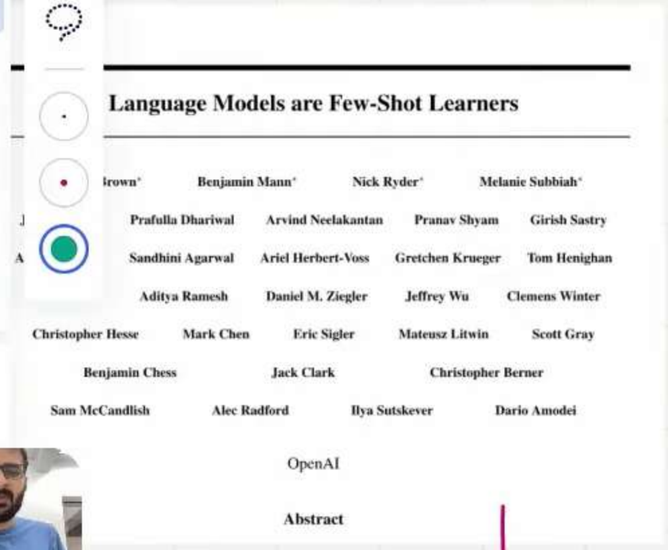
(2) UTILIZING LARGE DATASETS

Let us look at pretraining dataset of GPT3 LLM.

Dataset name	Dataset description	Number of tokens	Proportion in training data
CommonCrawl (filtered)	Web crawl data	410 billion	60%
WebText2	Web crawl data	19 billion	22%
Books1	Internet-based book corpus	12 billion	8%
Books2	Internet-based book corpus	55 billion	8%
Wikipedia	High-quality text	3 billion	3%

Books1	Internet-based book corpus	12 billion	8%
ks2	Internet-based book corpus	55 billion	8%
pedia	High-quality text	3 billion	3%

A token is a unit of text which the model reads. For now, you can think of 1 token = 1 word.



Language Models are Few-Shot Learners

rown* Benjamin Mann* Nick Ryder* Melanie Subbiah*

Prafulla Dhariwal Arvind Neelakantan Pranav Shyam Girish Sastry

Sandhini Agarwal Ariel Herbert-Voss Gretchen Krueger Tom Henighan

Aditya Ramesh Daniel M. Ziegler Jeffrey Wu Clemens Winter

Christopher Hesse Mark Chen Eric Sigler Mateusz Litwin Scott Gray

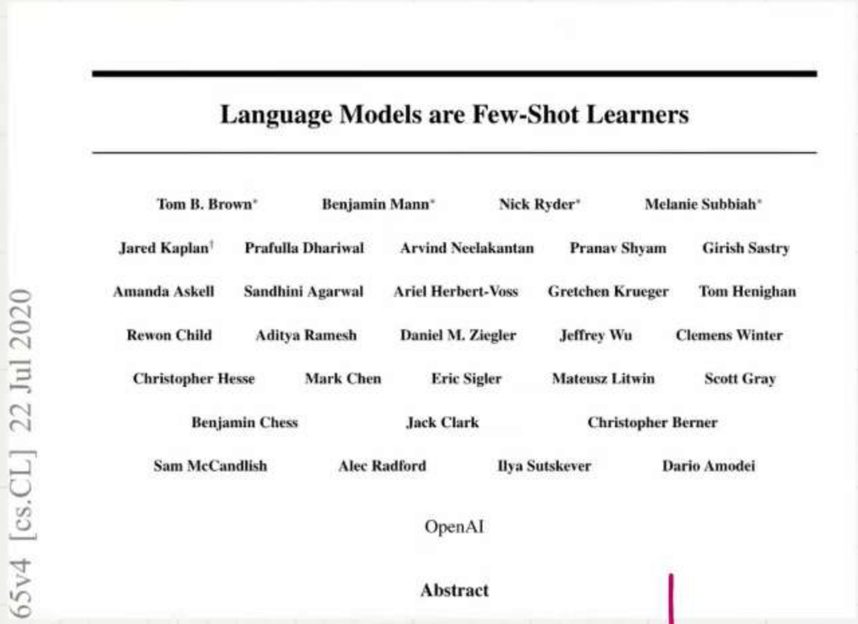
Benjamin Chess Jack Clark Christopher Berner

Sam McCandlish Alec Radford Ilya Sutskever Dario Amodei

OpenAI

Abstract

reads. For now, you can think of 1 token = 1 word.



65v4 [cs.CL] 22 Jul 2020

Language Models are Few-Shot Learners

Tom B. Brown* Benjamin Mann* Nick Ryder* Melanie Subbiah*

Jared Kaplan† Prafulla Dhariwal Arvind Neelakantan Pranav Shyam Girish Sastry

Amanda Askell Sandhini Agarwal Ariel Herbert-Voss Gretchen Krueger Tom Henighan

Rewon Child Aditya Ramesh Daniel M. Ziegler Jeffrey Wu Clemens Winter

Christopher Hesse Mark Chen Eric Sigler Mateusz Litwin Scott Gray

Benjamin Chess Jack Clark Christopher Berner

Sam McCandlish Alec Radford Ilya Sutskever Dario Amodei

OpenAI

Abstract

GPT-3 main paper

The total pre-training cost for GPT-3

miro.com/app/board/uXjVKENK1Qc=/

miro | Convolutional Neural Networks | Upgrade | Present | Share

Abstract

GPT-3 main paper

The total pre-training cost for GPT-3

≈ 4.6 million dollars

These pretrained models are base/ foundational models which can be used for further finetuning

27%

≈ 4·6 million dollars

* These pretrained models are base/ foundational models which can be used for further finetuning

* Many pretrained LLMs are available as open-source models → can be used as general purpose tools to write, extract and edit text which was not part

* These pretrained models are base/ foundational models which can be used for further finetuning

* Many pretrained LLMs are available as open-source models → can be used as general purpose tools to write, extract and edit text which was not part of the training data.

(3) GPT Architecture

GPT: Generative Pretrained Transformer.

(3) GPT Architecture

GPT: Generative Pretrained Transformer

Improving Language Understanding by Generative Pre-Training

Alec Radford OpenAI alec@openai.com
Karthik Narasimhan OpenAI karthikn@openai.com
Tim Salimans OpenAI tim@openai.com
Ilya Sutskever OpenAI ilyasu@openai.com

Abstract

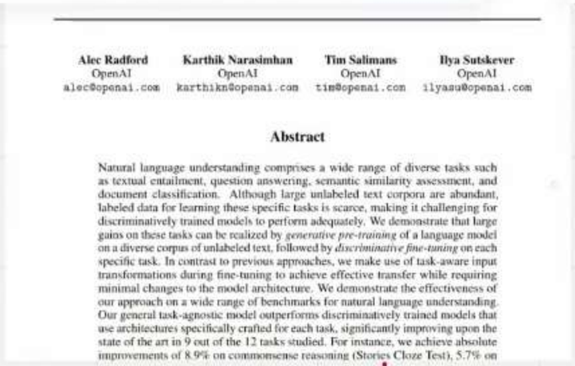
Natural language understanding comprises a wide range of diverse tasks such as textual entailment, question answering, semantic similarity assessment, and document classification. Although large unlabeled text corpora are abundant, labeled data for learning these specific tasks is scarce, making it challenging for discriminatively trained models to perform adequately. We demonstrate that large gains on these tasks can be realized by *generative pre-training* of a language model on a diverse corpus of unlabeled text, followed by *discriminative fine-tuning* on each specific task. In contrast to previous approaches, we make use of task-aware input transformations during fine-tuning to achieve effective transfer while requiring minimal changes to the model architecture. We demonstrate the effectiveness of our approach on a wide range of benchmarks for natural language understanding. Our general task-agnostic model outperforms discriminatively trained models that use architectures specifically crafted for each task, significantly improving upon the state of the art in 9 out of the 12 tasks studied. For instance, we achieve absolute improvements of 8.9% on commonsense reasoning (Stories Cloze Test), 5.7% on

→ Original paper (2018)

→ GPT 3 is a scaled up version of this model, implemented on

21%

original paper (2018)



Alec Radford, OpenAI, alec@openai.com
Karthik Narasimhan, OpenAI, karthikn@openai.com
Tim Salimans, OpenAI, tim@openai.com
Ilya Sutskever, OpenAI, ilyasu@openai.com

Abstract

Natural language understanding comprises a wide range of diverse tasks such as textual entailment, question answering, semantic similarity assessment, and document classification. Although large unlabeled text corpora are abundant, labeled data for learning these specific tasks is scarce, making it challenging for discriminatively trained models to perform adequately. We demonstrate that large gains on these tasks can be realized by *generative pre-training* of a language model on a diverse corpus of unlabeled text, followed by *discriminative fine-tuning* on each specific task. In contrast to previous approaches, we make use of task-aware input transformations during fine-tuning to achieve effective transfer while requiring minimal changes to the model architecture. We demonstrate the effectiveness of our approach on a wide range of benchmarks for natural language understanding. Our general task-agnostic model outperforms discriminatively trained models that use architectures specifically crafted for each task, significantly improving upon the state of the art in 9 out of the 12 tasks studied. For instance, we achieve absolute improvements of 8.9% on commonsense reasoning (Stories Cloze Test), 5.7% on

GPT 3 is a scaled up version of this model, implemented on a larger dataset

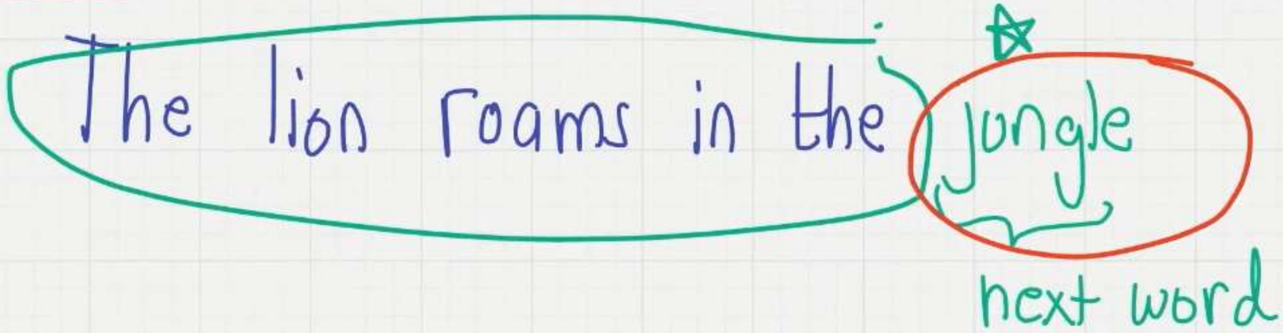
* GPT models are simply trained on "next-word" prediction tasks.

* GPT models are simply trained on "next-word" prediction tasks.

The lion roams in the jungle

next word

* With this training, they can do a wide range of other tasks like translation, spelling correction etc!



* With this training, they can do a wide range of other tasks like translation, spelling correction etc!

* Next word prediction: Self-supervised learning
  - self labeling

* Next word prediction: Self-supervised learning
  - self labeling

* We dont collect labels for training data, but use the structure of the data itself.
  - next word in sentence is used as label

→ Auto regr
use previo
for future

* Compared to original transformer architecture, GPT architecture is simpler.

self labeling

labels for training data, but
of the data itself.

word in sentence is used as label.

→ Auto regressive model :
use previous outputs as inputs
for future predictions.

nal transformer architecture,
is simpler.

miro.com/app/board/uXjVKENK1Qc=/

miro | Convolutional Neural Networks | Upgrade | Present | Share

gressive model :

outputs as inputs

redictions.

Pretraining

↓

Unsupervised,

Auto Regressive models

20%

next word in sentence is used as label

for fut

* Compared to original transformer architecture, GPT architecture is simpler.

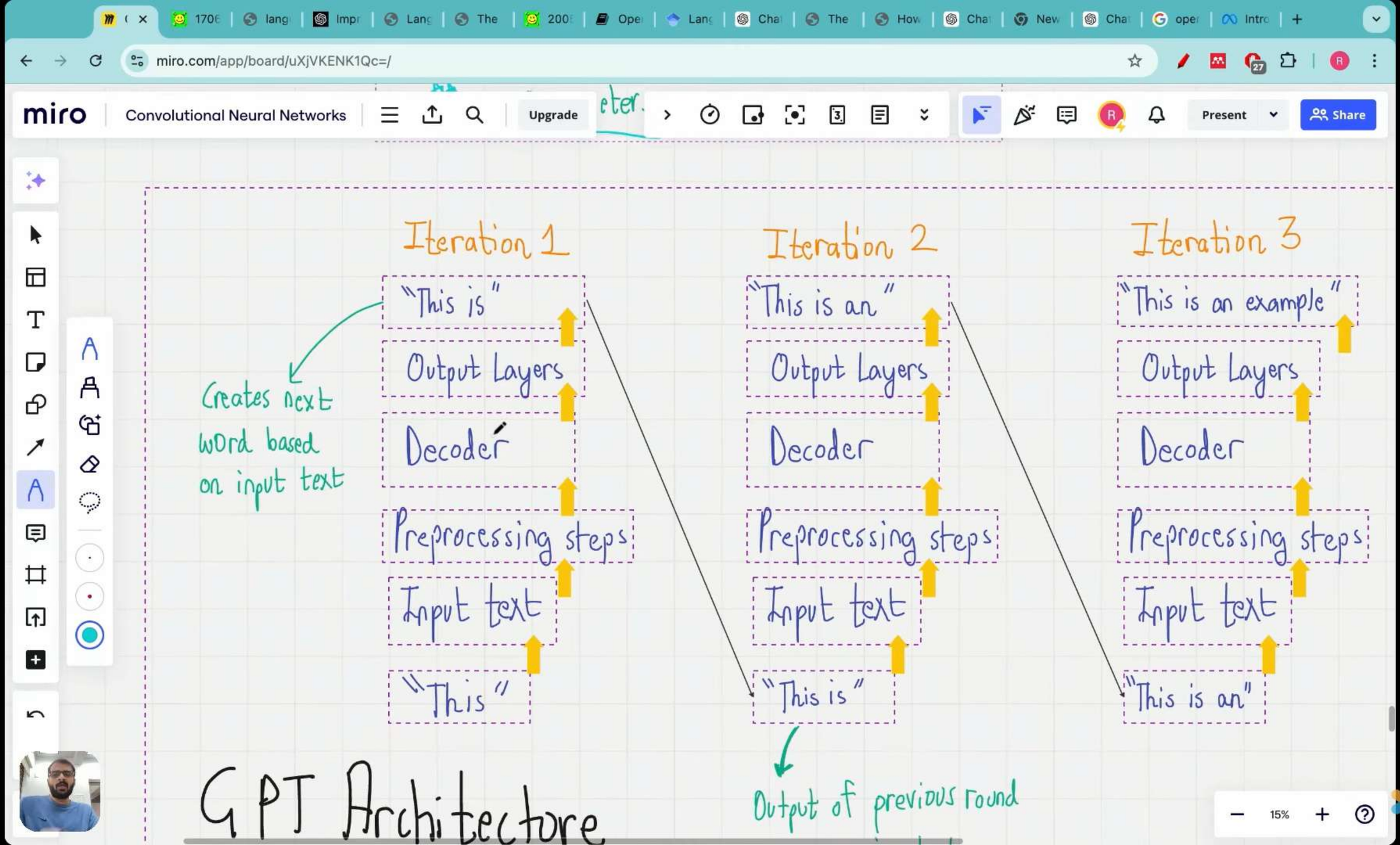
* In GPT architecture: there is no encoder. We just have the decoder.

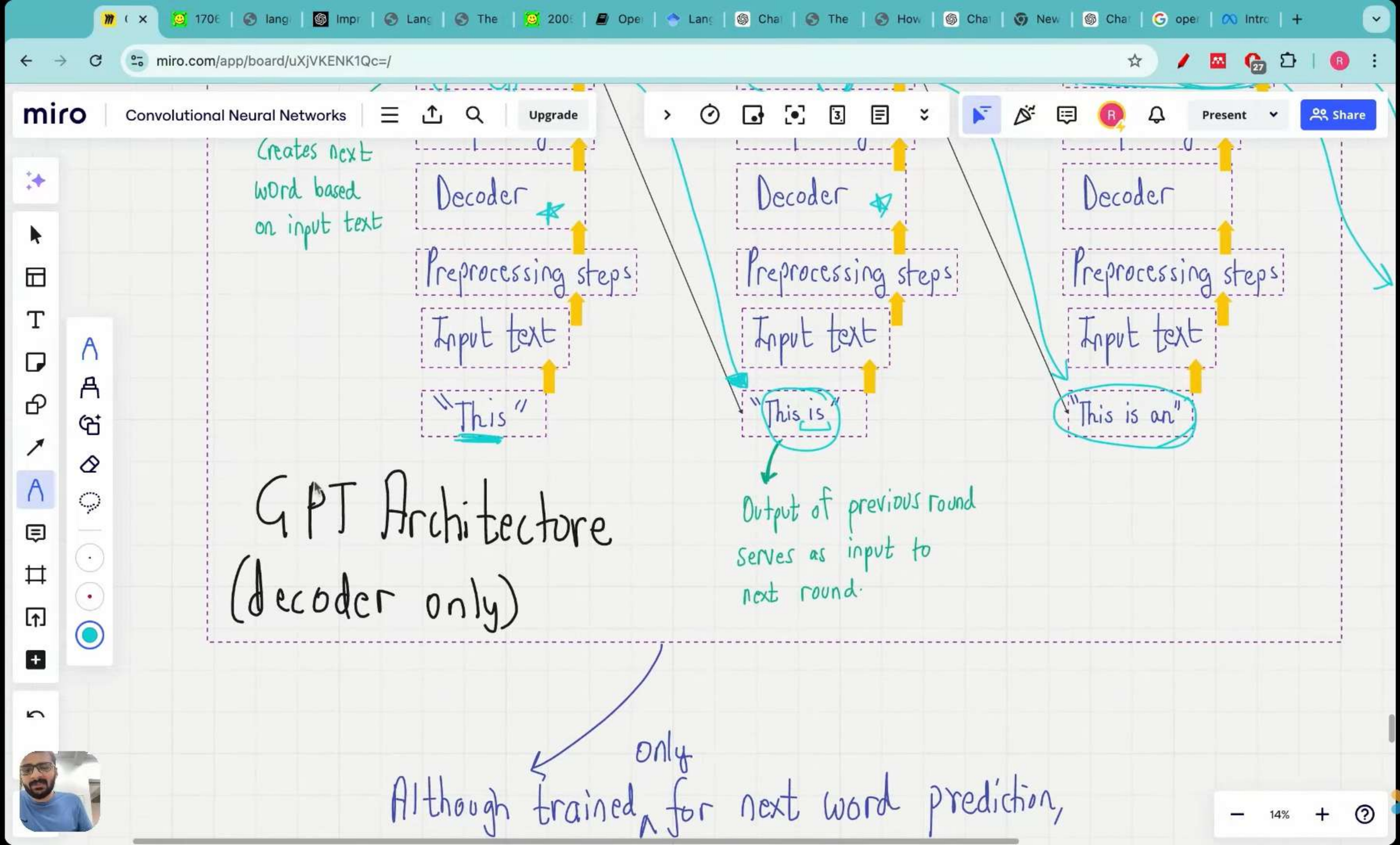
Original Transformer: 6 encoder-decoder blocks

GPT-3: 96 transformer layers, 175 billion

* In GPT architecture: there is no encoder.
We just have the decoder.

Original Transformer: 6 encoder-decoder blocks.
GPT-3: 96 transformer layers, 175 billion parameters.


miro
Convolutional Neural Networks
Upgrade
Present
Share
Iteration 1
"This is"
Output Layers
Decoder
Preprocessing steps
Input text
"This"
Creates next word based on input text
Iteration 2
"This is an"
Output Layers
Decoder
Preprocessing steps
Input text
"This is"
Output of previous round
Iteration 3
"This is an example"
Output Layers
Decoder
Preprocessing steps
Input text
"This is an"
GPT Architecture
15%


miro
Convolutional Neural Networks
Upgrade
Present
Share
Creates next word based on input text
Decoder
Preprocessing steps
Input text
"This"
"This is"
"This is an"
GPT Architecture (decoder only)
Output of previous round serves as input to next round.
Although trained only for next word prediction,
14%

(decoder only)

Although trained only for next word prediction, GPT model can perform other tasks like language translation

This is called "emergent behavior"

behavior"

Ability of a model to perform tasks that the model wasn't explicitly trained to perform

Lecture6: Stages of building LLM → Our plan for this playlist!