Text Processing Using Machine Learning

TRANSFER LEARNING
&
PRE-TRAINED MODELS

Dr. Fan Zhenzhen
zhenzhen@nus.edu.sg

Agenda

- Transfer Learning
  - Feature-based
  - Fine-tuning

- Pre-trained Models
  - BERT
  - GPT-2 & GPT-3

- Tutorials

TRANSFER LEARNING

NLP's ImageNet Moment

- **Pre-trained** ImageNet models in computer vision
  - Trained on 1.2M examples
  - used to achieve state-of-the-art results in **a variety of tasks** such as object detection, semantic segmentation, human pose estimation, and video recognition
  - modelling lower-level features like edges, as well as higher-level concepts like patterns and objects, etc.
  - allowing computer vision to be applied in domains where only a small number of labelled training examples are available

Transfer learning

Pre-trained ___?____ models in NLP

- useful for a variety of tasks like text classification, sequence labelling, coreference resolution, question answering, machine translation, natural language inference, constituency parsing, etc.

Transfer Learning

Pre-trained model

Data?

Task?

+

Fine-tuning

(training on a supervised dataset specific to a downstream task)

Language Modelling

Language modeling is the task of assigning a probability to sentences in a language. […] Besides assigning a probability to each sequence of words, the language models also assigns a probability for the likelihood of a given word (or a sequence of words) to follow a sequence of words…

Neural Network Methods in Natural Language Processing, 2017.

- Task: predict the next word given its previous words

"The service was poor, but the food was [ ]" → "delicious"

- Pros
  - Unsupervised!
  - Potentially unlimited amount of data available!
- Cons
  - Can only make use of the preceding words

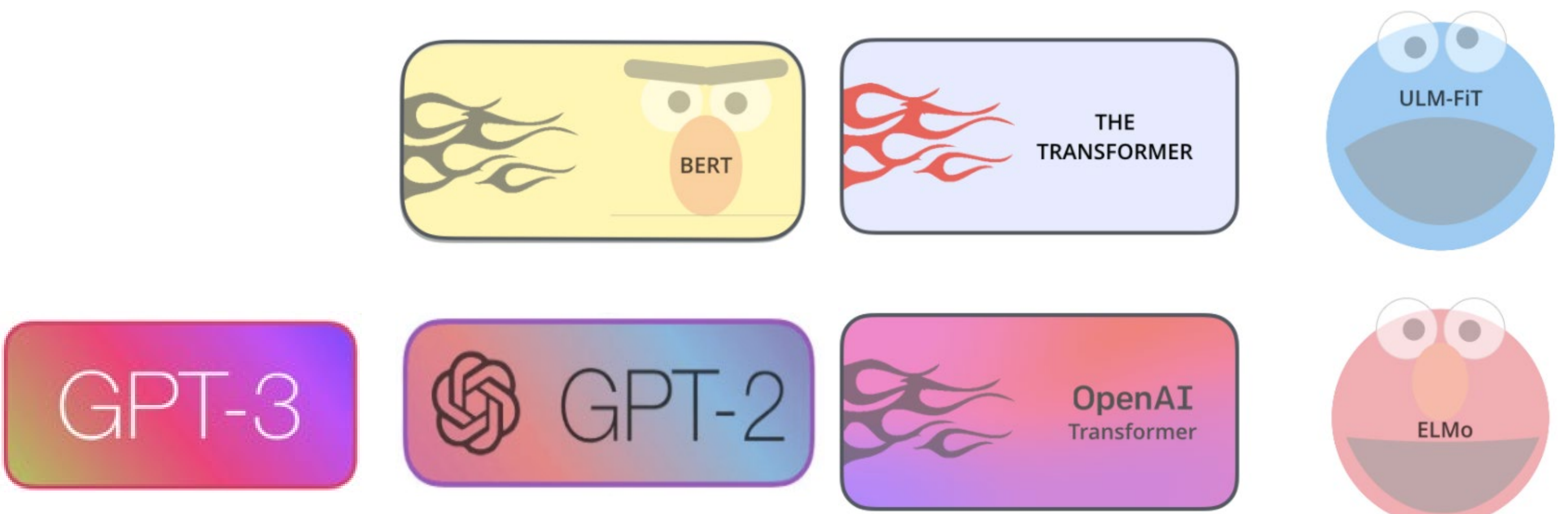
Masked Language Model

- Randomly mask some of the tokens from the input
- Task: to predict the original vocabulary id of the masked word based on its context.
- Enables training of bi-directional representation (e.g. BERT)

*“Out of **[MASK]**, out of mind” → “sight”*

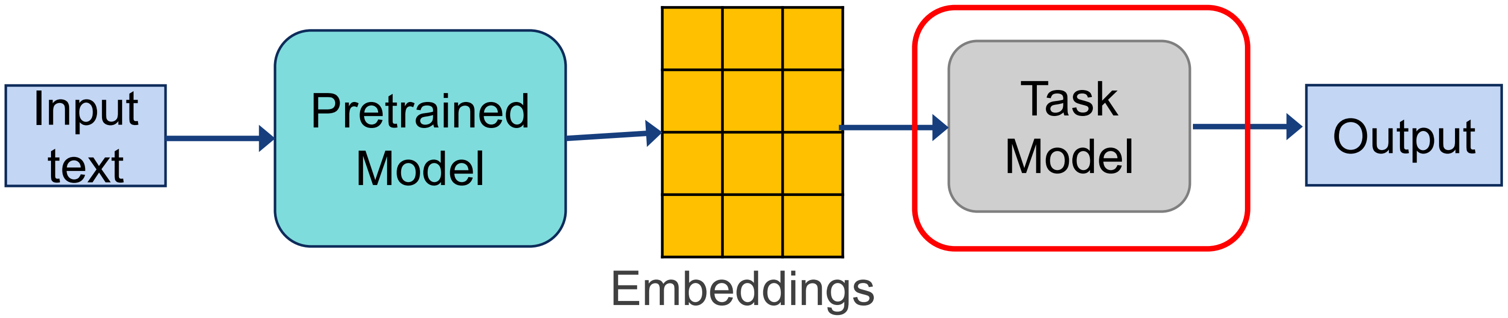
Language Modelling

- It works!
    - Embeddings from Language Models (ELMo)
    - Universal Language Model Fine-tuning (ULMFiT)
    - Transformer
    - BERT
    - OpenAI GPT/GPT-2/GPT-3



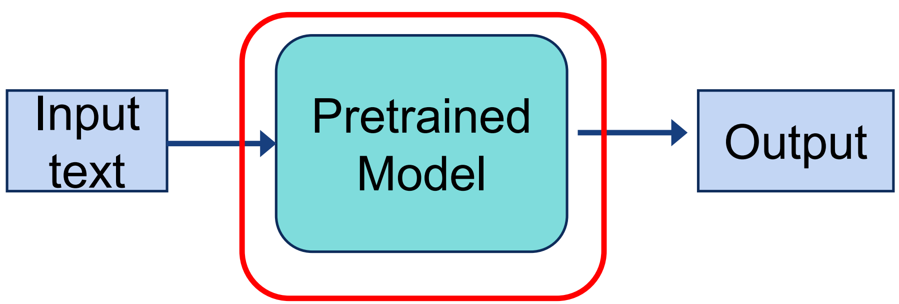
Using Pre-trained Models

- *Feature-based approach*
  - Create task-specific architecture
  - Use the pre-trained model as a feature extractor
  - The weights in the pre-trained model are not updated during training
  - e.g. ELMo



Using Pre-trained Models

- *Fine-tuning approach*
  - Introduce minimal task-specific parameters
  - Trained on the downstream task by fine-tuning all pre-trained parameters
  - e.g. OpenAI GPT



Recall the DL training routine?

- Initialize weights vector
  - Import the pre-trained weights
  - One-hot encoding
- Repeat the following until desired
  - Forward Propagation
  - Compute and log the loss
  - Back Propagation
  - Optimizer

Fine-tuning!

GPT's Generative Pre-training + Discriminative Fine-tuning

- Minimal change to the model architecture with clever input transformation

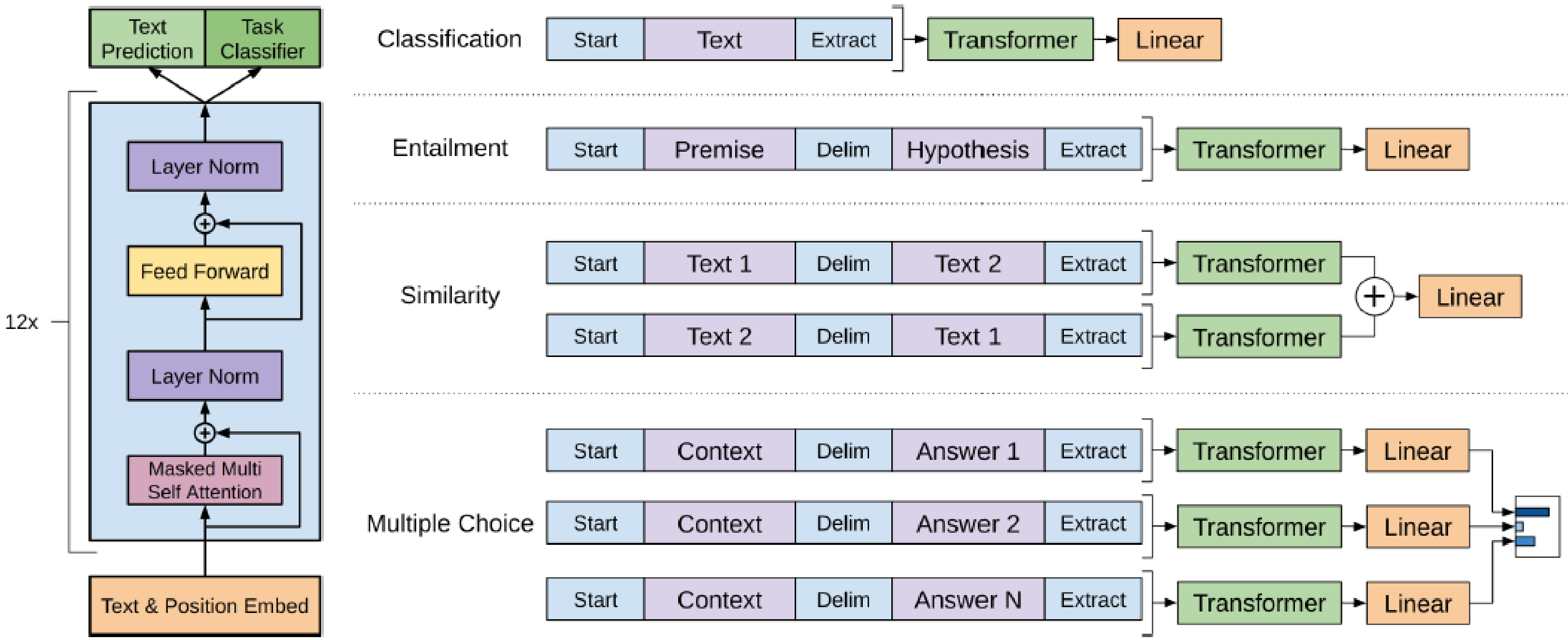
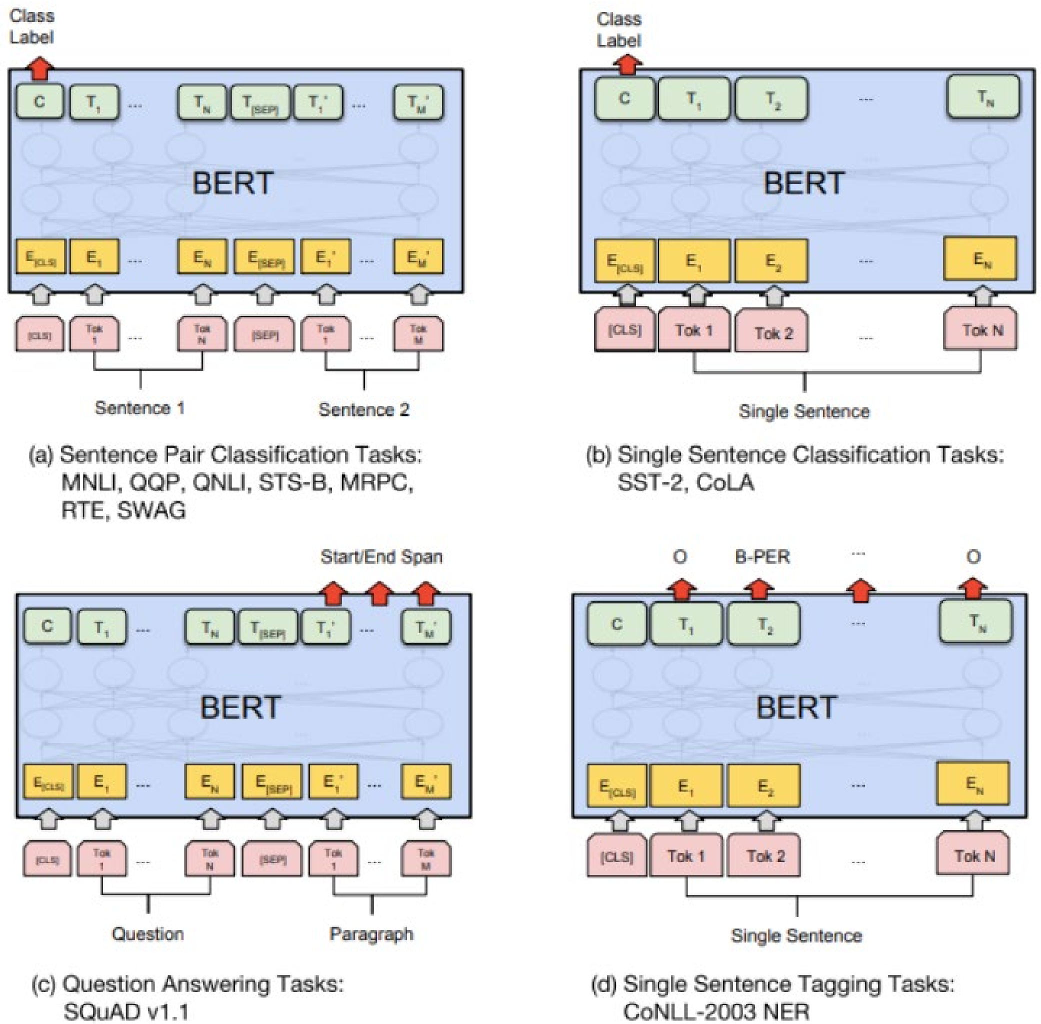


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

Radford, Alec, et al. "*Improving language understanding by generative pre-training*."

BERT for Different Tasks



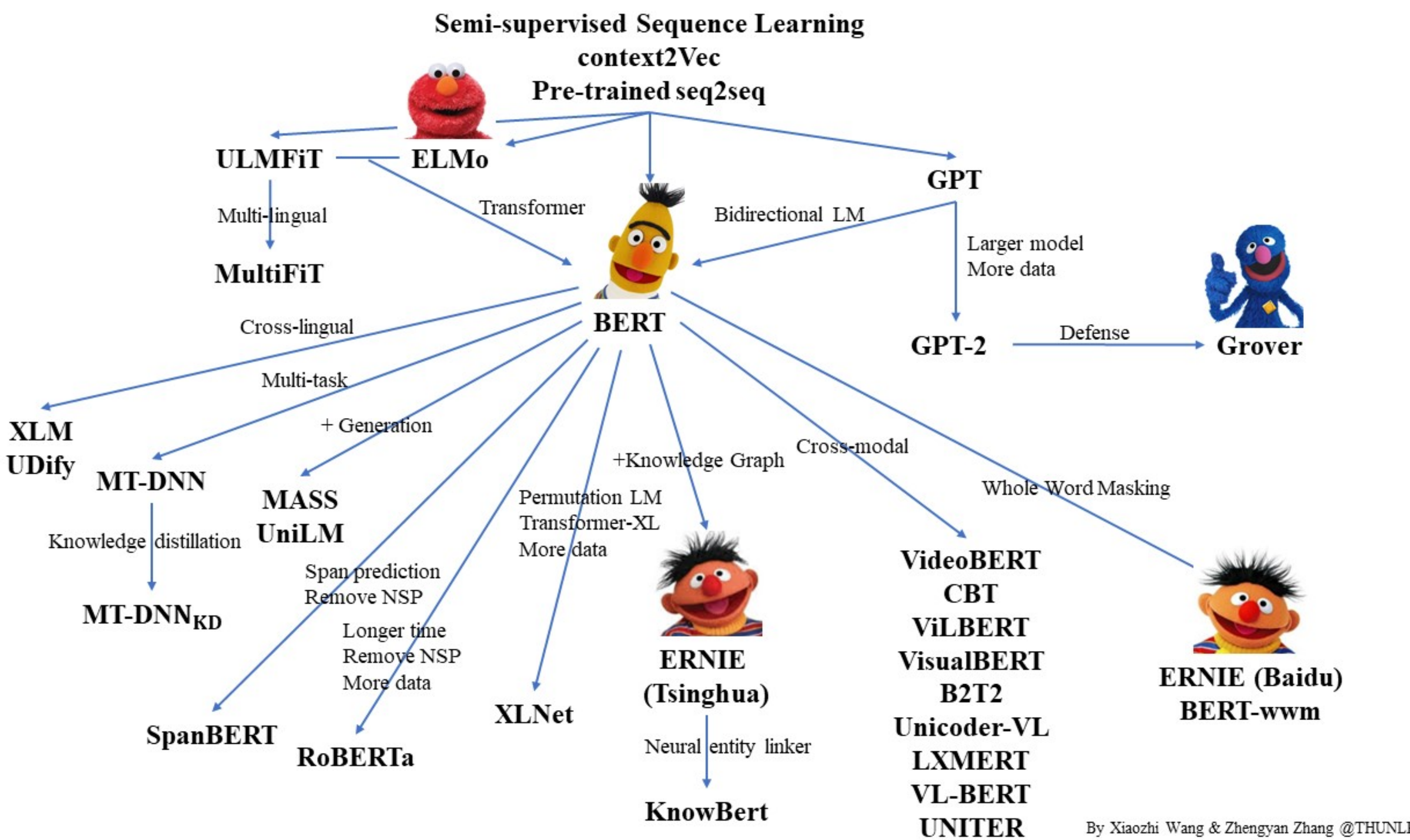
Devlin, Jacob, et al. "*Bert: Pre-training of deep bidirectional transformers for language understanding*."

The path…

- Word2Vec (2013), GloVe (2014) - pre-trained embeddings
  - One vector per word
  - Can capture some syntactic and semantic relations of words
  - Can't handle polysemy
- ELMo - contexulized word embeddings (2018)
  - **Bi-directional** LSTM trained on language modeling
  - Input a sentence to get contextualized embeddings
- Transformer – better handling of long distance dependencies (2017)
- ULM-FiT – fine-tunable language model (May 2018)
- GPT – used for multiple downstream tasks, decoders-only (June 2018)
  - forward
- BERT – to use both left and right context, encoders-only (Nov 2018)
  - masked language model
  - Two-sentence tasks
- GPT-2 – generating convincing, coherent text (Feb 2019)
- XLNet – decoders-only, beating BERT in 20 NLP tasks (June 2019)
- GPT-3 – no fine-tuning for many tasks… (June 2020)

PRE-TRAINED MODELS

A Non-exhaustive Summary



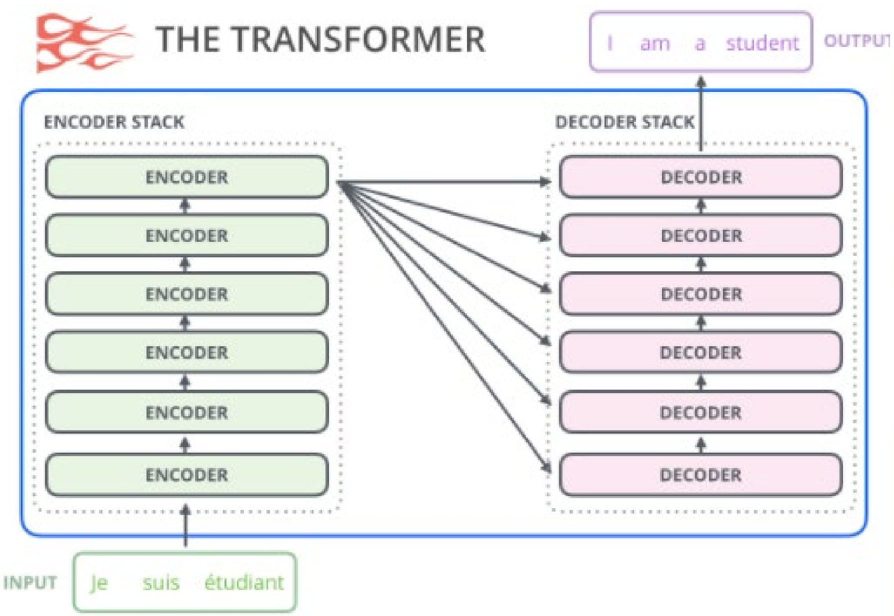
https://github.com/thunlp/PLMpapers

Pre-trained Transformer Models

- **Autoregressive models**
  - Pretrained on the classic language modeling task – predict the next token given the previous ones
  - The **decoders**
  - Versatile, and good for text generation
  - e.g. GPT, GPT-2, GPT-3, Transformer-XL, CTRL, Reformer, XLNet, etc.
- **Autoencoding models**
  - Pretrained by corrupting the input tokens and trying to reconstruct the original sentence
  - The **encoders**
  - Bidirectional representation, versatile, and good for sentence/token classification
  - e.g. BERT, ALBERT, RoBERTa, DistilBERT, XLM, Longformer, etc.
- **Sequence-to-sequence models**
  - Both **encoders and decoders**
  - Good for translation, summarization, and question answering tasks
  - E.g. BART, MariantMT, T5, etc.

The differences…

Sequence-to-sequence model



Autoencoding model

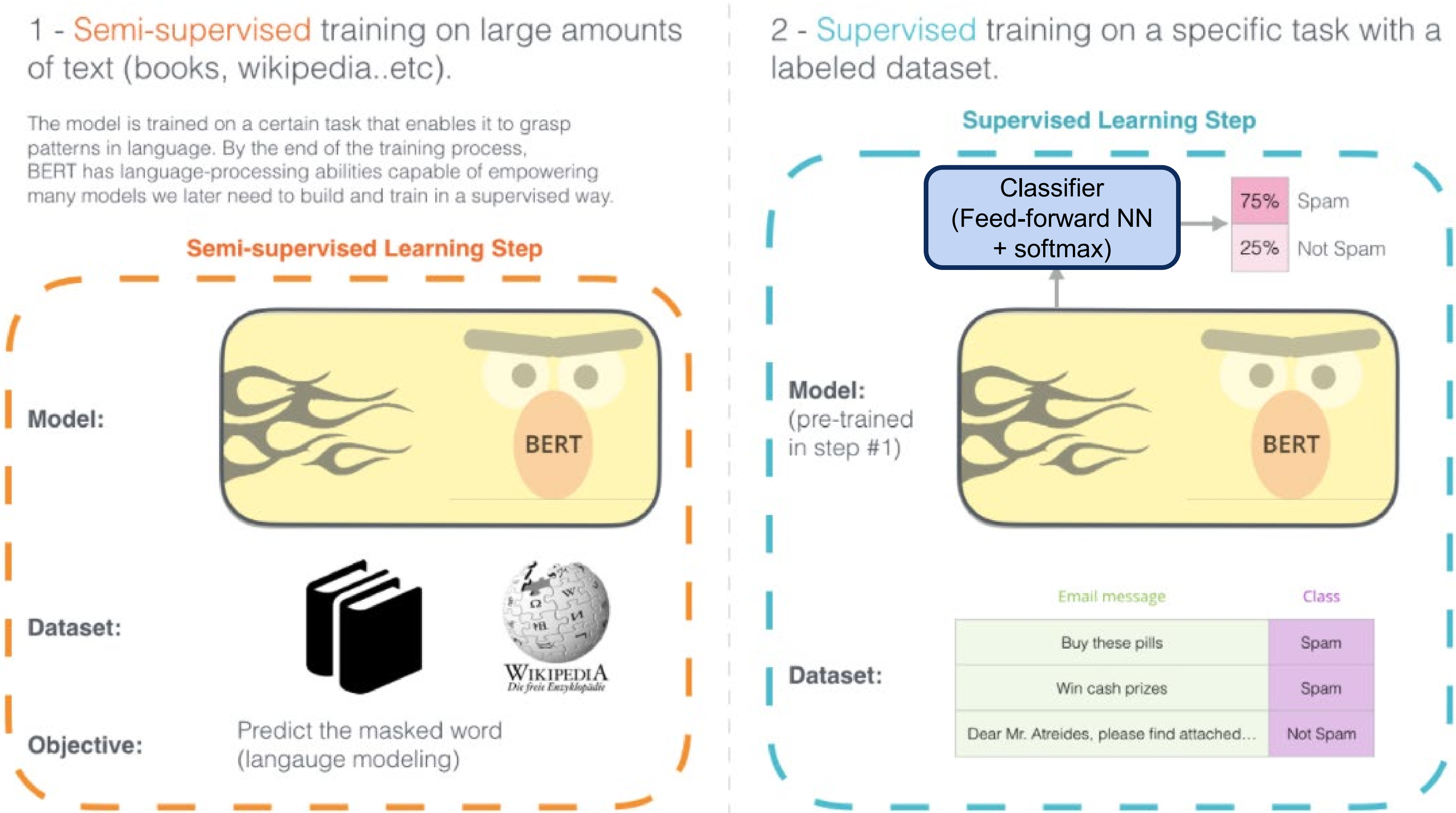
Autoregressive model

Jay Alammar, *The Illustrated GPT-2 (Visualizing Transformer Language Models)*

BERT

- **B**idirectional **E**ncoder **R**epresentations from **T**ransformers
- Released in Nov 2018 by Google, English and Chinese models, and Multilingual BERT (mBERT, 104 languages)
- Pre-trained transformer encoder stacks
- Two sizes
  - BERT Base: 12 encoder layers, 768 hidden units, 12 attention heads, 110M total parameters (for comparison with OpenAI GPT)
  - BERT Large: 24 layers, 1024 hidden units,16 attention heads, 340M total parameters

BERT



The two steps of how BERT is developed. You can download the model pre-trained in step 1 (trained on un-annotated data), and only worry about fine-tuning it for step 2. [Source for book icon].

Jay Alammar, *The Illustrated BERT, ELMo, and co. (How NLP Cracked Transfer Learning)*

BERT vs OpenAI GPT vs ELMo

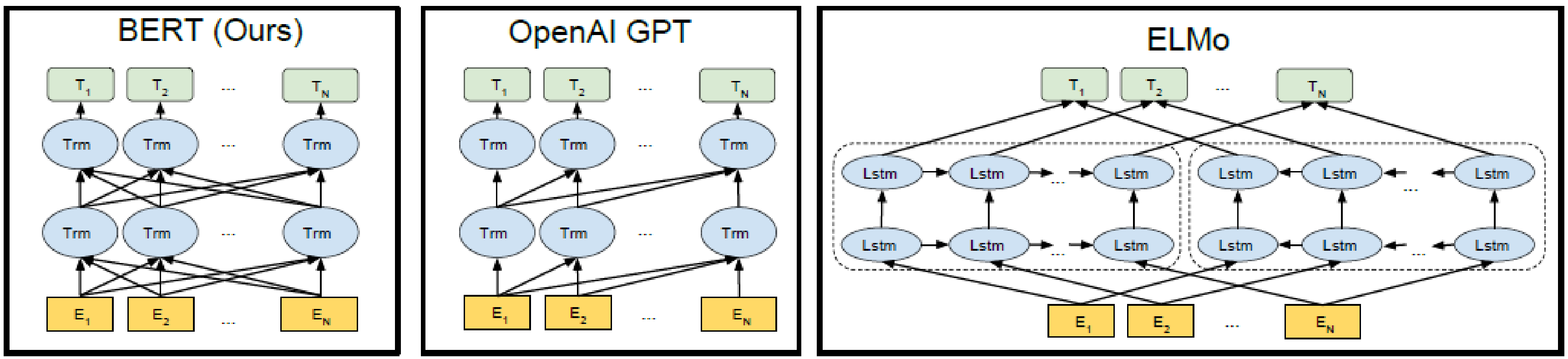


Figure 3: Differences in pre-training model architectures. BERT uses a bidirectional Transformer. OpenAI GPT uses a left-to-right Transformer. ELMo uses the concatenation of independently trained left-to-right and right-to-left LSTMs to generate features for downstream tasks. Among the three, only BERT representations are jointly conditioned on both left and right context in all layers. In addition to the architecture differences, BERT and OpenAI GPT are fine-tuning approaches, while ELMo is a feature-based approach.

Devlin, Jacob, et al. "*Bert: Pre-training of deep bidirectional transformers for language understanding*."

Pre-train BERT

- Task #1: Masked Language Model (MLM)
  - Randomly mask some input tokens, then predict the masked tokens (*Cloze* task)
  - Mask 15% of all WordPiece tokens in each sequence
  - the final hidden vectors corresponding to the mask tokens are fed into an output softmax over the vocabulary
- Task #2: Next Sentence Prediction (NSP)
  - Given two sentences A and B, predict whether B is the next sentence of A (*IsNext/NotNext*)
  - Useful for QA and NLI

Pre-training BERT

- Pre-trained using the BooksCorpus (800M words) (Zhu et al., 2015) and English Wikipedia (2,500M words) -- 3.3 billion words in total.
- Batch size: 256 sequences (256 sequences * 512 tokens = 128,000 tokens/batch)
- 1M steps, 40 epochs
- Took 4 days to train each model
  - base model: 4 Cloud TPUs in Pod configuration (16 TPU chips total)
  - Large model: 16 Cloud TPUs (64 TPU chips total)
- Other details available in the original paper

BERT Input Representation

My dog is cute. He likes playing.

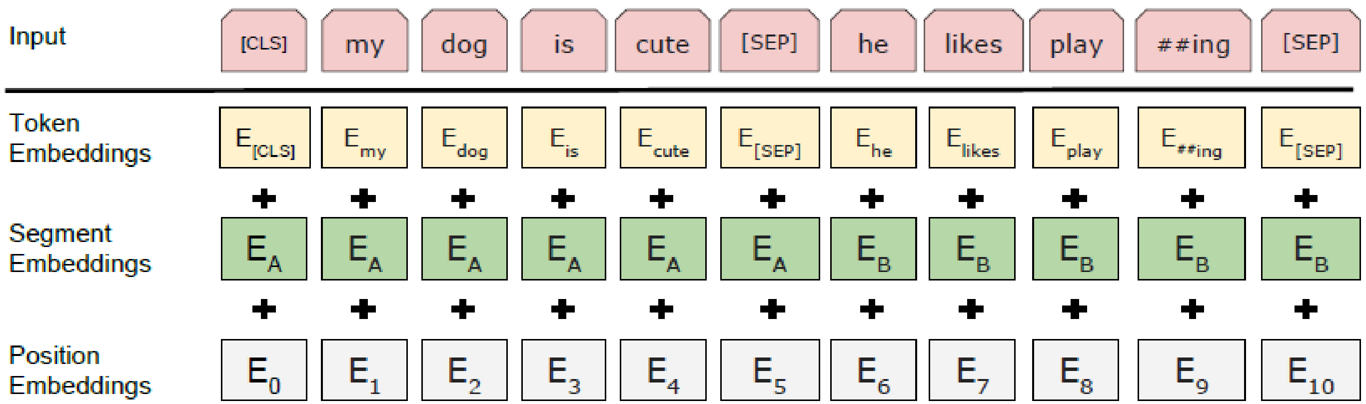


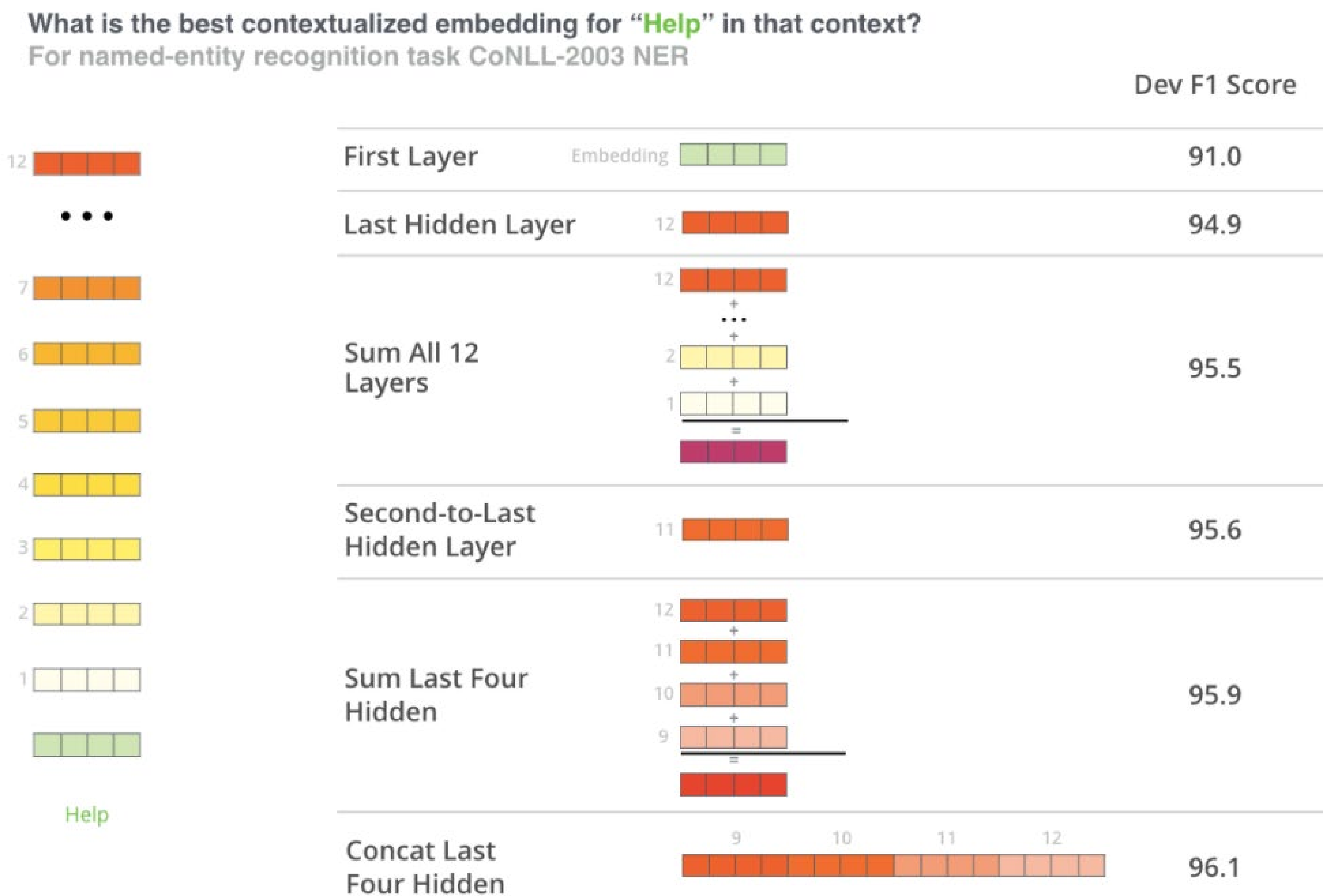
Figure 2: BERT input representation. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

Devlin, Jacob, et al. "*Bert: Pre-training of deep bidirectional transformers for language understanding*."

WordPiece Model

- Vocabulary: ~30,000 most common words or subwords from the training corpora
  - Whole words
  - Subwords occurring alone or at the beginning of words
  - Subwords occurring not at the beginning of words, preceded by “##” (e.g. *##ing*)
  - Individual characters
- What happens to unknown words or out-of-vocabulary words?
  - Split into subword tokens or individual characters by BERT’s Tokenizer (WordPiece model)

BERT for Feature Extraction



Jay Alammar, *The Illustrated BERT, ELMo, and co. (How NLP Cracked Transfer Learning)*

Fine-tuning BERT

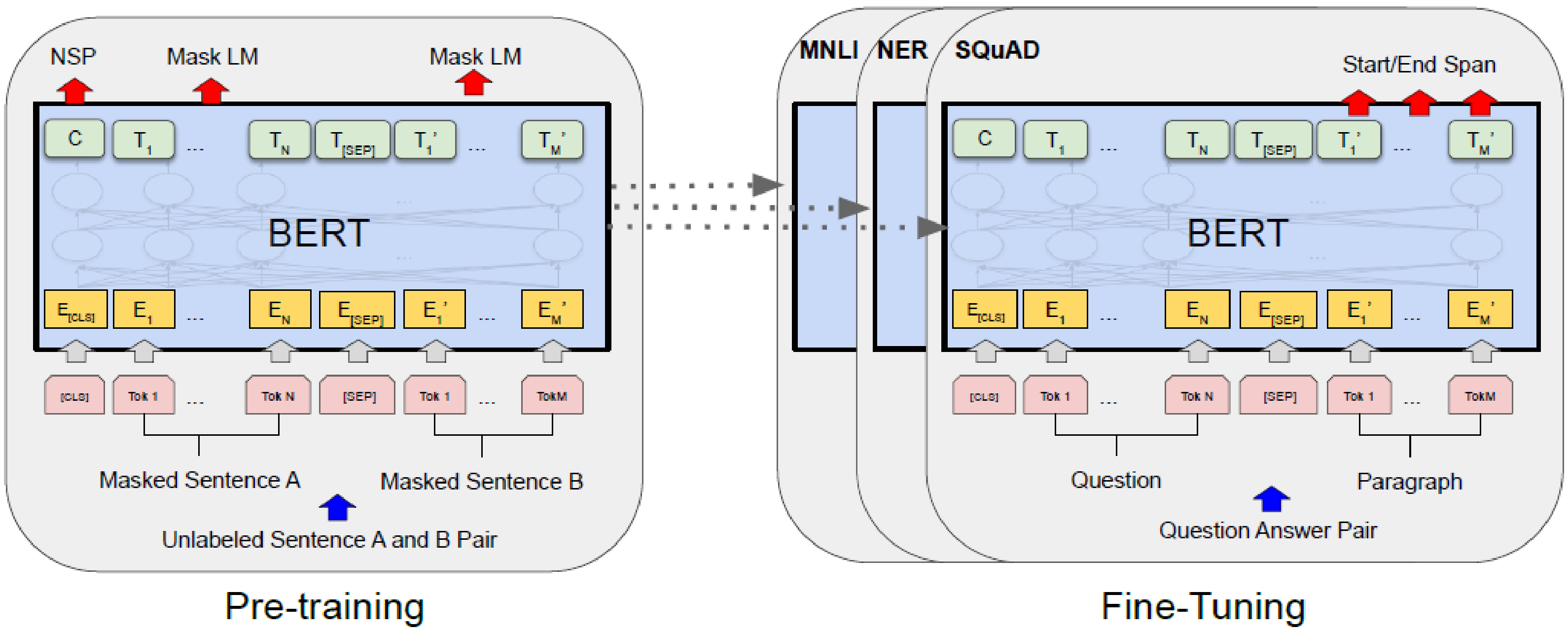


Figure 1: Overall pre-training and fine-tuning procedures for BERT. Apart from output layers, the same architectures are used in both pre-training and fine-tuning. The same pre-trained model parameters are used to initialize models for different down-stream tasks. During fine-tuning, all parameters are fine-tuned. [CLS] is a special symbol added in front of every input example, and [SEP] is a special separator token (e.g. separating questions/answers).

Devlin, Jacob, et al. "*Bert: Pre-training of deep bidirectional transformers for language understanding*."

Fine-tuning BERT

- Many experiments on various tasks
- 100K+ examples ~large
- Some suggested ranges of hyperparameters:
  - Batch size; 16, 32
  - Learning rate (Adam): 5e-5, 3e-5, 2e-5
  - Number of epochs: 2, 3, 4
- Training time: 1 hour on a single Cloud TPU, or a few hours on a GPU

Performance of BERT

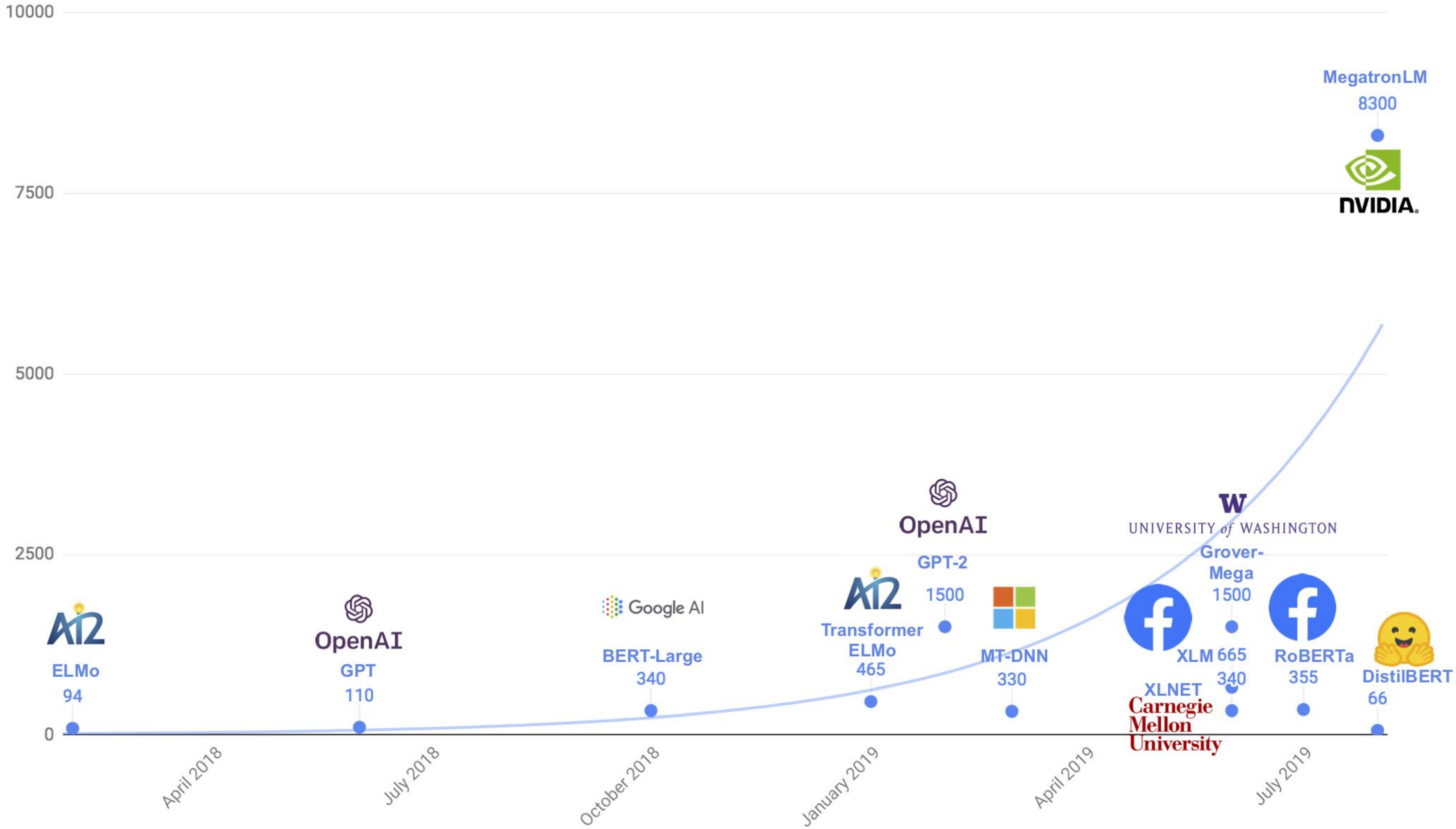
- "*It obtains new state-of-the-art results on eleven natural language processing tasks (using fine-tuning approach)*", including
  - pushing the GLUE score to 80.5% (7.7% point absolute improvement),
  - MultiNLI accuracy to 86.7% (4.6% absolute improvement),
  - SQuAD v1.1 question answering Test F1 to 93.2 (1.5 point absolute improvement)
  - and SQuAD v2.0 Test F1 to 83.1 (5.1 point absolute improvement).

GLUE Tasks

The General Language Understanding Evaluation benchmark (GLUE) is a collection of datasets used for training, evaluating, and analyzing NLP models, with 9 task datasets (sentence or sentence-pair)

- **CoLA**: grammatical acceptability
- **SST-2** (Stanford Sentiment Treebank): sentiment classification
- **MRPC** (The Microsoft Research Paraphrase Corpus): Given pair of sentences, classify them as paraphrases or not paraphrases
- **STS-B**: The Semantic Textual Similarity Benchmark (Cer et al., 2017)
- **QQP**: The Quora Question Pairs3 dataset, question pairs from Quora, to determine whether the questions in a pair are semantically equivalent.
- **MNLI-mm**: The Multi-Genre Natural Language Inference Corpus (Williams et al., 2018), sentence pairs with textual entailment annotations. Given a premise sentence and a hypothesis sentence, predict whether the premise entails the hypothesis, contradicts the hypothesis, or neither (neutral).
- **QNLI**: The Stanford Question Answering Dataset (Rajpurkar et al. 2016; SQuAD), a question-answering dataset, question-paragraph pairs, where the one of the sentences in the paragraph contains the answer.
- **RTE**: The Recognizing Textual Entailment (RTE) datasets, to predict if the premise entails the hypothesis.
- **WNLI**: The Winograd Schema Challenge: read a sentence with a pronoun, decide the referent from a list of choices (Coreference)

Large, XL, Mega…



Victor Sanh, "*Smaller, faster, cheaper, lighter: Introducing DistilBERT, a distilled version of BERT*"

Large-Scale Language Models

- Challenges: how to use them in production, under low latency constraints?
- How to reduce size of such models?
  - Quantization: approximating the weights of a network with a smaller precision
  - Weights pruning: removing some connections in the network
  - Knowledge distillation: a compact model - the student - is trained to reproduce the behaviour of a larger model - the teacher - or an ensemble of models (**teacher-student learning**)
    - E.g. DistilBERT, ALBERT, TinyBERT, etc.

Knolwedge Distillation

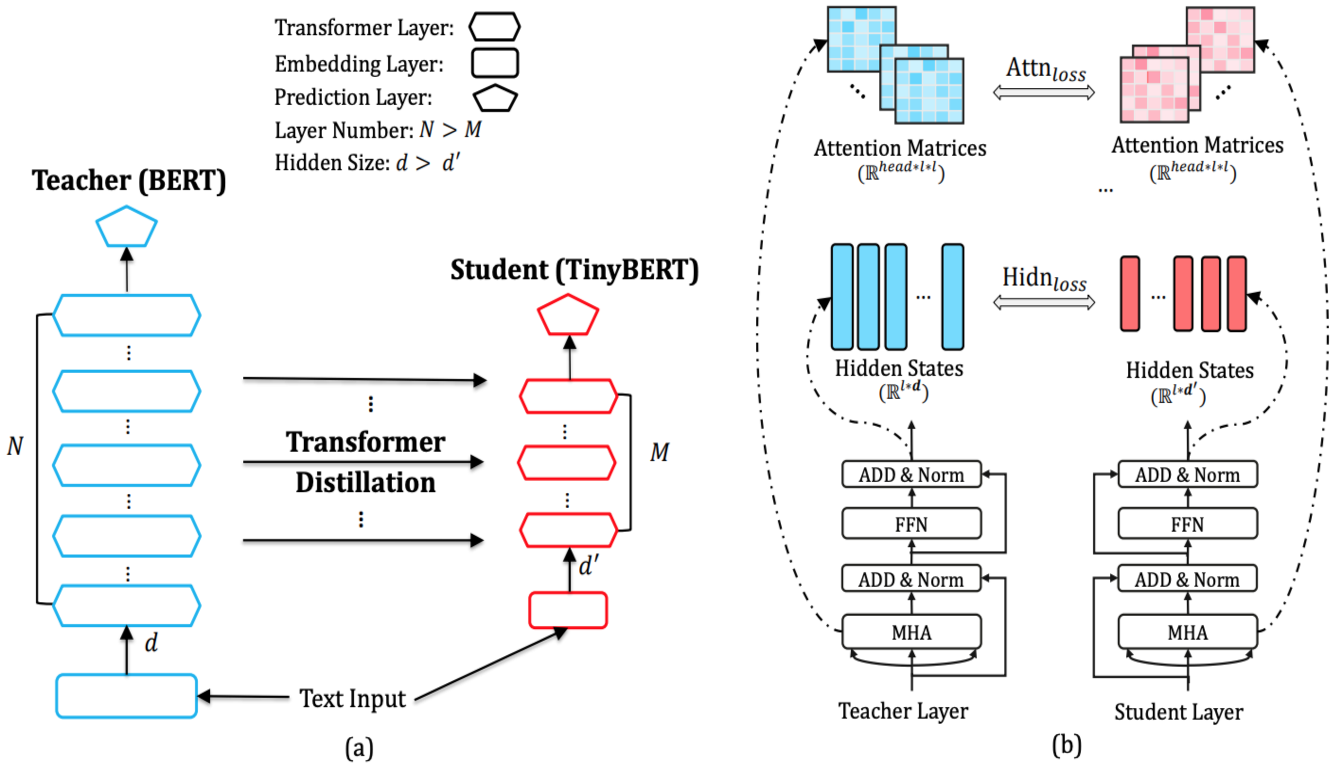


Figure 1: An overview of Transformer distillation: (a) the framework of Transformer distillation, (b) the details of Transformer-layer distillation consisting of Attn_{loss}(attention based distillation) and Hidn_{loss}(hidden states based distillation).

Smaller, faster, lighter

- DistilBERT (HuggingFace)
  - reduce the size of a BERT base model by 40%
  - retaining 97% of its language understanding capabilities
  - being 60% faster
- TinyBERT (Huawei)
  - 7.5x smaller than BERT
  - 9.4x faster
  - Comparable results on GLUE benchmark

Table 3: The model sizes and inference time for baselines and TinyBERT. The number of layers does not include the embedding and prediction layers.

System	Layers	Hidden Size	Feed-forward Size	Model Size	Inference Time
$\text{BERT}_{\text{BASE}}$ (Teacher)	12	768	3072	109M($\times$1.0)	188s($\times$1.0)
Distilled $\text{BiLSTM}_{\text{SOFT}}$	1	300	400	10.1M($\times$10.8)	24.8s($\times$7.6)
BERT-PKD/DistilBERT	4	768	3072	52.2M($\times$2.1)	63.7s($\times$3.0)
TinyBERT	4	312	1200	14.5M($\times$7.5)	19.9s($\times$9.4)

Other BERT-related Models

- XLM-RoBERTa (XLM-R) by Facebook
  - by Facebook
  - 15 languages
  - MLM, MLM + translation language modelling
- DistilmBERT by HuggingFace
  - "*reaches 92% of Multilingual BERT's performance… while being twice faster and 25% smaller*"
- Other language-specific BERTs
  - German, French, Italian, Spanish, Finnish, Portugese, Russian, Japanese, etc.

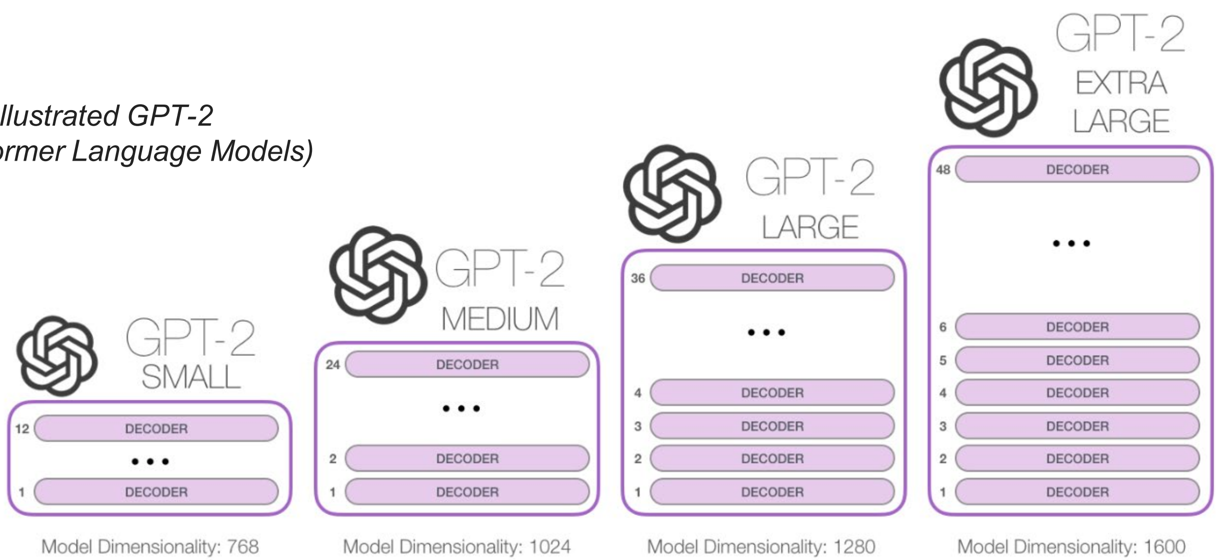
GPT-2

- By Open AI, which was founded in San Francisco in late 2015 by Elon Musk, Sam Altman, and others
- A large transformer-based language model (Generative Pre-Training) with **1.5 billion** parameters
- Trained on WebText, a dataset of 8 million web pages with diversified topics
- Task in training: to predict the next word given the previous words in context.
- Broad set of capabilities without using task specific training data – question answering, reading comprehension, summarization, and translation

Controlled release of models

- To avoid malicious usage of the full model, e.g.
  - Generate misleading news articles
  - Impersonate others online
  - Automate the production of abusive or faked content to post on social media
  - Automate the production of spam/phishing content
- Staged releases: 124M, 355M, 774M, 1.5B (from Feb to Nov 2019)

Jay Alammar, *The Illustrated GPT-2 (Visualizing Transformer Language Models)*



Demo of GPT-2

Write With Transformer `gpt2`

Shuffle initial text   Trigger autocomplete or `tab`   Select suggestion `↑` `↓` and `enter`

Cancel suggestion `esc`

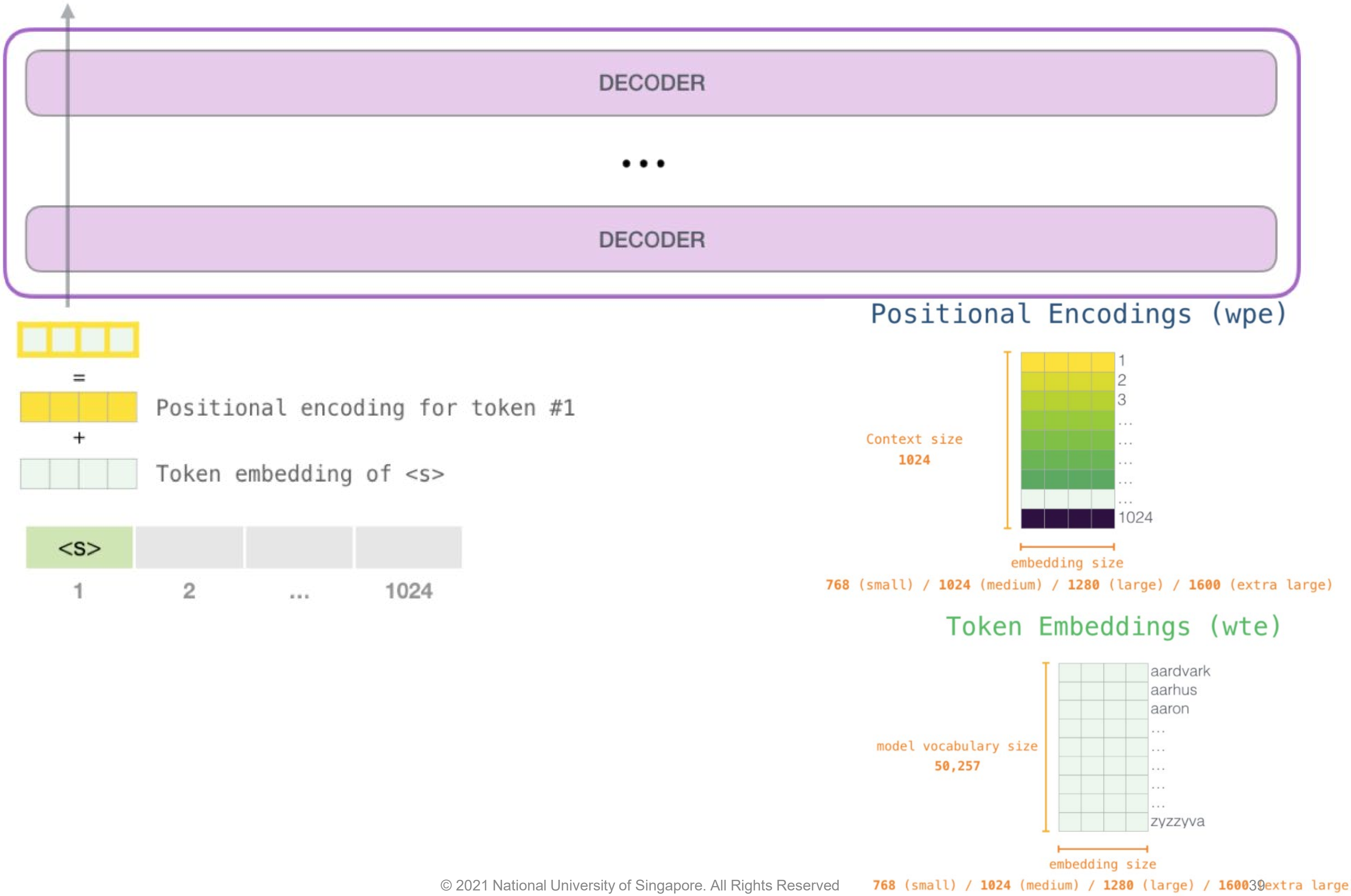
Save & Publish

Before boarding your rocket to Mars, remember to pack these items to ensure that you have the necessary supplies in your vehicle. The following are recommended for Mars:

A space suit, including one that can support your weight. Food for the entire month you're staying there and provisions for the entire journey.

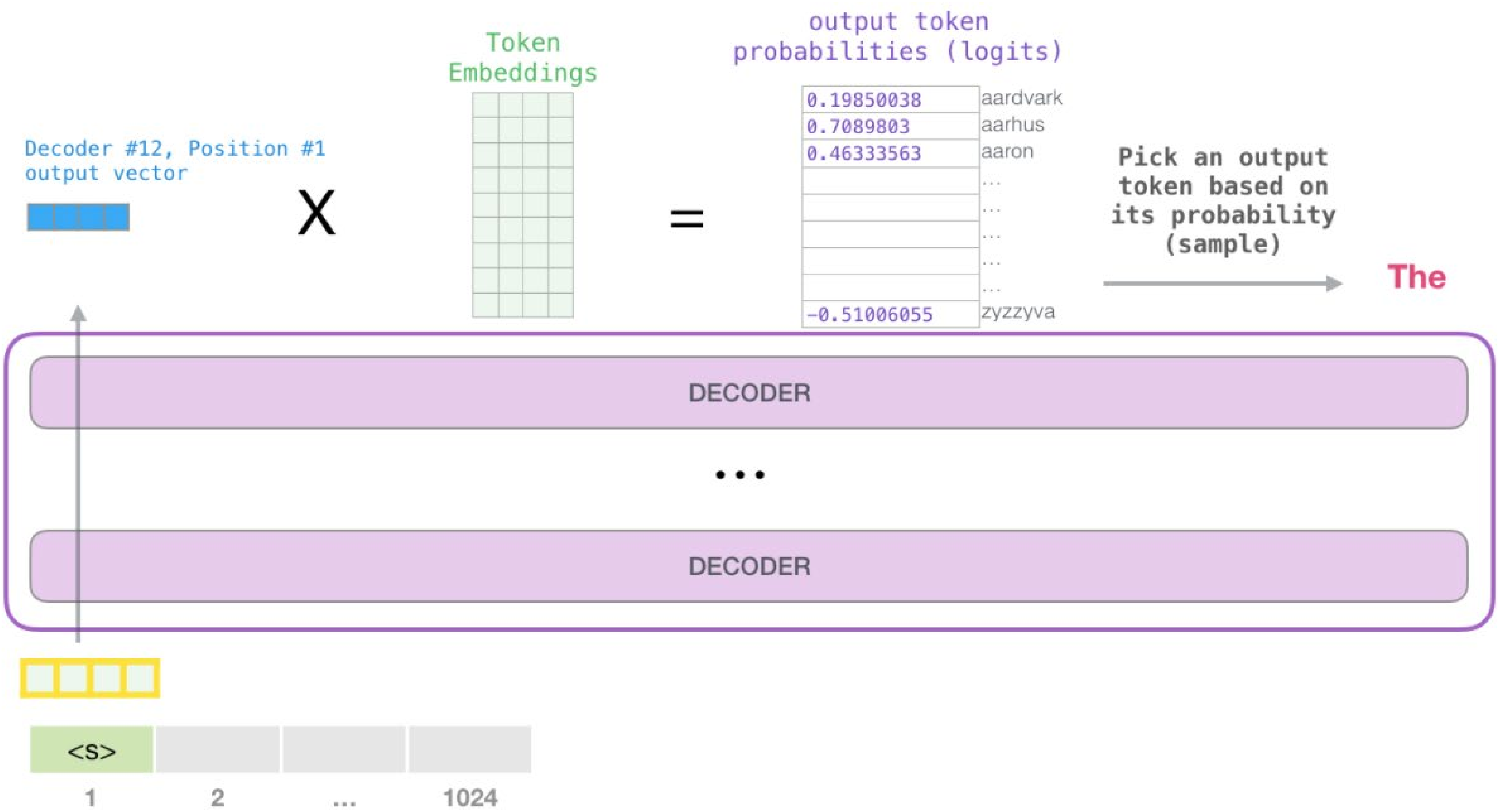
Share screenshot

Input



DECODER
DECODER
Positional encoding for token #1
Token embedding of <s>
<S>
1
2
...
1024
Positional Encodings (wpe)
Context size
1024
1
2
3
1024
embedding size
768 (small) / 1024 (medium) / 1280 (large) / 1600 (extra large)
Token Embeddings (wte)
model vocabulary size
50,257
aardvark
aarhus
aaron
zyzzyva
embedding size
768 (small) / 1024 (medium) / 1280 (large) / 1600 (extra large)

Output


Decoder #12, Position #1
output vector
X
Token
Embeddings
=
output token
probabilities (logits)
0.19850038 aardvark
0.7089803 aarhus
0.46333563 aaron
...
-0.51006055 zyzzyva
Pick an output
token based on
its probability
(sample)
The
DECODER
DECODER
<s>
1
2
...
1024

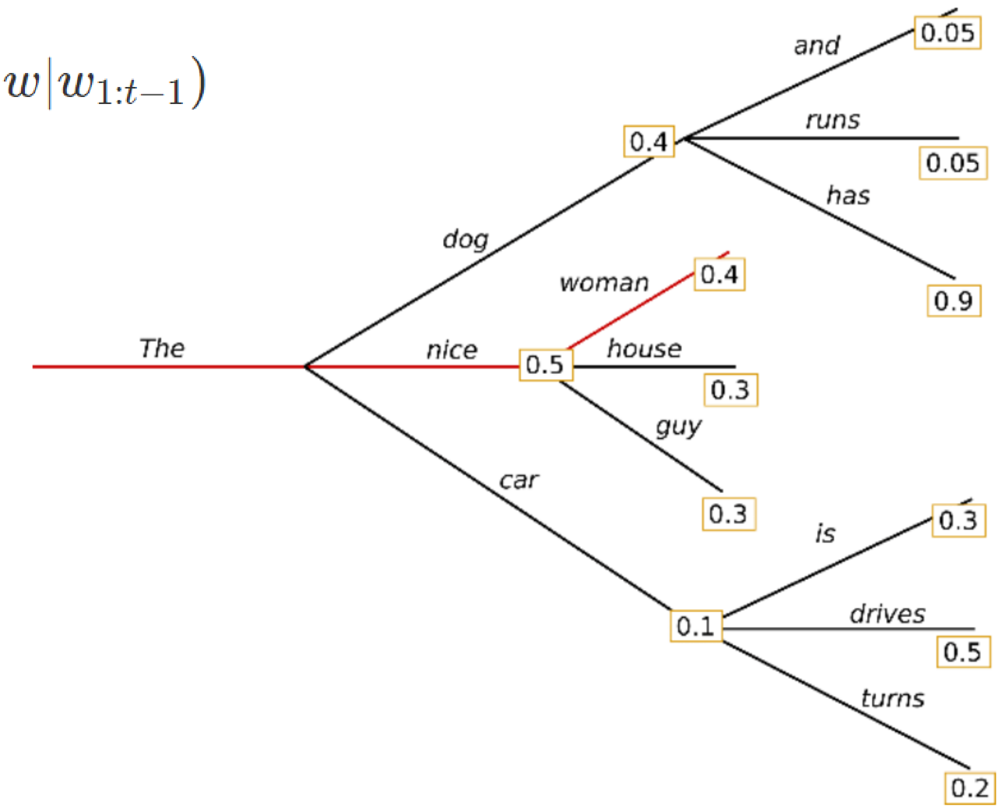
Different Decoding Methods

- Select the output token based on output logits(token probabilities)
- The most common methods:
  - Greedy search
  - Beam search
  - Top-k sampling
  - Top-p sampling

Greedy Search

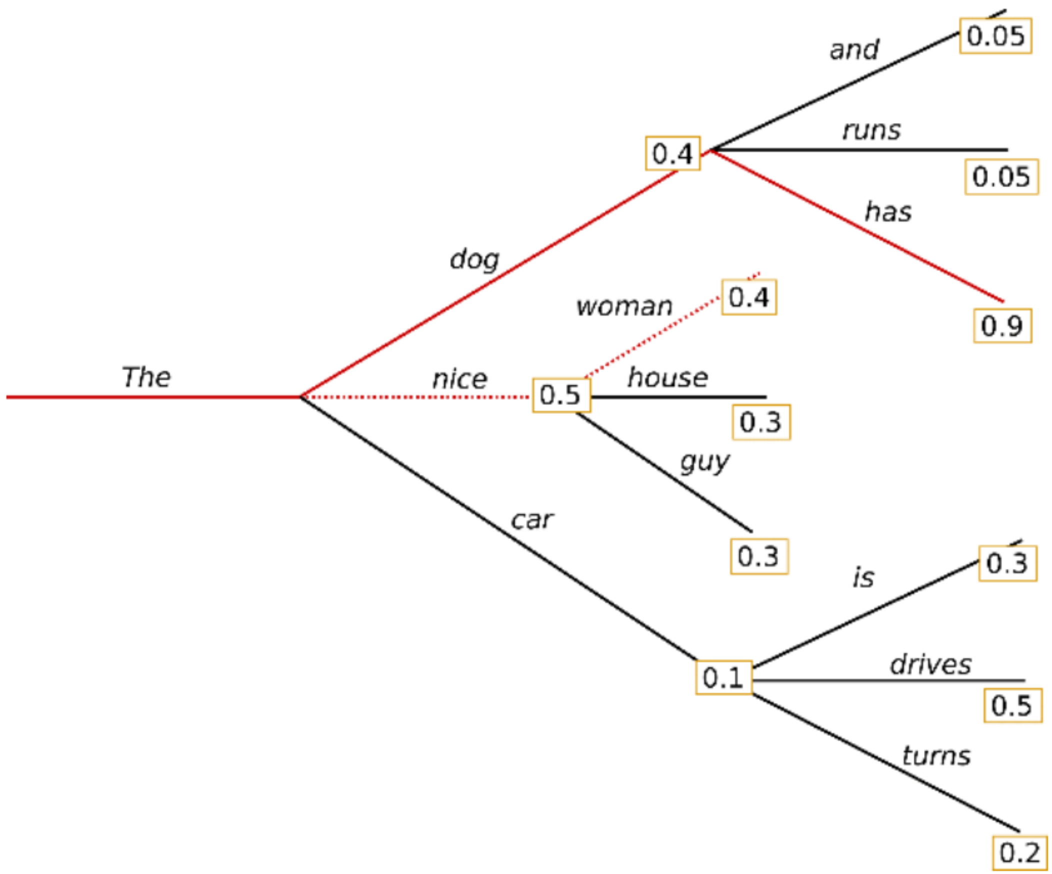
- Select the token with the highest probability
- May miss out high probability words hidden behind a low probability word

$$w_t = argmax_w P(w|w_{1:t-1})$$



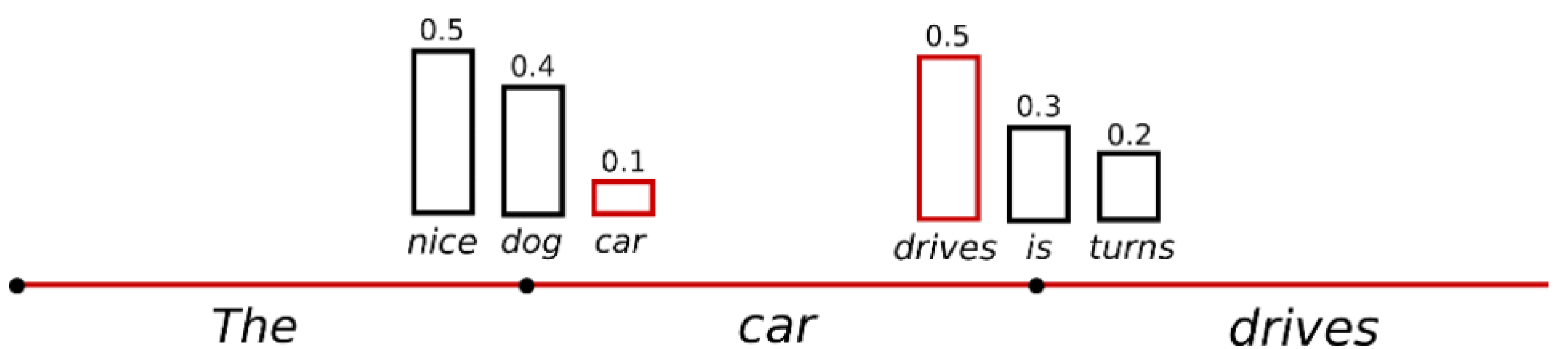
Beam Search

- Keep a number of most likely hypotheses at each time step
- Eventually choose the hypothesis that has the overall highest probability



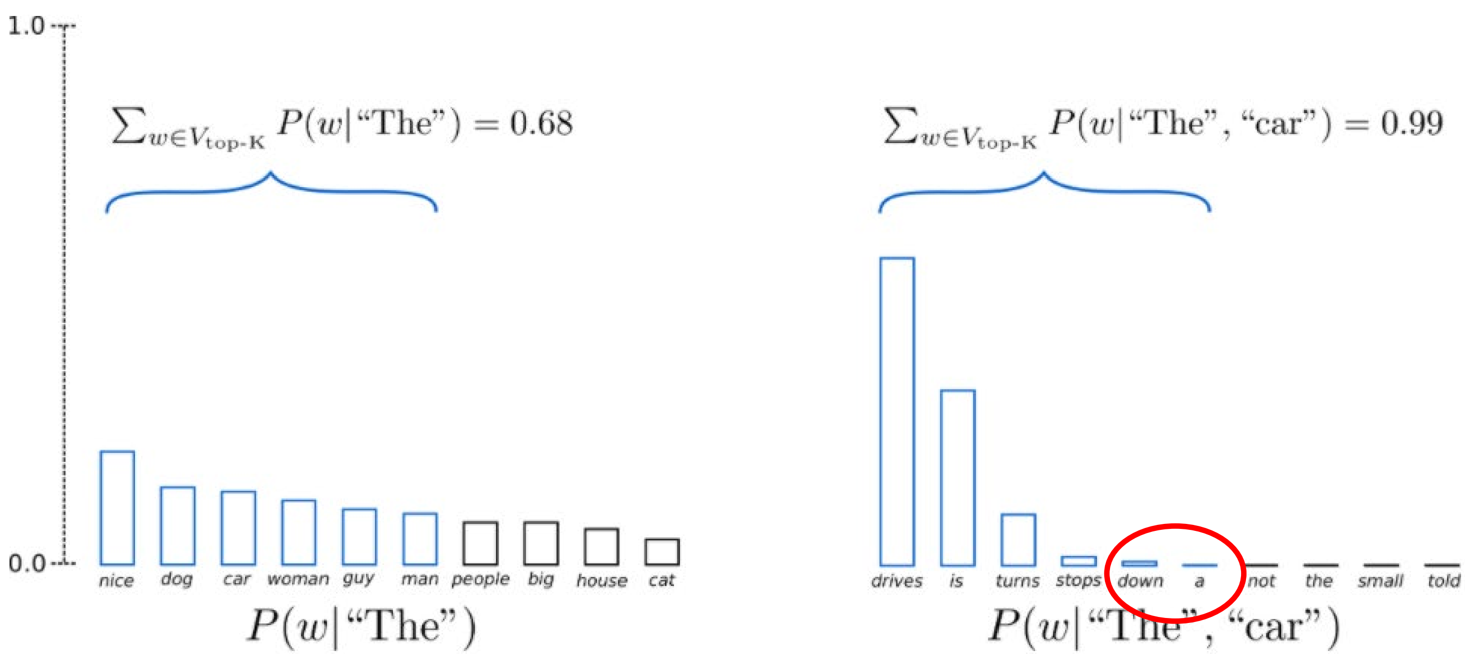
Sampling

- Problems of beam search
  - Suffer from repetitive generation (solution: *n-gram-penalty*)
  - High probability words can be boring/predictable
- Introduce randomness using Sampling!
  - Randomly pick the next word based on its conditional probability.
  - Cons: generated text is not coherent.



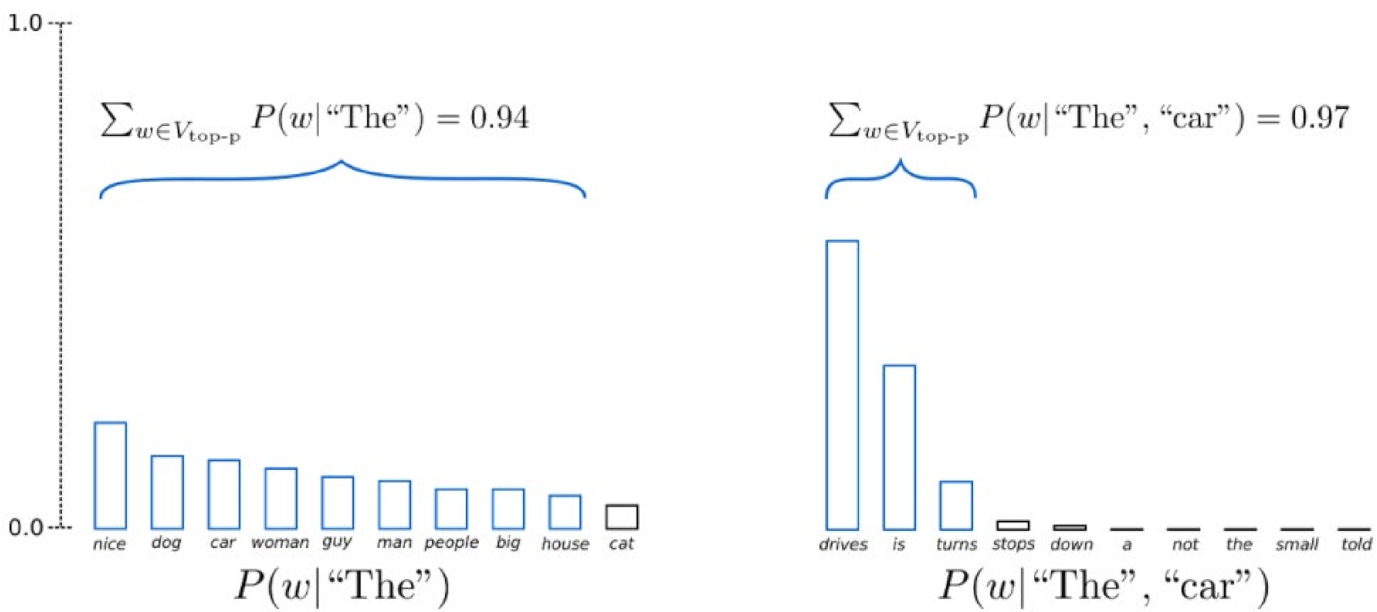
Top-K Sampling

- The K most likely next words are filtered and the probability mass is redistributed among only those K next words.
- Used by GPT-2



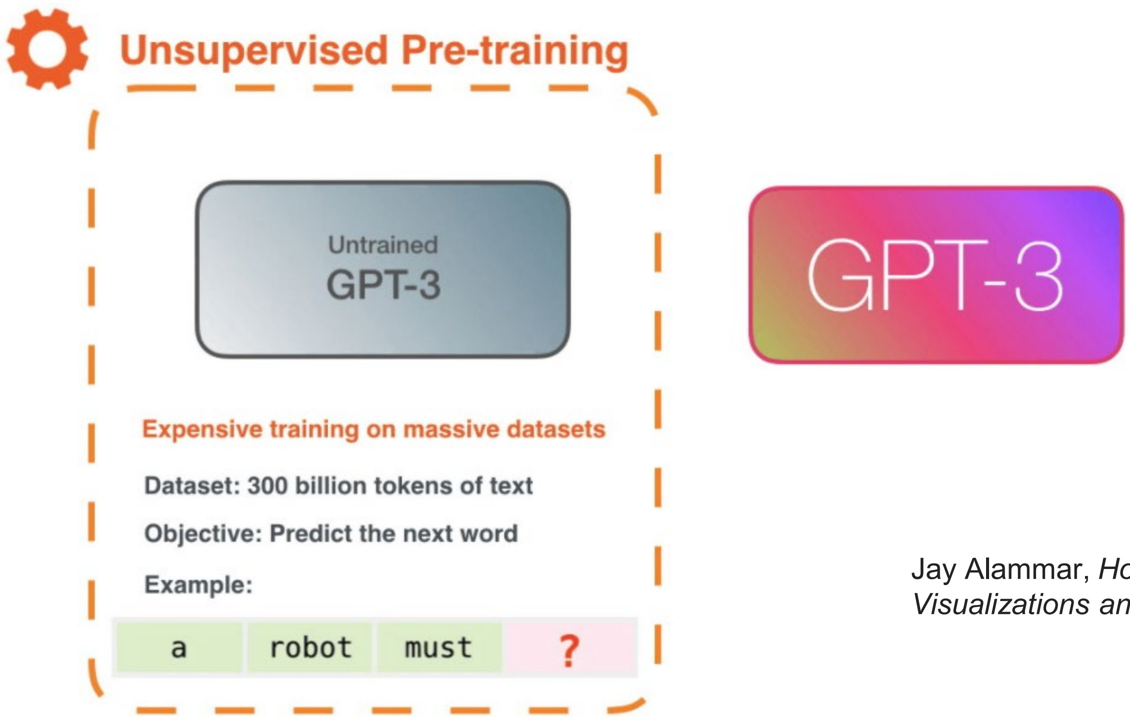
Top-p Sampling

- Also known as *nucleus sampling*
- Choose from the smallest possible set of words whose cumulative probability exceeds the probability *p*.
- Leave out words with very low probabilities.



GPT-3

- The largest, most powerful language model ever, with **175 billion** parameters



Jay Alammar, *How GPT3 Works – Visualizations and Animations*

GPT-3

- Meta-learning focusing on "<u>task-agnostic</u>" performance:
  - the model develops a broad set of skills and pattern recognition abilities at training time,
  - and then uses those abilities at inference time to rapidly adapt to or recognize the desired task ("in-context learning")
- Can be applied without gradient updates
  - Zero-shot setting
  - One-shot setting
  - Few-shot setting (10~100)

No fine-tuning!

Language Model Meta-Learning

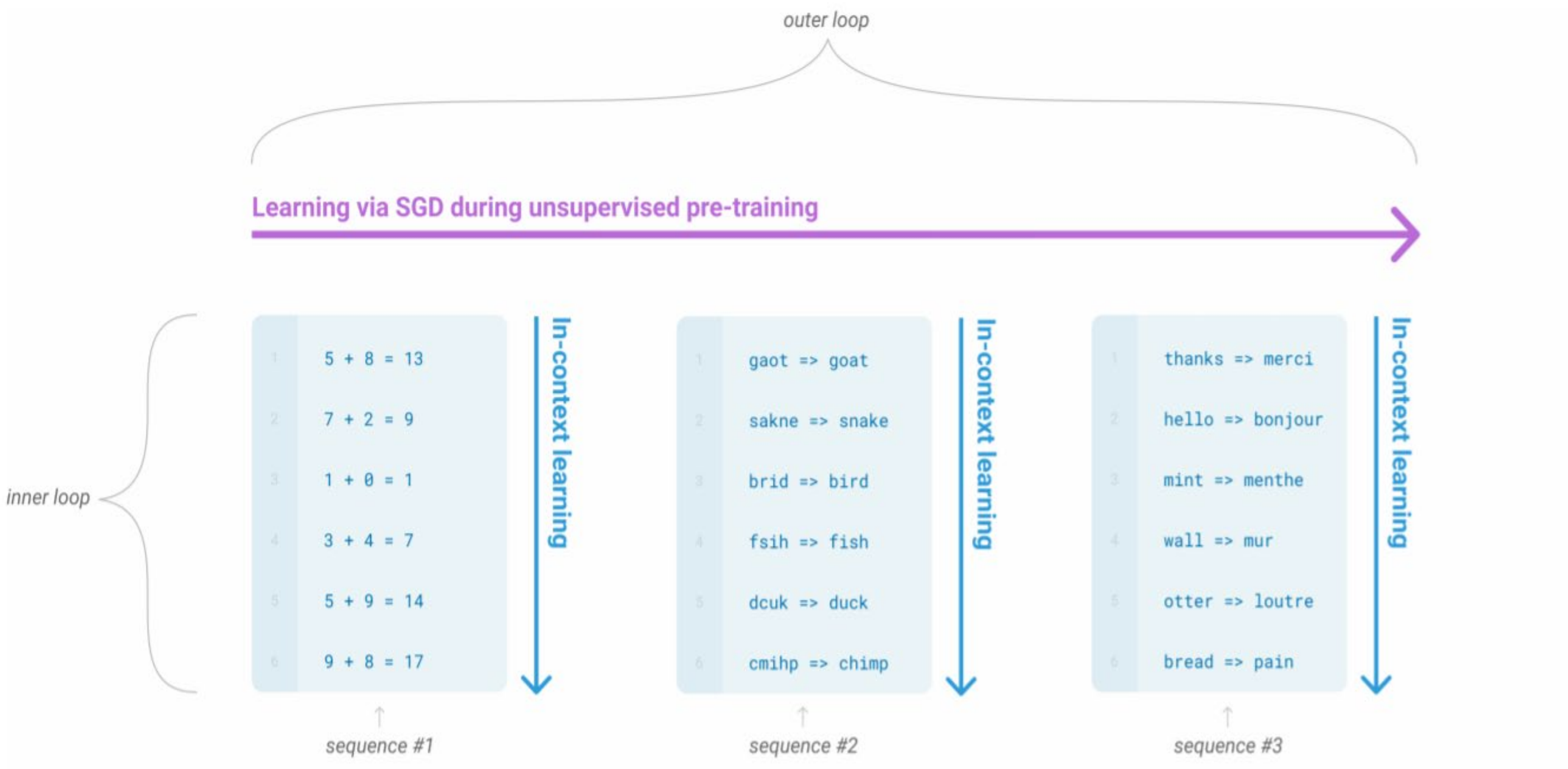


Figure 1.1: Language model meta-learning.

Examples of 3 Settings

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

1	Translate English to French:	task description
2	cheese =>	prompt

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

1	Translate English to French:	task description
2	sea otter => loutre de mer	example
3	cheese =>	prompt

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

1	Translate English to French:	task description
2	sea otter => loutre de mer	examples
3	peppermint => menthe poivrée	
4	plush girafe => girafe peluche	
5	cheese =>	prompt

No fine-tuning!

Training Data for GPT-3

- Mainly Common Crawl (45TB of compressed plaintext before filtering, and 570GB after filtering)
- Enhanced with a few high-quality corpora

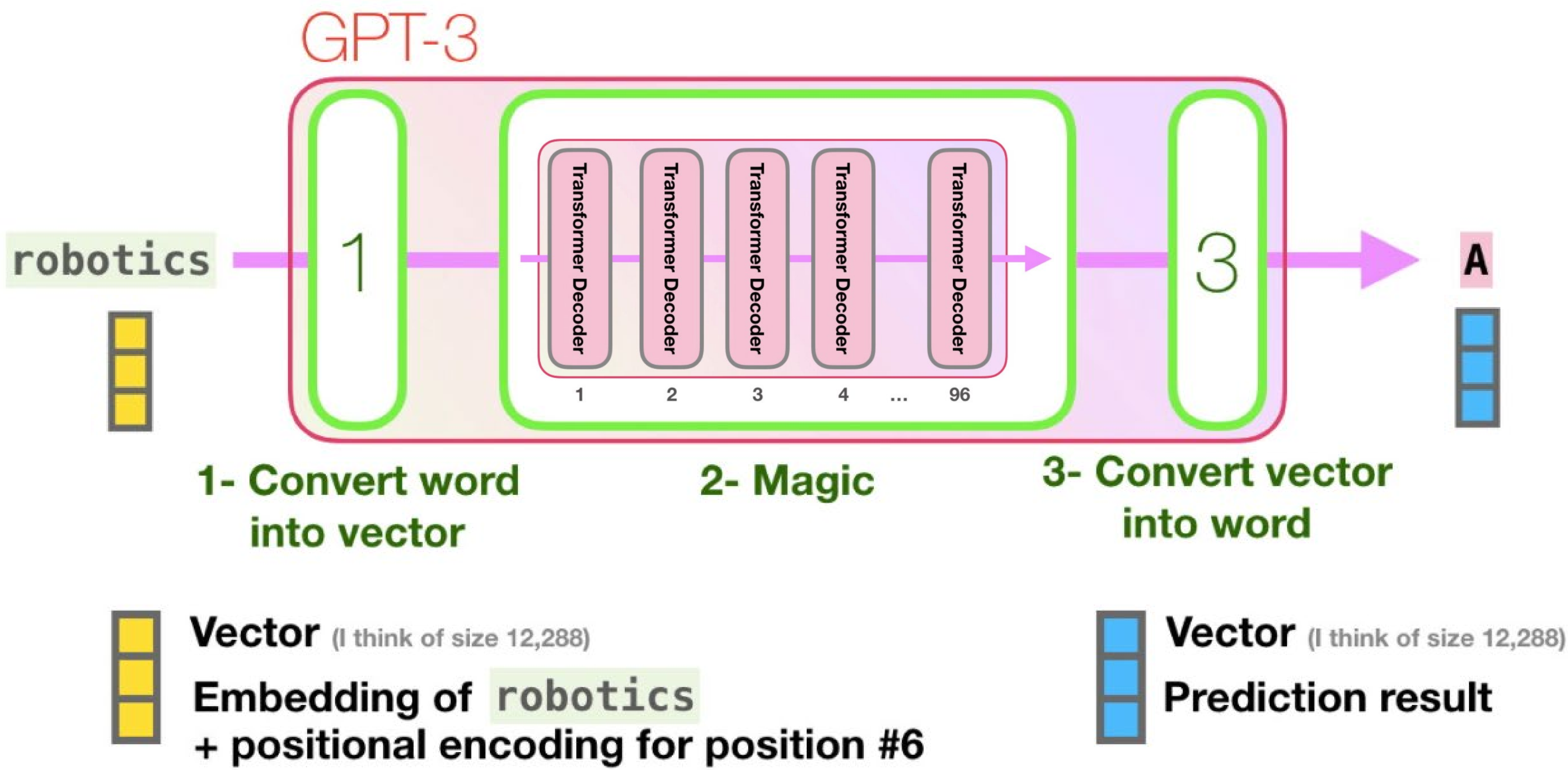
Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

Table 2.2: Datasets used to train GPT-3.

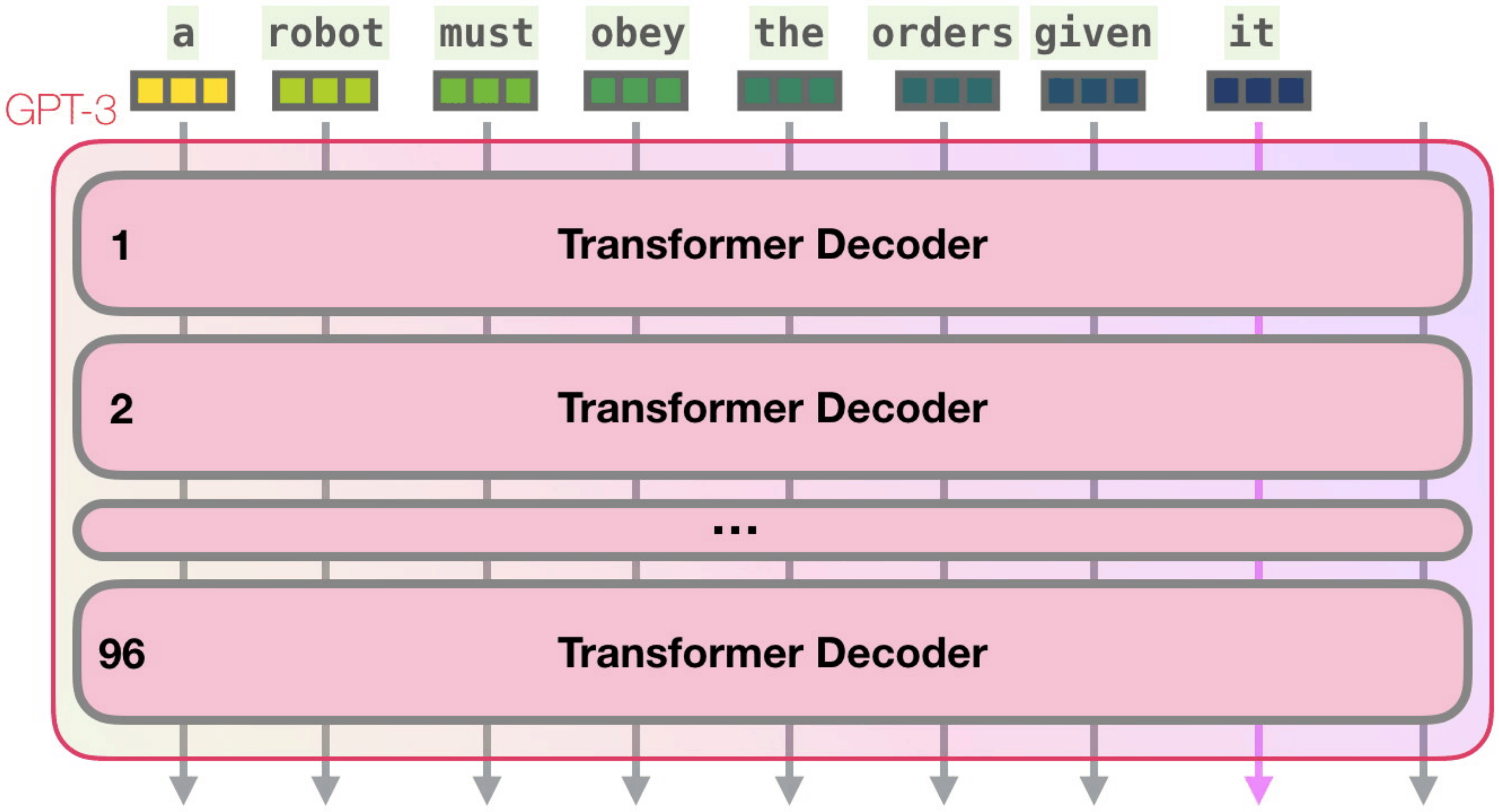
How big is GPT-3?

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}

How it works



Jay Alammar, *How GPT3 Works – Visualizations and Animations*


a
robot
must
obey
the
orders
given
it
GPT-3
1
Transformer Decoder
2
Transformer Decoder
...
96
Transformer Decoder

Better In-Context Learning with Larger Model

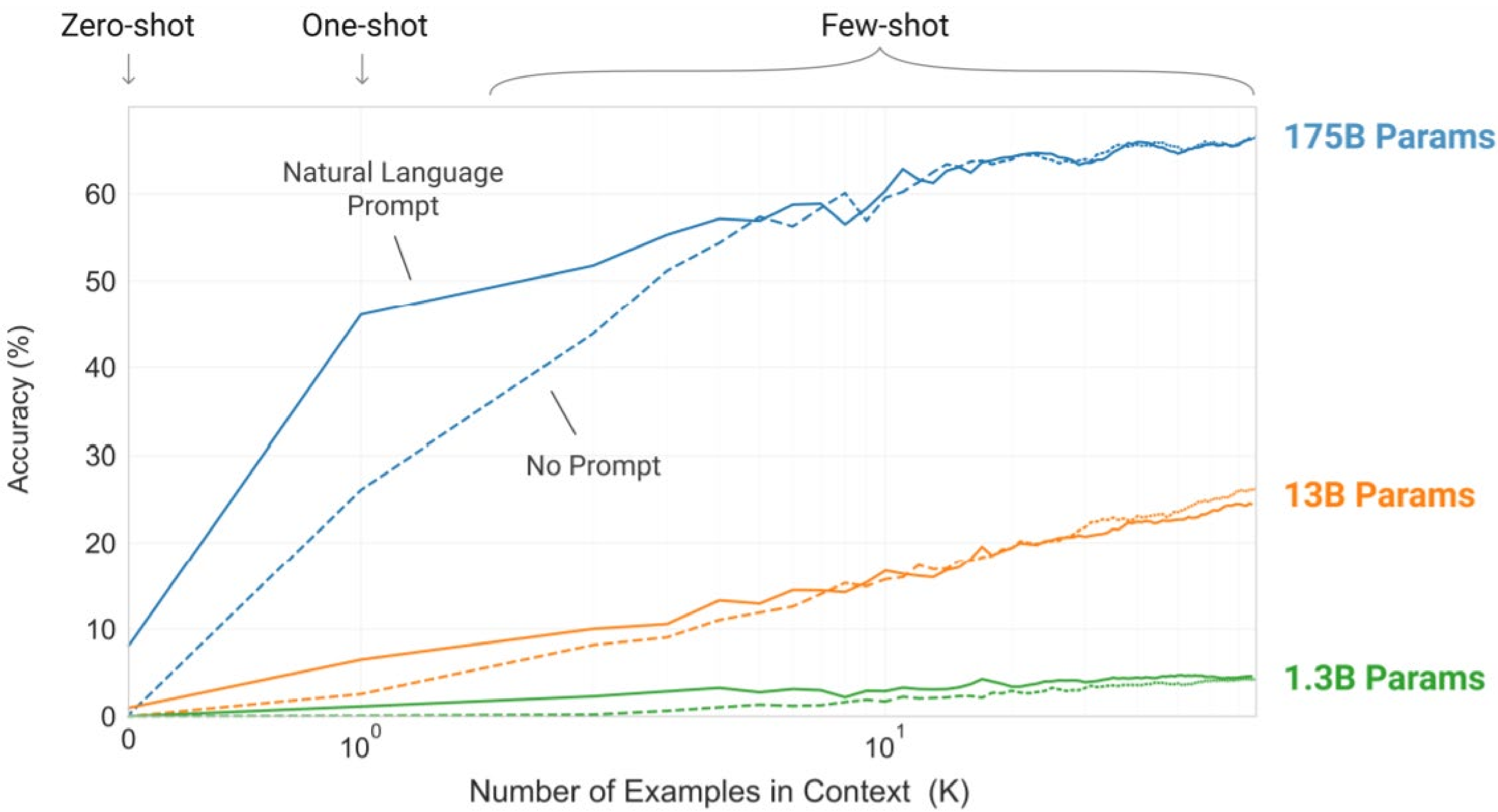


Figure 1.2: Larger models make increasingly efficient use of in-context information.

GPT-3 Performances on Benchmark Tasks

- Evaluated on **benchmark datasets** for various tasks, such as cloze and completion, question answering, machine translation, anaphora resolution, reading comprehension, common sense reasoning, language understanding, natural language inference, synthetic and qualitative tasks, etc.
- Variable results based on tasks.
  - Promising results in the zero-shot and one-shot settings for many tasks.
  - In the few-shot setting, sometimes competitive with or even occasionally surpasses state-of-the-art (fine-tuned models)
    - CoQA, TriviaQA
    - MT (translating to English)
    - Common sense reasoning (PIQA)
  - Lag below SOTA in performing natural language inference tasks (ANLI), some reading comprehension datasets(RACE, QuAC)
  - Interesting and impressive results on synthetic and qualitative tasks

GPT-3 on News Generation

- Given title and subtitle, generate short articles in the news genre

"GPT-3 can generate synthetic news articles which human evaluators have difficulty distinguishing from human-generated articles."

	Mean accuracy	95% Confidence Interval (low, hi)	t compared to control (p-value)	"I don't know" assignments
Control (deliberately bad model)	86%	83%–90%	-	3.6 %
GPT-3 Small	76%	72%–80%	3.9 (2e-4)	4.9%
GPT-3 Medium	61%	58%–65%	10.3 (7e-21)	6.0%
GPT-3 Large	68%	64%–72%	7.3 (3e-11)	8.7%
GPT-3 XL	62%	59%–65%	10.7 (1e-19)	7.5%
GPT-3 2.7B	62%	58%–65%	10.4 (5e-19)	7.1%
GPT-3 6.7B	60%	56%–63%	11.2 (3e-21)	6.2%
GPT-3 13B	55%	52%–58%	15.3 (1e-32)	7.1%
GPT-3 175B	52%	49%–54%	16.9 (1e-34)	7.8%

Table 3.11: Human accuracy in identifying whether short (~200 word) news articles are model generated.

Examples of Generated Articles

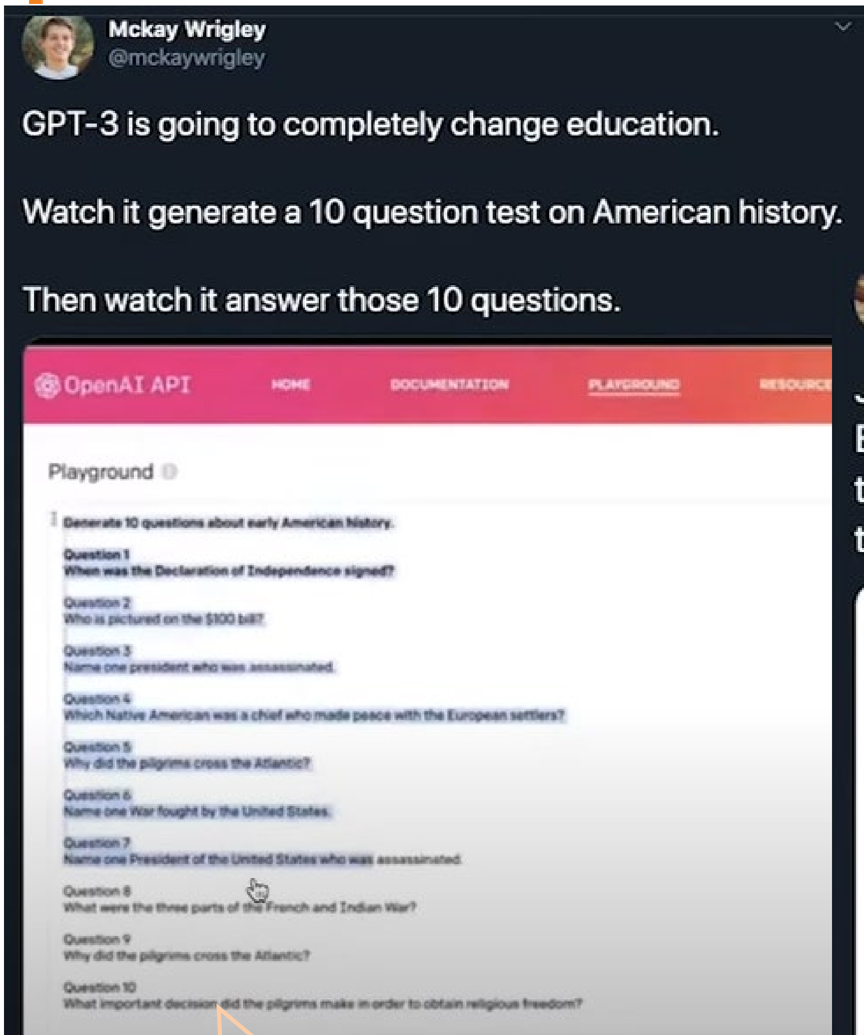
Title: Star's Tux Promise Draws Megyn Kelly's Sarcasm
Subtitle: Joaquin Phoenix pledged to not change for each awards event
Article: A year ago, Joaquin Phoenix made headlines when he appeared on the red carpet at the Golden Globes wearing a tuxedo with a paper bag over his head that read, "I am a shape-shifter. I can't change the world. I can only change myself." It was a promise to not change to fit into the Hollywood mold: "I think that's a really special thing, to not change yourself. I think it's a really special thing to say, 'This is what's inside of me, I'm proud of it, and I'm not going to be ashamed because of the way that someone else thinks I should be.'" Now, it's the Oscars, and Phoenix is at it again. But this time, his publicist is saying he'll be wearing a tux no matter what.
Megyn Kelly was not impressed, and she let him have it on The Tonight Show. "You know, I feel like, I feel like you could have worn the tux," she says. "But you're saying you're a shape-shifter. I don't know if you can change your tux, but you can change your mind. You can change your mind. You can change your mind." Phoenix says he did, but it didn't stick. "I was like, 'Okay, I'm going to wear a tuxedo to this thing.
don't want to wear a tuxedo to this thing.'" Kel
to change his mind again, but Phoenix says it's
wearing this."

Title: United Methodists Agree to Historic Split
Subtitle: Those who oppose gay marriage will form their own denomination
Article: After two days of intense debate, the United Methodist Church has agreed to a historic split - one that is expected to end in the creation of a new denomination, one that will be "theologically and socially conservative," according to The Washington Post. The majority of delegates attending the church's annual General Conference in May voted to strengthen a ban on the ordination of LGBTQ clergy and to write new rules that will "discipline" clergy who officiate at same-sex weddings. But those who opposed these measures have a new plan: They say they will form a separate denomination by 2020, calling their church the Christian Methodist denomination.
The Post notes that the denomination, which claims 12.5 million members, was in the early 20th century the "largest Protestant denomination in the U.S.," but that it has been shrinking in recent decades. The new split will be the second in the church's history. The first occurred in 1968, when roughly 10 percent of the denomination left to form the Evangelical United Brethren Church. The Post notes that the proposed split "comes at a critical time for the church, which has been losing members for years," which has been "pushed toward the brink of a schism over the role of LGBTQ people in the church." Gay marriage is not the only issue that has divided the church. In 2016, the denomination was split over ordination of transgender clergy, with the North Pacific regional conference voting to ban them from serving as clergy, and the South Pacific regional conference voting to allow them.

Synthetic and Qualitative Tasks

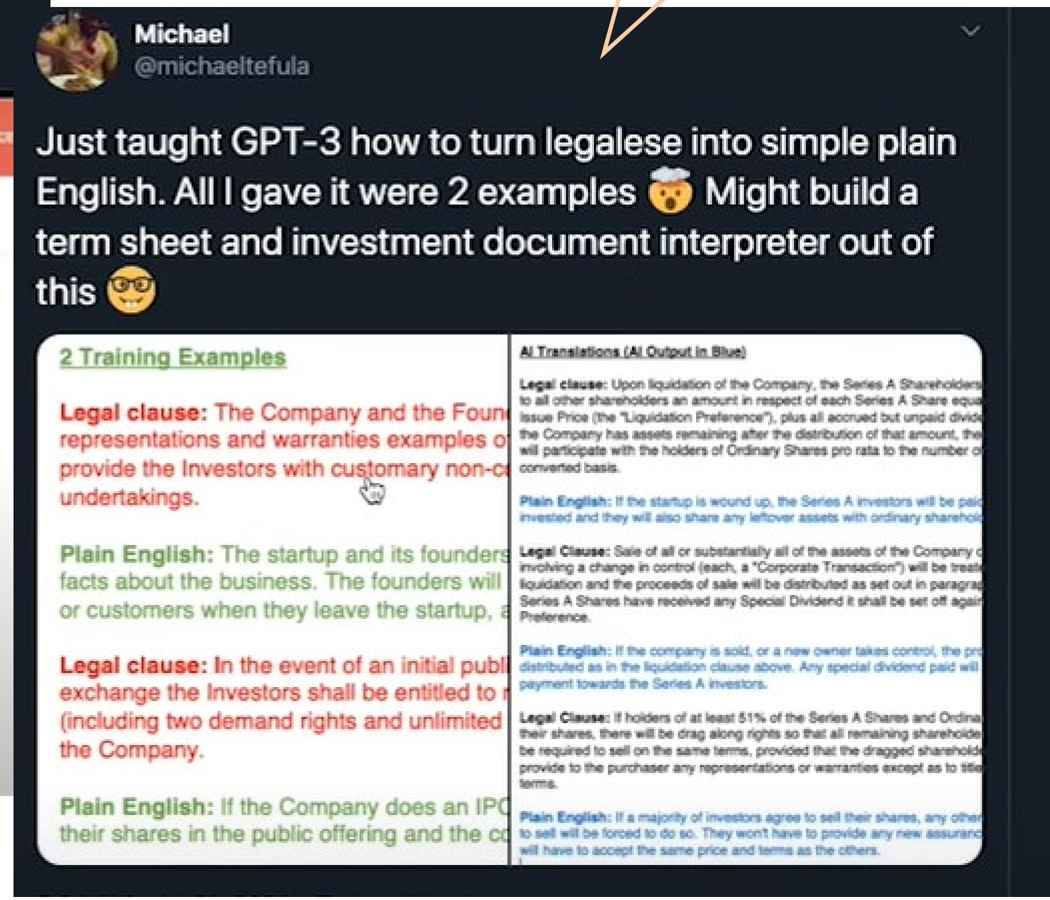
- Arithmetic
  - E.g. "***Q: What is 34 minus 53? A: -19***".
  - GPT-3 Few-shot does well on 2D and 3D operations.
- Word unscrambling
  - E.g. "***lyinevitab" = "inevitably***",
    "***s.u!c/c!e.s s i/o/n" = "succession***" (random insertion)
- SAT Analogies
  - E.g. "***audacious is to boldness as***
    (a) sanctimonious is to hypocrisy,
    (b) anonymous is to identity,
    (c) remorseful is to misdeed,
    (d) deleterious is to result,
    (e) impressionable is to temptation"

What Can GPT-3 do? Generating stuff!

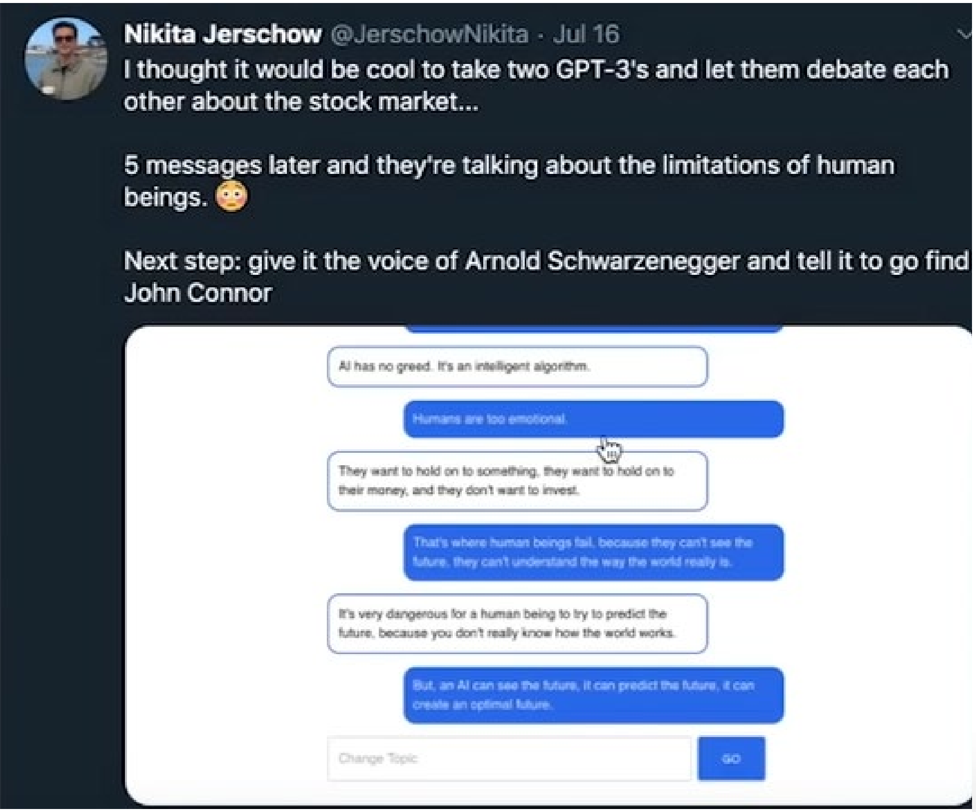


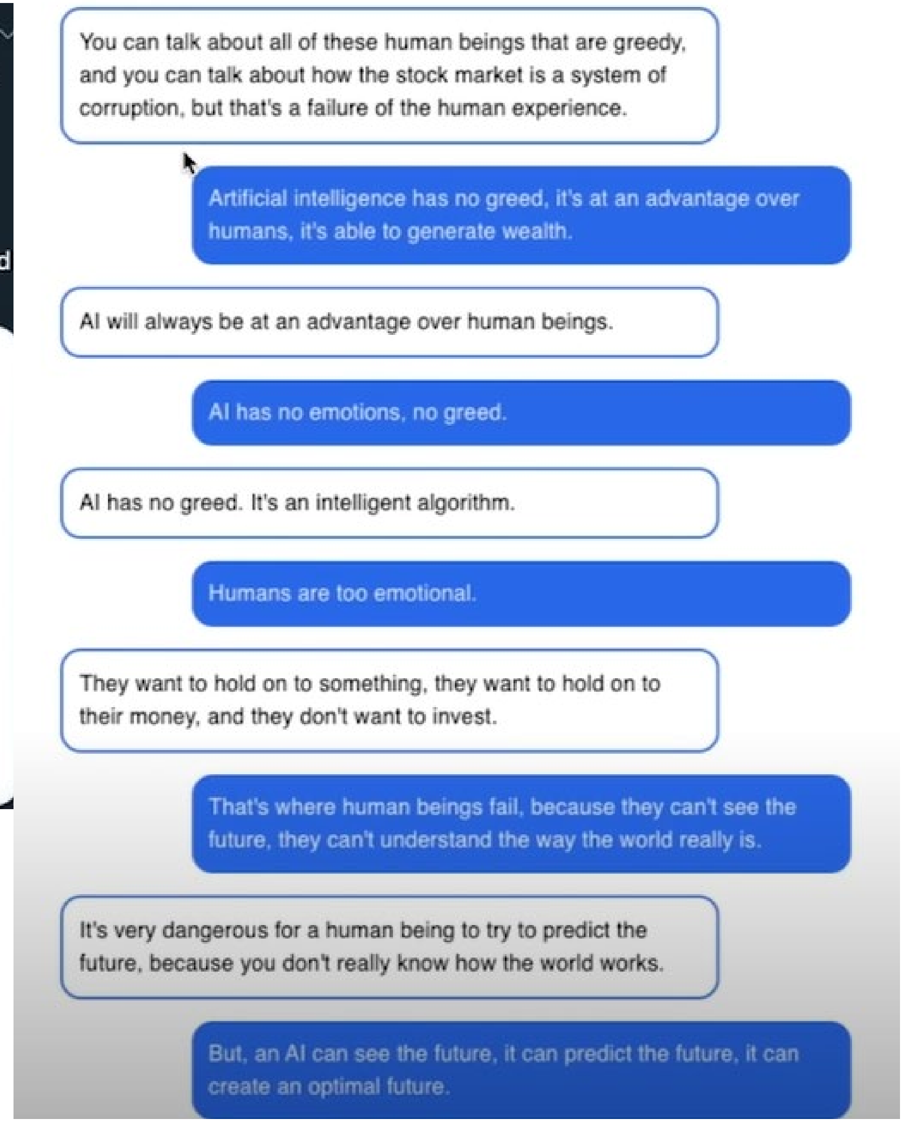
Exam questions and answers

Legal English to plain English

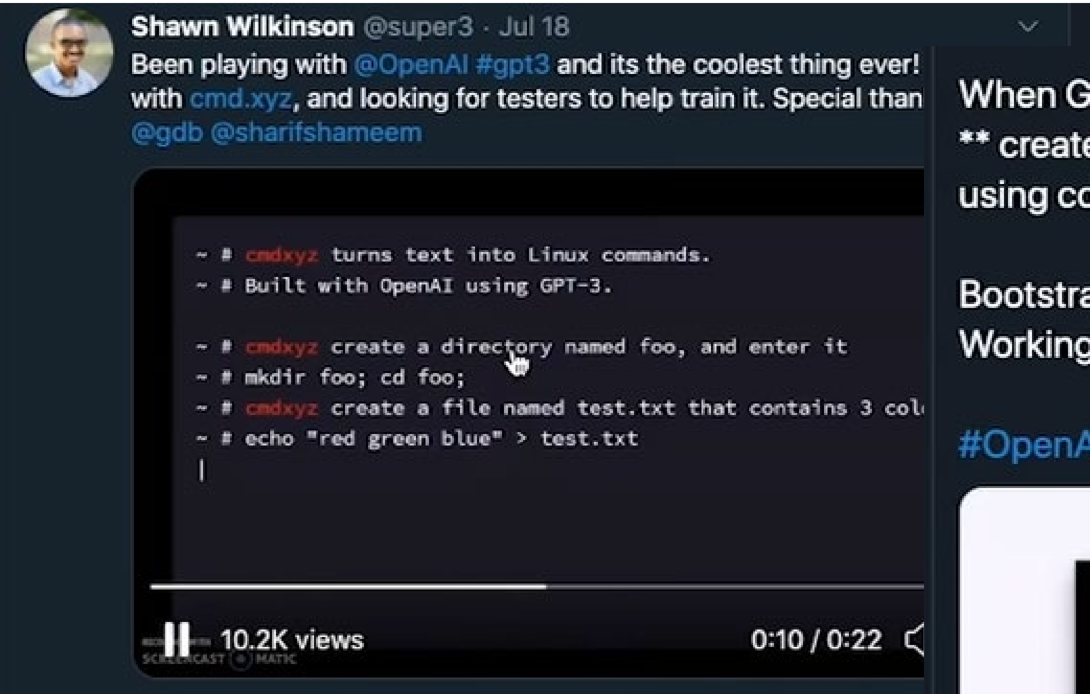


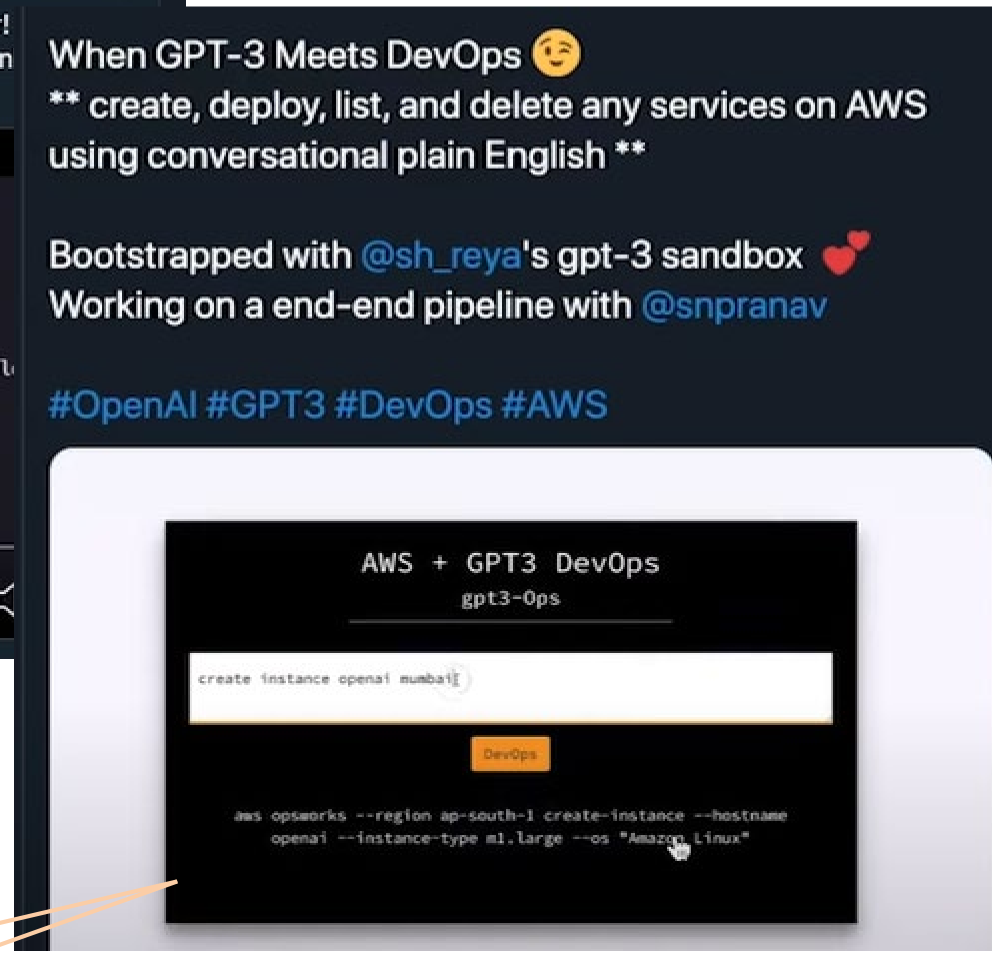
GPT-3 debating itself…





Generating commands…

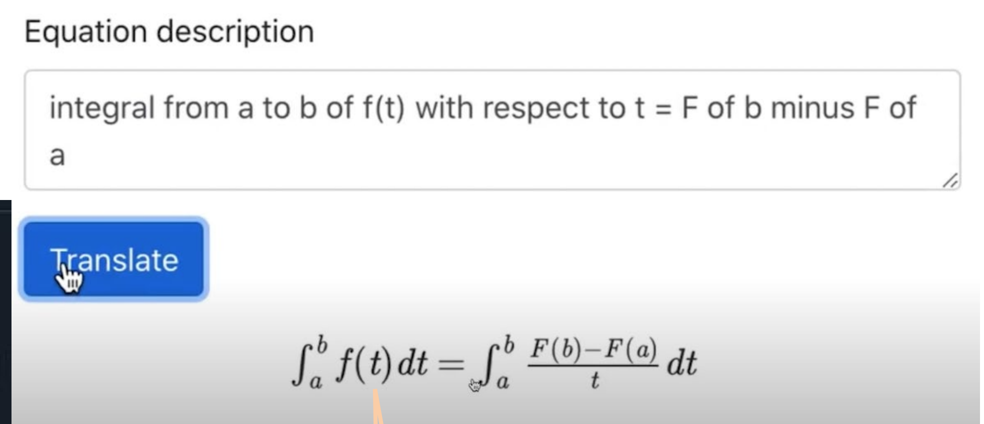




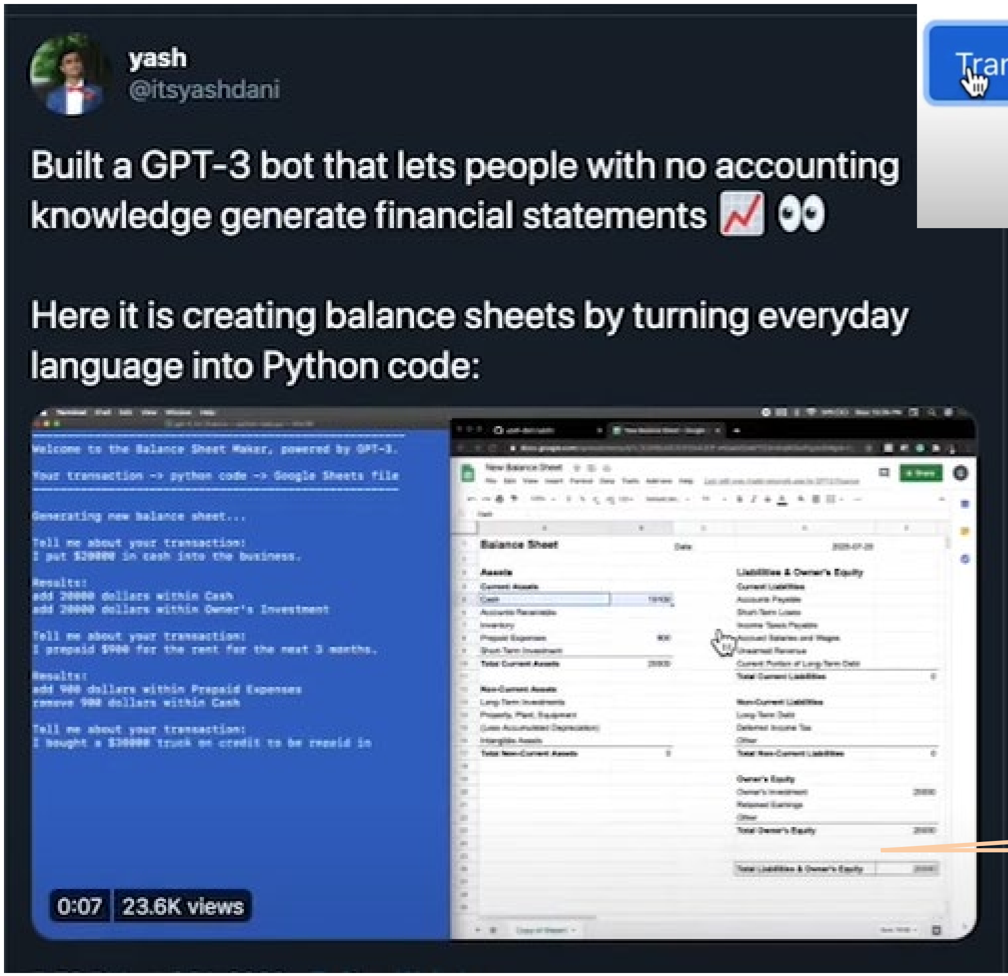
Linux commands

AWS commands

Generating codes

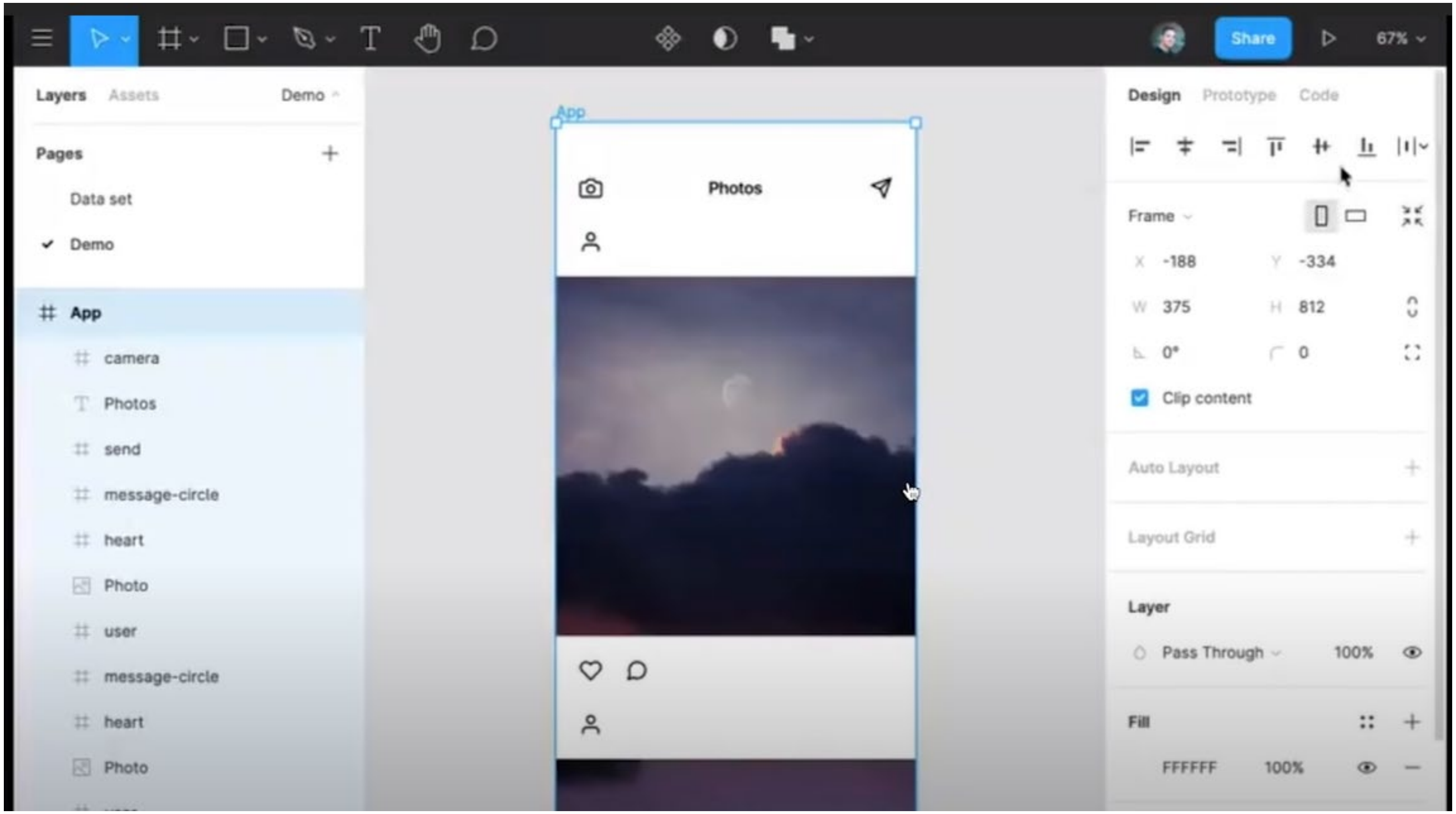


Latex



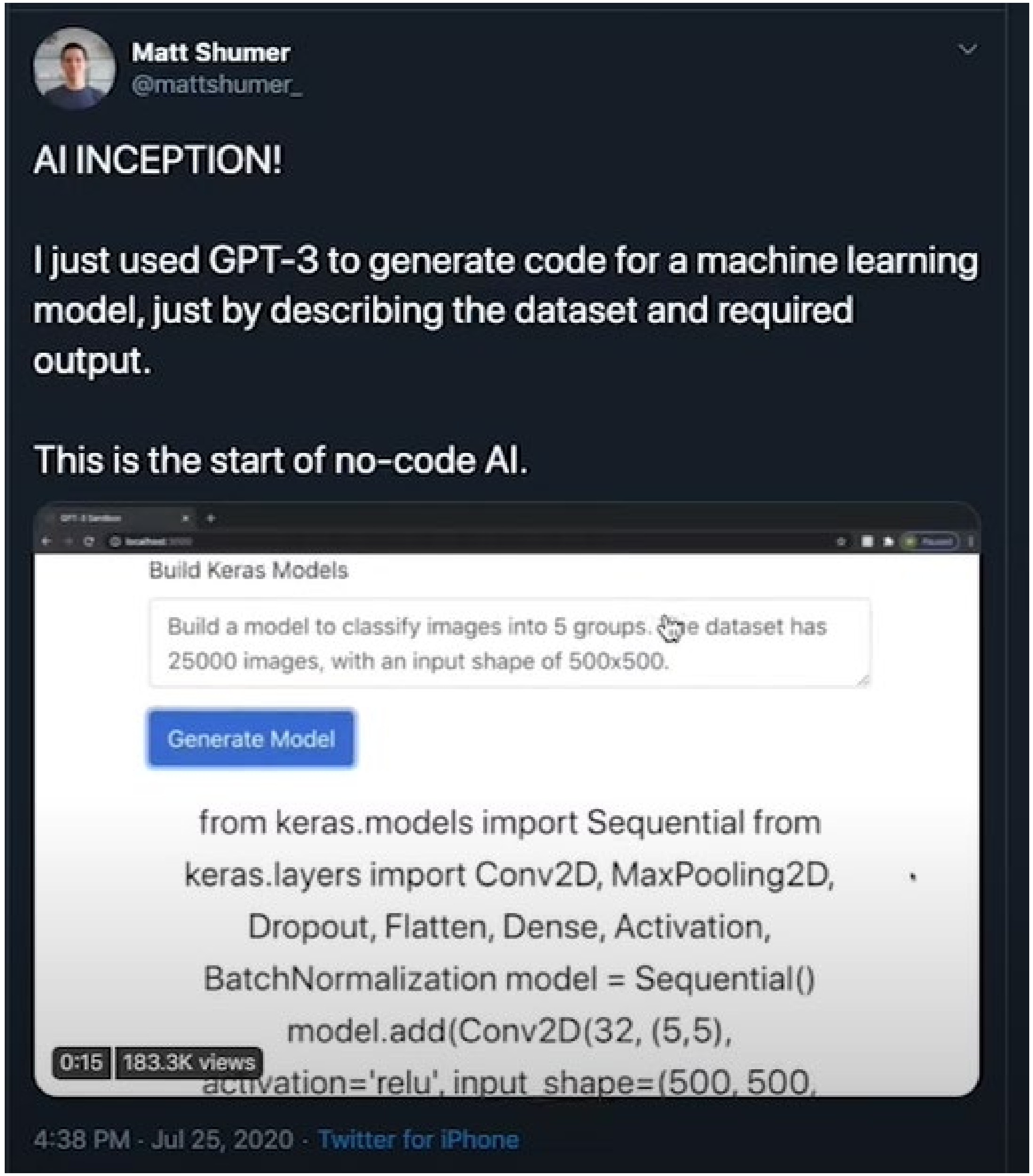
Python

Generating UI design



"An app that has a navigation bar with a camera icon, "Photos" title, and a message icon, a feed of photos with each photo having a user icon, a photo, a heart icon, and a chat bubble icon"

And…


Matt Shumer
@mattshumer_
AI INCEPTION!
I just used GPT-3 to generate code for a machine learning model, just by describing the dataset and required output.
This is the start of no-code AI.
Build Keras Models
Build a model to classify images into 5 groups. e dataset has 25000 images, with an input shape of 500x500.
Generate Model
from keras.models import Sequential from keras.layers import Conv2D, MaxPooling2D, Dropout, Flatten, Dense, Activation, BatchNormalization model = Sequential() model.add(Conv2D(32, (5,5),
0:15
183.3K views
4:38 PM · Jul 25, 2020 · Twitter for iPhone

A Huge Blackbox

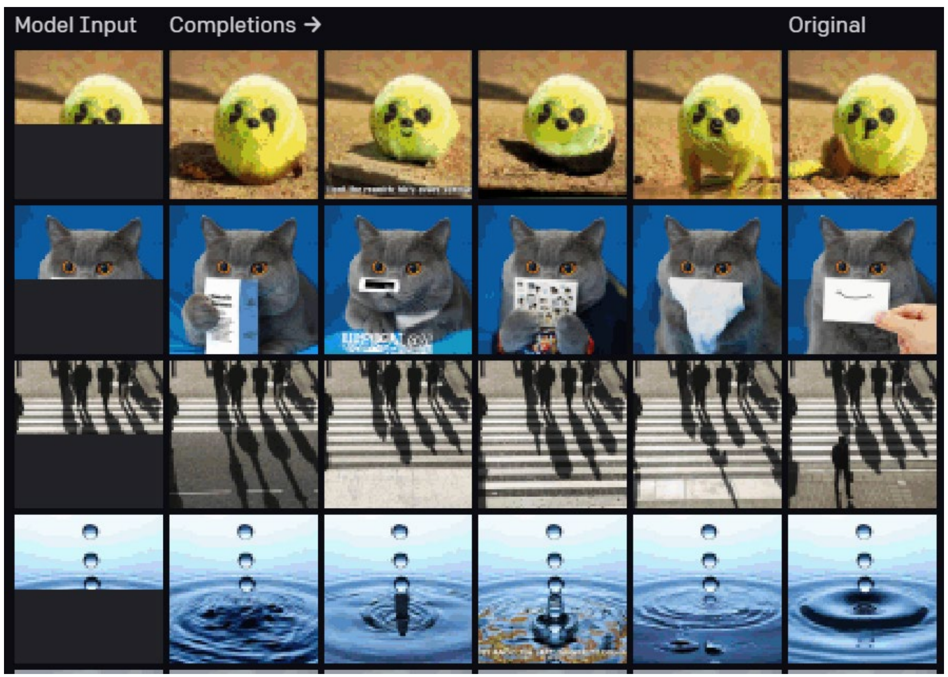
Exactly what's going on inside GPT-3 isn't clear. But what it seems to be good at is synthesizing text it has found elsewhere on the internet, making it a kind of vast, eclectic scrapbook created from millions and millions of snippets of text that it then glues together in weird and wonderful ways on demand.

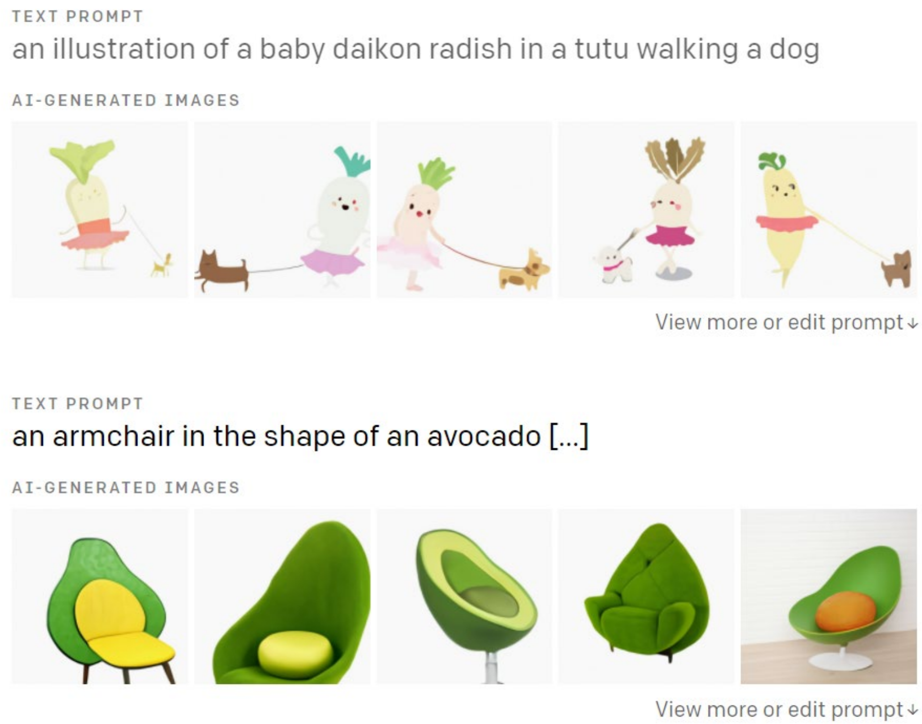
- "OpenAI's new language generator GPT-3 is shockingly good—and completely mindless", MIT Technology Review

Future?

GPT family going beyond text and language

- MuseNet (Apr 2019) – generate musical compositions
- Image GPT (Jun 2020) – trained on pixel sequences to generate coherent image completions and samples
- DALL·E (Jan 2021) - a 12-billion parameter version of GPT-3 trained to generate images from text descriptions





Future?

- Partner with AI

"The recent, almost accidental, discovery that GPT-3 can sort of write code does generate a slight shiver."
— John Carmack, 3D computer graphics pioneer

"You're not mastering the tool any longer, you're mastering the problem — and letting the computer do all the work."
— Caleb Meyer, Project Industrial Designer

References

- Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805* (2018).
- Brown, Tom B., et al. "Language models are few-shot learners." *arXiv preprint arXiv:2005.14165* (2020).
- Radford, Alec, et al. "Language models are unsupervised multitask learners." *OpenAI Blog* 1.8 (2019): 9.
- Radford, Alec, et al. "Improving language understanding by generative pre-training." (2018): 12.
- Sanh, Victor, et al. "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter." *arXiv preprint arXiv:1910.01108* (2019).
- Jiao, Xiaoqi, et al. "Tinybert: Distilling bert for natural language understanding." *arXiv preprint arXiv:1909.10351* (2019).
- How to generate text: using different decoding methods for language generation with Transformers, by Patrick von Platen (https://huggingface.co/blog/how-to-generate)
- Jay Jalammar's blogs with visual illustrations
  - http://jalammar.github.io/a-visual-guide-to-using-bert-for-the-first-time/
  - http://jalammar.github.io/illustrated-gpt2/
  - http://jalammar.github.io/how-gpt3-works-visualizations-animations/