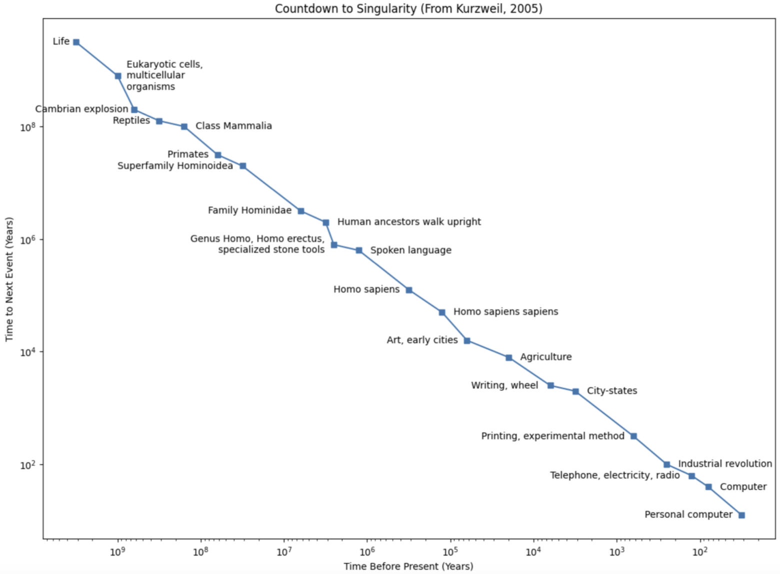
Generative Pre-Training: the (after)math

Carlos García Meixide

Instituto de Ciencias Matemáticas

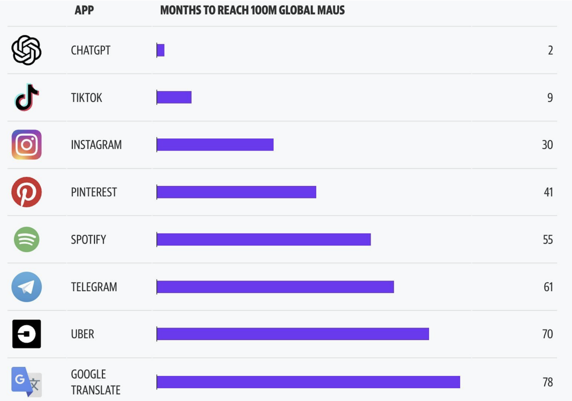
Madrid, April 2024

AI doomers: the *singularity*



Source: David Donoho, "Data Science at the Singularity"

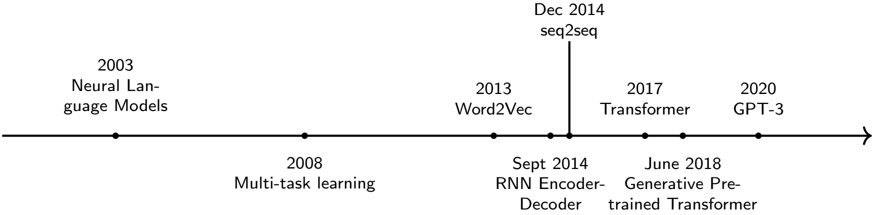
The fastest-growing consumer application to date



- ChatGPT currently has over 180 million users
- In just 5 days, ChatGPT surpassed 1 million users
- openai.com gets approximately 1.6 billion visits per month

Source: `https://explodingtopics.com/blog/chatgpt-users`

Historical context



From last Thursday

Welcome Llama 3 - Meta's new open LLM

Published April 18, 2024

Update on GitHub

philschmid Philipp Schmid · osanseviero Omar Sanseviero · pcuenq Pedro Cuenca · ybelkada Younes Belkada · lvwerra Leandro von Werra

Introduction

Meta's Llama 3, the next iteration of the open-access Llama family, is now released and available at Hugging Face. It's great to see Meta continuing its commitment to open AI, and we're excited to fully support the launch with comprehensive integration in the Hugging Face ecosystem.

Llama 3 comes in two sizes: 8B for efficient deployment and development on consumer-size GPU, and 70B for large-scale AI native applications. Both come in base and instruction-tuned variants. In addition to the 4 models, a new version of Llama Guard was fine-tuned on Llama 3 8B and is released as Llama Guard 2

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 question answering (RACE), and 1.5% on textual entailment (MultiNLI).

How does GPT work?

Stage 1: unsupervised pre-training

Given token observations $U_1, \ldots, U_n$

$$L_1(W, \theta, W) = \sum_i \log P_{W,\theta,W}\left(U_i \mid U_{i-k}, \ldots, U_{i-1}\right)$$

$$\mathcal{M} = \{P_{\theta,W} : (\theta, W) \in \Theta \times \mathbb{R}^H\}$$

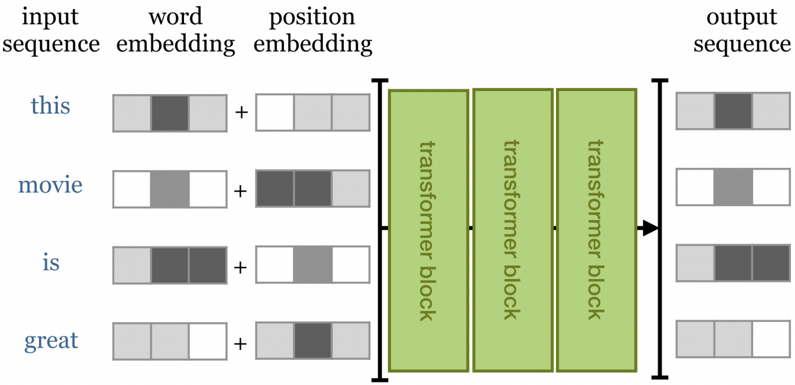
$$H = 50257 \times 12888 \text{ in GPT-3}$$

$$W \cdot \{\texttt{hot-encoded}\} =: h_0 \longrightarrow \texttt{decoder\_block}(h_0) = h_1 \longrightarrow \ldots$$
$$\ldots \longrightarrow \texttt{decoder\_block}(h_{n-1}) = h_n \longrightarrow \texttt{softmax}\left(h_n W^T\right)$$

Stage 2: supervised fine-tuning

Given $(X_1, Y_1), \ldots, (X_m, Y_m)$ with $X_i = (X_i^1, \ldots, X_i^p)$

$$L_2(W_y) = \sum_i \log P_{\hat{W},\hat{\theta},W_y}\left(Y_i \mid X_i^1, \ldots X_i^p\right) = \sum_i \texttt{softmax}\left(\widehat{h}_p(i) W_y^T\right)$$



https://peterbloem.nl/blog/transformers

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

Recent work has demonstrated substantial gains on many NLP tasks and benchmarks by pre-training on a large corpus of text followed by fine-tuning on a specific task. While typically task-agnostic in architecture, this method still requires task-specific fine-tuning datasets of thousands or tens of thousands of examples. By contrast, humans can generally perform a new language task from only a few examples or from simple instructions – something which current NLP systems still largely struggle to do. Here we show that scaling up language models greatly improves task-agnostic, few-shot performance, sometimes even reaching competitiveness with prior state-of-the-art fine-tuning approaches. Specifically, we train GPT-3, an autoregressive language model with 175 billion parameters, 10x more than any previous non-sparse language model, and test its performance in the few-shot setting. For all tasks, GPT-3 is applied without any gradient updates or fine-tuning, with tasks and few-shot demonstrations specified purely via text interaction with the model. GPT-3 achieves strong performance on many NLP datasets, including translation, question-answering, and cloze tasks, as well as several tasks that require on-the-fly reasoning or domain adaptation, such as unscrambling words, using a novel word in a sentence, or performing 3-digit arithmetic. At the same time, we also identify some datasets where GPT-3's few-shot learning still struggles, as well as some datasets where GPT-3 faces methodological issues related to training on large web corpora. Finally, we find that GPT-3 can generate samples of news articles which human evaluators have difficulty distinguishing from articles written by humans. We discuss broader societal impacts of this finding and of GPT-3 in general.

The three settings we explore for in-context learning

Zero-shot

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

```
Translate English to French:     ← task description
cheese => ...................    ← prompt
```

One-shot

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

```
Translate English to French:     ← task description
sea otter => loutre de mer       ← example
cheese => ...................    ← prompt
```

Few-shot

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

```
Translate English to French:          ← task description
sea otter => loutre de mer            ← examples
peppermint => menthe poivrée          ←
plush girafe => girafe peluche        ←
cheese => ...................         ← prompt
```

Traditional fine-tuning (not used for GPT-3)

Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.

```
sea otter => loutre de mer            ← example #1
        ↓
[gradient update]
        ↓
peppermint => menthe poivrée          ← example #2
        ↓
[gradient update]
        ↓
       • • •
        ↓
plush giraffe => girafe peluche       ← example #N

[gradient update]

cheese => ...................         ← prompt
```

What is pre-training?

Standard constrained empirical risk minimization (ERM) problem over transformers with L layers, M heads, and norm bound B

$$\widehat{\boldsymbol{\theta}} := \operatorname*{arg\,min}_{\boldsymbol{\theta} \in \Theta_{L,M,D',B}} \widehat{L}_{\rm icl}(\boldsymbol{\theta})$$

$$\Theta_{L,M,D',B} := \{\boldsymbol{\theta} = \left(\boldsymbol{\theta}_{\rm attn}^{(1:L)}, \boldsymbol{\theta}_{\rm mlp}^{(1:L)}\right) : \max_{\ell \in [L]} M^{(\ell)} \le M, \max_{\ell \in [L]} D^{(\ell)} \le D', \|\boldsymbol{\theta}\| \le B\}$$

Metric entropy

- Upper and lower bounds on the metric entropy of a given unit ball in terms of its own norm.
- $\mathbb{B}$ unit ball of a normed vector space.

$$d \log(1/\delta) \leq \log N(\delta; \mathbb{B}, \|\cdot\|) \leq d \log\left(1 + \frac{2}{\delta}\right)$$

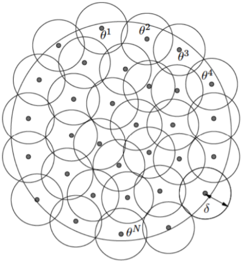
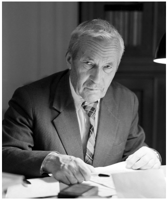
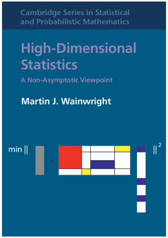
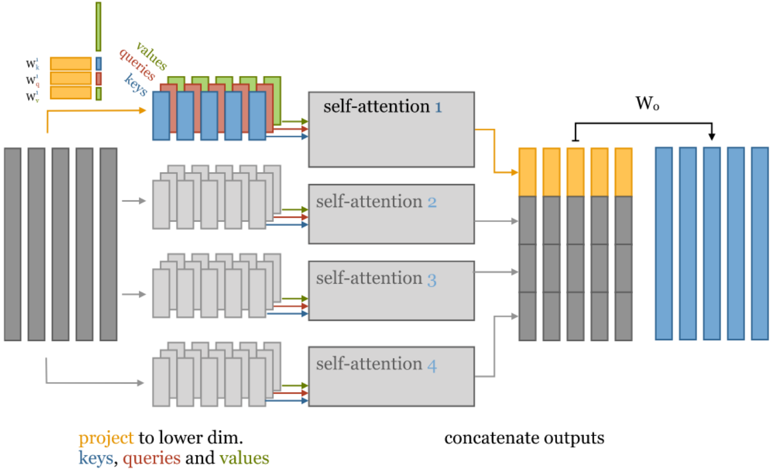


Figure 1: [courtesy: Martin Wainwright's book]





Multi-head self-attention



https://peterbloem.nl/blog/transformers

Theorem (In-context linear regression, Bai et al. (2023))

Suppose $\mathrm{P} \sim \pi$ *is almost surely well-posed for in-context linear regression. Then, for* $N \geq \widetilde{\mathcal{O}}(d)$*, with probability at least* $1 - \xi$ *, the solution* $\widehat{\boldsymbol{\theta}}$ *of (TF-ERM) with* $L = \mathcal{O}(\kappa \log(\kappa N/\sigma))$ *layers,* $M = 3$ *heads,* $D' = 0$ *(attention-only), and* $B = \mathcal{O}(\sqrt{\kappa d})$ *achieves small excess ICL risk over* $\mathbf{w}_{\mathrm{p}}^{\star}$ *:*

$$L_{icl}\left(\widehat{\boldsymbol{\theta}}\right) - \mathbb{E}_{\mathrm{P}\sim\pi}\mathbb{E}_{(\mathbf{x},y)\sim\mathrm{P}}\left[\frac{1}{2}\left(y - \langle \mathbf{w}_{\mathrm{P}}^{\star}, \mathbf{x}\rangle\right)^2\right] \leq \widetilde{\mathcal{O}}\left(\sqrt{\frac{L^2MD^2 + \log(1/\xi)}{n}} + \frac{d\sigma^2}{N}\right)$$

where $\widetilde{\mathcal{O}}(\cdot)$ *only hides polylogarithmic factors in* $\kappa, N, 1/\sigma$*.*

GPT-1: the first autoregressive transformer model (2018)

- trained on the Books corpus.
- train on a language modeling task, as well as a multi-task that adds a supervised learning task.

GPT-2 (2019):

- all articles linked from Reddit with at least 3 upvotes (8 million documents, 40 GB of text)
- dispense with supervised learning task, make some other architectural adjustments
- make model much bigger

GPT-3 (2020):

- use an even bigger corpus (Common Crawl, WebText2, Books1, Books2 and Wikipedia)
- make model much, much bigger

GPT-1:

- 768-dimensional word embeddings
- 12 transformer blocks with 12 attention heads
- 512-token context window
- $\approx 117\mathrm{M}$ parameters

GPT-2:

- 1600-dimensional word embeddings
- 48 blocks with 48 attention heads
- 1024-token context window
- $\approx 1.5\ \mathrm{B}$ parameters

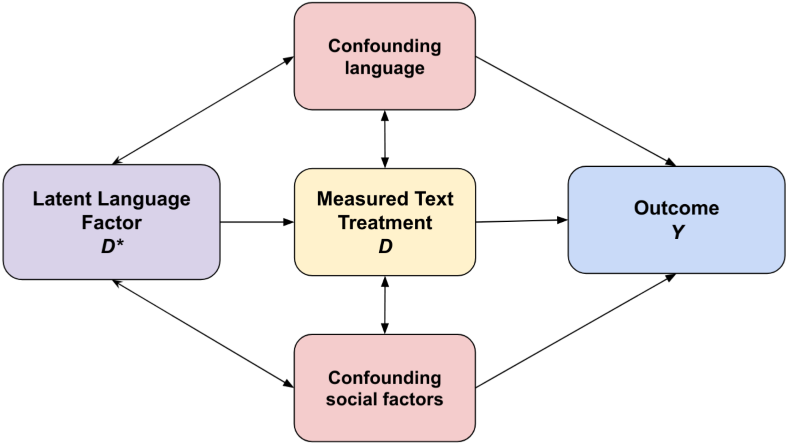
GPT-3:

- 12,888-dimensional word embeddings
- 96 blocks with 96 attention heads
- 2048-token context window
- $\approx 175\ \mathrm{B}$ parameters

What about GPT-4?

- Embedding dimension, architecture... are corporate secret
- 128k-token context window (*turbo*)
- $\approx 1.76\ \mathrm{B}$ **trillion** parameters

Epilogue: causal inference with text



Source: NLP for Law and Social Sciences course by Elliot Ash, ETH Zürich

Sridhar et al. (2019): causally sufficient text embeddings

Biometrika (1983), **70**, 1, *pp*. 41–55
Printed in Great Britain

The central role of the propensity score in observational studies for causal effects

BY PAUL R. ROSENBAUM

Departments of Statistics and Human Oncology, University of Wisconsin, Madison, Wisconsin, U.S.A.

AND DONALD B. RUBIN

University of Chicago, Chicago, Illinois, U.S.A.

THEOREM 3. *If treatment assignment is strongly ignorable given* x, *then it is strongly ignorable given any balancing score* $b(x)$; *that is,*

$$(r_1, r_0) \perp\!\!\!\perp z \mid x$$

and

$$0 < \mathrm{pr}(z = 1 \mid x) < 1$$

for all x *imply*

$$(r_1, r_0) \perp\!\!\!\perp z \mid b(x)$$

and

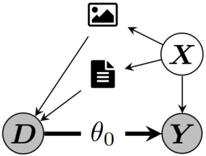
$$0 < \mathrm{pr}\{z = 1 \mid b(x)\} < 1$$

for all $b(x)$.

February 2024

DoubleML$_{\text{Deep}}$: Estimation of Causal Effects with Multimodal Data

Sven Klaassen [1 2] **Jan Teichert-Kluge** [2] **Philipp Bach** [2] **Victor Chernozhukov** [3] **Martin Spindler** [1 2]
Suhas Vijaykumar [3]



$$Y = \theta_0 D + g_0(X) + \varepsilon, \quad \mathbb{E}[\varepsilon \mid X, D] = 0$$
$$D = m_0(X) + \vartheta, \qquad \mathbb{E}[\vartheta \mid X] = 0$$

Thank you!