大規模言語モデル

出典: フリー百科事典『ウィキペディア（Wikipedia）』

大規模言語モデル（だいきぼげんごモデル、英: large language model、**LLM**）は、多数のパラメータ（数千万から数十億）を持つ人工ニューラルネットワークで構成されるコンピュータ言語モデルで、膨大なラベルなしテキストを使用して自己教師あり学習または半教師あり学習によって訓練が行われる[1]。

LLMは2018年頃に登場し、さまざまなタスク（仕事）で優れた性能を発揮している。これにより、自然言語処理の研究の焦点は、特定のタスクに特化した教師ありモデルを訓練するという以前のパラダイムから転換した[2]。大規模言語モデルの応用は目覚ましい成果を上げているが、大規模言語モデルの開発はまだ始まったばかりであり、多くの研究者が大規模言語モデルの改良に貢献している[3]。

大規模言語モデルという用語の正式な定義はないが、大規模コーパスで事前訓練された、数百万から数十億以上のパラメータを持つディープラーニングモデルを指すことが多い。LLMは、特定のタスク（感情分析、固有表現抽出、数学的推論など）のために訓練されたものとは異なり、幅広いタスクに優れた汎用モデルである[2][4]。LLMがタスクを実行する能力や対応可能な範囲は、ある意味では設計における画期的な進歩には依存せず、LLMに費やされた資源（データ、パラメータサイズ、計算力）の量の関数であるように見える[5]。多数のパラメータを持ったニューラル言語モデルは、文の次の単語を予測するという単純なタスクで十分に訓練することで、人間の言葉の構文や意味の多くを捉えられることがわかった。さらに、大規模な言語モデルは、世の中に関するかなりの一般知識を示し、訓練中に大量の事実を「記憶」することができる[2]。

質の高い証拠とされる2023年のメタ分析によれば、大規模言語モデルの創造性に目を輝かせる研究者はもちろん世界中に存在し、小規模言語モデルにはできないタスクで大規模言語モデルが創造的であると主張する学者もいるが、これは測定基準の選択によるものであり、創造性によるものではないことが示唆されている。異なる測定基準を選択した場合、大規模言語モデルの創造性の優位性は見られない可能性が示唆されている[6]。

特性

事前訓練データセット

「機械学習研究のためのデータセットリスト」も参照

大規模言語モデル（LLM）は通常、さまざまな分野や言語にわたる大量のテキストデータで事前訓練が行われる[7]。著名な事前訓練データとしては、Common Crawl、The Pile、MassiveText[8]、Wikipedia、GitHubなどが知られている。大半のオープンソースのLLMは一般公開されているデータを利用しているが、非公開のデータで事前訓練が行われることもある[9]。事前訓練データは、重複排除、毒性が高いシーケンスの除外、低品質データの破棄など、生テキストをさまざまな手順で前処理して作成される[10]。言語データの蓄積は年率7%で増加しており、2022年10月現在、高品質な言語データは4兆6,000億語から17兆語の範囲内にあると推定されている[11]。LLMでは事前訓練データを広範に使用するため、事前訓練データに評価データが混入すると、ベンチマーク評価時のモデル性能に影響を与えるデータ汚染が起こる[12]。

スケーリング則

詳細は「ニューラルスケーリング則」を参照

一般にLLMは、モデルの大きさ、訓練データセットの大きさ、訓練費用、訓練後の性能という4つのパラメータにより特徴づけられる。これらの4つの変数はそれぞれ実数で正確に定義することができ、経験から「スケーリング則（scaling laws）」と呼ばれている単純な統計的法則によって関係していることがわかっている。

ケーリング則（Chinchillaスケーリング）を、次のように表している[13]。

$$\begin{cases} C = C_0 N D \\ L = \frac{A}{N^\alpha} + \frac{B}{D^\beta} + L_0 \end{cases}$$

ここで、変数は次のとおりである。

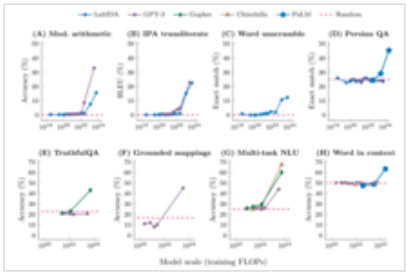
- C：モデルの訓練に掛かる費用（FLOPS単位）
- N：モデル内のパラメータ数
- D：訓練セット内のトークン数
- L：テストデータセットで訓練されたLLMにより達成される、トークン当たりの平均の負対数尤度損失（ナット/トークン）

統計パラメータは次のとおりである。

- $C_0 = 6$、すなわち、1つのトークンで訓練するにはパラメータごとに6 FLOPSの費用がかかる[14]。ここで、訓練費用は推論費用よりもはるかに高いことに注意を要する。1つのトークンを推論する費用はパラメータあたり1〜2 FLOPSである。
- $\alpha = 0.34, \beta = 0.28, A = 406.4, B = 410.7, L_0 = 1.69$

創発的能力

一般に、さまざまなタスクに対する大規模モデルの性能は、同様の小規模モデルの性能に基づいて推定することができるが、ときには、下流におけるスケーリング則が「破綻」し[15]、大規模モデルが、小規模モデルとは異なる速度で突然に能力を獲得することがある。これは「**創発的能力**」（英: emergent abilities）として知られているもので、これまでも多くの研究の対象であった。研究者は、こうした能力は「小規模モデルの性能を外挿することでは予測できない」ことが多いと指摘している[4]。このような能力は、プログラムされたり設計されたりするものではなく、むしろ「発見される」ものであり、場合によっては、LLMが一般公開されて初めて発見されることすらある[5]。これまでに数百もの創発的能力が報告されている。たとえば、多段階の算術、大学レベルの試験、単語の意図する意味の特定[4]、思考の連鎖[4]、国際音声記号の解読、マス埋めパズル、ヒングリッシュ（ヒンディー語と英語の混成語）の段落内の不快な内容の特定、およびスワヒリ語のことわざに相当する英語の生成などがある[16]。



質問応答などのタスクを含め、多くの自然言語ベンチマークでは、モデルがある規模に達するまでは偶然によるものよりも性能が劣り、ある時点で性能が急激に向上する。それぞれの図は創発的能力の例を示している。モデル規模は訓練の計算量によって測定されている。

Schaefferらは、創発的な能力は予測不可能な形で獲得されるのではなく、滑らかなスケーリング則に従って予測通りに獲得されると主張している[17]。著者らは、LLMが多肢選択問題を解く統計的トイモデルを検討し、他の種類のタスクを考慮して修正されたこの統計モデルが、これらのタスクにも適用できることを示した。

ここで、x をパラメータ数、y をモデルの性能とする。

- $y = \text{average } Pr(\text{correct token})$のとき、$(\log x, y)$ は指数曲線（1でプラトーに達する前）となり、創発のように見える。
- $y = \text{average } \log(Pr(\text{correct token}))$ のとき、$(\log x, y)$ のプロットは直線（0でプラトーに達する前）となり、創発には見えない。

- y = average Pr(the most likely token is correct) のとき、$(\log x, y)$ はスケック関数となり、創発のように見える。

歴史

先駆者

大規模言語モデルの基本的な考え方は、単純で反復的なアーキテクチャを持つランダムな重みを持つニューラルネットワークを出発点とし、大規模な言語コーパスで訓練することである。

この最も初期の例のひとつがエルマンネットワークで[18]、「犬が男を追いかける」のような単純な文でリカレントネットワークを訓練した。訓練したネットワークは、各単語をベクトル（内部表現）に変換した。次にこれらのベクトルを接近度によって木構造にクラスタリングした。その結果、ツリーはある構造を示すことがわかった。動詞と名詞はそれぞれ別の大きなクラスターに属していた。名詞のクラスター内には、無生物（inanimates）と生物（animates）の2つの小さなクラスターがある、などである。

別の方法として、自然言語理解を記号プログラムによってコンピュータにプログラムする論理AIがあった。この方法は1990年代まで主流であった。単純な機構と大規模なコーパスによって自然言語を学習するという着想は1950年代に始まったが、商業的に最初に成功したのは、統計的機械翻訳のためのIBMアライメントモデル（1990年代）であった。

Transformerフレームワークへの進化

初期の「大規模」言語モデルは、長期・短期記憶（LSTM、1997年）などのリカレントアーキテクチャを使用して構築された。AlexNet（2012年）が画像認識における大規模ニューラルネットワークの有効性を実証した後、研究者は大規模ニューラルネットワークを他のタスクに適用した。2014年には、2つの主要な手法が提案された。

- seq2seqモデル（3億8,000万パラメータ）は、2つのLSTMを使用して機械翻訳を行い[19]、単純化されたアーキテクチャ（ゲート付き回帰型ユニット、GRU）で同じ手法が使われた（1億3000万パラメータ）[20]。
- アテンション機構は、2つのLSTMの中間に「アテンション機構」を追加してseq2seqモデルを改良されたものとして提案された[21]。これはTransformerのアテンション機構とは異なるが、同様のタスクを実行する。

2016年、Google翻訳はその機構を統計的機械翻訳からニューラル機械翻訳へと変更した。これは、LSTMとアテンションによるseq2seqである。10年かけて構築された以前のシステムよりも高いレベルの性能に到達するのに9カ月を要したという[22][23]。

2017年の論文「*Attention is all you need*」[24]では、アテンション機構を抽象化して[21]、アテンション機構を中心としたTransformerアーキテクチャを構築した。seq2seqモデルは、他のリカレントネットワークと同様、入力シーケンスを一度に1つずつ処理しなければならないのに対し、Transformerアーキテクチャはシーケンス上で並列に実行することができる。これによって、より大規模なモデルを訓練できるようになった。

BERTとGPT

BERT（2018年）[25]は双方向Transformerであり、GPT（2018年）[26][27]は単方向（自己回帰）Transformerである。これらは2023年時点の主要なアーキテクチャである。

アーキテクチャ

大規模言語モデルでは、2018年以降、逐次データに対する標準的なディープラーニング手法となったTransformer（トランスフォーマー）アーキテクチャが最もよく使用されている[2]。別のアーキテクチャの系統として、混合エキスパート（Mixture of experts、MoE）がある。これはGoogleが開発したAIモデルでし

ましは使用されており、sparsely gated MoE（2017年）に始まり、Gshard（2021年）、GLaM（2022年）へと続いている[30]。

トークン化

LLMは数学的な関数であり、その入力と出力は数字のリストである。したがって、単語は数値に変換しなければならない。

一般に、LLMはこれを行うために固有のトークナイザを使用し、テキストと整数のリストを対応させている。通常、LLMを訓練する前にトークナイザを訓練データセットの全体に適用し、その後は凍結する。トークナイザにはバイト対符号化が選択されるのが一般的である。

トークナイザのもう一つの機能は、計算量を削減するためのテキスト圧縮である。たとえば「where is（どこにありますか)」などの一般的な単語やフレーズは、7文字ではなく1つのトークンでエンコードすることができる。OpenAI GPTシリーズでは、1つのトークンが一般的な英語テキストの約4文字、つまり約0.75語に相当するトークナイザを使用している[31]。珍しい英語のテキストは予測しにくく、そのため圧縮が困難となり、より多くのトークンを必要とする。

トークナイザは、任意の整数を出力することはできない。一般的には $\{0, 1, 2, \ldots, V-1\}$ の範囲の整数に限って出力される。ここで、V は語彙（ごい）サイズと言う。

トークナイザには、任意のテキストを扱えるものと（一般にUnicodeで直接操作する)、そうでないものがある。トークナイザは、エンコード不可能なテキストに遭遇した場合、「未知テキスト（unknown text)」を意味する特別なトークン（多くは0）を出力する。BERT論文にならって、[UNK] と表記されることが多い。

もう一つの特別なトークンは、「パディング」を表す [PAD]（多くは1）である。これは、一度に大量のテキストがLLMに入力されたときに、エンコードされたテキストが同じ長さになるよう調節するのに使用される。LLMでは一般に、入力の長さが一定のシーケンス（ジャグ配列という）であることを要求するため、エンコードした短いテキストを長いテキストにそろえるのにパディングを行う。

出力

LLMの出力は、その語彙の確率分布である。これは通常、次のように実装される。

- テキストを受信すると、大半のLLMはベクトル $\boldsymbol{y} \in \mathbb{R}^V$ を出力する。ここで、V は語彙サイズ（上述）である。
- ベクトル $\boldsymbol{y}$ はソフトマックス関数によって $\mathbf{softmax}(\boldsymbol{y})$ となる。

このプロセスでは通常、ベクトル $\boldsymbol{y}$ は非正規化ロジットベクトルといい、ベクトル $\mathbf{softmax}(\boldsymbol{y})$ は確率ベクトルと呼ばれる。ベクトル $\mathbf{softmax}(\boldsymbol{y})$ は V 個のエントリを持ち、すべて非負であり、その合計は1となるので、$\{0, 1, 2, \ldots, V-1\}$ に対する確率分布、つまりLLMの語彙に対する確率分布であると解釈することができる。

ソフトマックス関数は数学的に定義されており、変化するパラメータを持たないことに注意を要する。したがっては訓練は行われない。

コンテキストウィンドウ

LLMのコンテキストウィンドウは、LLMがトークンを生成するために使用できる最長のトークンシーケンスの長さである。もしLLMがコンテキストウィンドウより長いシーケンスに対してトークンを生成するときは、トークンシーケンスをコンテキストウィンドウまで切り詰めるか、アルゴリズムに一定の変更を加える必要がある。

2023年6月時点で、4kから16kまでのコンテキストウィンドウを備えたGPT-3.5を提供している[32]。

エンコーダーとデコーダーの用語

Transformerに基づくLLMでは、Transformerの原著論文で使われている用語とは多少異なる[33]。

- エンコーダのみ: フルエンコーダ、フルデコーダ
- エンコーダー - デコーダー: フルエンコーダー、自己回帰デコーダー
- デコーダのみ: 自己回帰エンコーダ、自己回帰デコーダ

ここでの「自己回帰」とは、「マスク化アテンション」節で説明したように、あるトークンからそれに続くすべてのトークンへのアテンションをゼロにするために、アテンションヘッドにマスクが挿入されることを意味する。

訓練

ほとんどのLLMは事前訓練されており、テキストトークンの訓練データセットが与えられると、モデルはデータセット内のトークンを予測する。このような事前訓練には一般に2つの形式がある[34]。

- 自己回帰モデル（GPT型、次単語予測）
  「私が食べるのが好きなのは」のようなテキスト部分が与えられると、モデルは「アイスクリーム」のような「次のトークン」を予測する。
- マスク済みモデル（BERT型[35]、穴埋め）
  「私は [MASK] クリームを [MASK] したい」 のようなテキスト部分が与えられると、モデルは「アイスを食べる」のような隠されたトークンを予測する。

LLMは、次文予測（Next Sentence Prediction、NSP）のように、データ分布の理解をテストする補助タスクを使用して訓練することもある[35]。この場合は、文の組が提示され、モデルはそれらが訓練コーパス内で連続して出現するかどうかを予測しなければならない。

通常、LLMは特定の損失関数、つまりトークンごとの平均負対数尤度（交差エントロピー損失とも呼ばれる）を最小化するように訓練する。たとえば、自己回帰モデルで「食べるのが好き」が与えられ、確率分布 $\boldsymbol{Pr}(\cdot|\text{I like to eat})$ を予測する場合、このトークンに対する負対数尤度損失は $-\log \boldsymbol{Pr}(\text{ice}|\text{I like to eat})$ となる。

訓練のとき、訓練を安定させるために正則化損失も使用される。ただし、正則化損失は通常、テストや評価の際には使用されない。また、負対数尤度だけでなく、他にも多くの評価項目がある。詳細については以下の節を参照のこと。

訓練用データセットの大きさ

最初期のLLMは、数十億語の規模のコーパスで訓練が行われた。

OpenAIのGPT（generative pre-trained transformer）シリーズの最初のモデルであるGPT-1は、2018年に、9億8500万語で構成されるBookCorpusで訓練された[36]。同年、BERTはBookCorpusと英語版Wikipediaの組み合わせで訓練され、合計で33億語になった[35]。それ以来、LLMの訓練用コーパスは桁違いに増加し続けており、トークン数は最大で数兆個に達した[35]。

訓練費用

LLMの訓練には計算費用がかかる。2020年の調査では、15億パラメータのモデル（当時の最先端技術より2桁小さい）の訓練にかかる費用は8万ドルから160万ドルと見積もられた[37][38]。その後、ソフトウェアとハードウェアの進歩により費用は大幅に下がり、2023年の論文では、120億パラメータのモデルを訓練するた

の費用は$2,300 A100 GPU時間であると報告されている[39]。

TransformerベースのLLMの場合、訓練コストは推論コストよりもはるかに高くなる。1つのトークンを訓練するのに1パラメータあたり6 FLOPSのコストがかかるのに対し、1つのトークンを推論するには1パラメータあたり1〜2 FLOPSである[14]。

2020年代の企業は、ますます大規模になるLLMに巨額の投資を行った。GPT-2（15億パラメータ、2019年）の訓練費用に5万ドル、またGoogle PaLM（54億パラメータ、2022年）は800万ドルを要した[40]。

下流タスクへの適用

2018年から2020年にかけて、特定の自然言語処理（NLP）タスクでLLMを使用するための標準的な方法は、「タスクに特化」した追加訓練によってモデルをファインチューニングすることであった。その後、GPT-3のような「より強力」なLLMでは、解決すべき問題をテキストプロンプトとしてモデルに提示したり、場合によっては、類似の問題とその解決策のいくつかのテキスト例とともに提示する「プロンプティング技術」を使用して、追加の訓練なしでタスクを解決できることがわかった[2]。

ファインチューニング

詳細は「ファインチューニング(機械学習)」を参照

ファインチューニング（英: fine-tuning、微調整）とは、事前訓練された既存の言語モデルを、特定のタスク（例: 感情分析、固有表現識別、品詞タグ付け）で（教師ありの）訓練を行うことによって修正する手法である。これは転移学習の一種である。一般的には、言語モデルの最終層と下流タスク（英: downstream tasks）の出力とを接続する新しい重みのセットを導入することになる。言語モデルの元の重みは「凍結」したまま、それらを出力に接続する新しい重み層のみが訓練中に調節されるように構成する。また、元の重みをわずかずつ更新させたり、あるいは以前の凍結された層と一緒に更新されることもある[35]。

プロンプト

「プロンプトエンジニアリング」および「少数ショット学習」も参照

GPT-3によって普及したプロンプトパラダイムでは[4]、解決すべき問題はテキストプロンプト（回答を促す指示）で定式化され、モデルは（推論して）補完を生成することによってそれを解決しなければならない。「少数ショットプロンプト」（英: few-shot prompting）の場合、プロンプトには類似した組（問題、解決）の少数の例が含まれる[2]。たとえば、映画レビューに対する感情をラベル付けする感情分析タスクは、次のような例で回答が促される[4]。

```
レビュー: この映画は気が沈む。
感情: ネガティブ

レビュー: この映画は素晴らしい!
感情:
```

もしモデルが「ポジティブ」と出力すれば、正しくタスクが解決されたことになる[37][41]。一方、「ゼロショットプロンプト」（英: zero-shot prompting）の場合、解決例を提供しない。同じ感情分析タスクに対するゼロショットプロンプトの例は、『映画レビューに関連するセンチメントは「この映画は素晴らしい!」』である[42]。

LLMにおける少数ショットの性能は、NLPタスクで競争力のある結果を達成することが示されており、ときには先行する最先端のファインチューニング手法を凌ぐことさえある。このようなNLPタスクの例としては、翻訳、質問応答、穴埋め、マス埋めパズル、文中の新語検出などがある[41]。優れたプロンプトを作成し、最適化することをプロンプトエンジニアリングと呼ぶ。

インストラクション・チューニング

ショットプロンプトによる対話を促進するために考案されたファインチューニングの一形態である。テキストが入力されると、事前訓練された言語モデルは、訓練に使用したテキストコーパスの分布に一致するような補完を生成する。たとえば、「ハムレットの主要テーマについてエッセイを書いてください」というプロンプトが与えられたとき、単純な言語モデルは「3月17日以降に受け取った提出物には、1日あたり10%の遅延損害金が適用されます」といった（意図しない）補完を出力するかもしれない。インストラクション・チューニングでは、自然言語による命令として定式化された多くのタスクの例と、適切な応答を用いて言語モデルを訓練する。

インストラクション・チューニングでは、さまざまな手法が実践されている。その一例である「自己学習（英: self-instruct）」は、LLMによって生成された事例（人間が作成した少数の初期事例からブートストラップしたもの）の訓練セットで言語モデルをファインチューニングする[43]。

強化学習によるファインチューニング

OpenAIのInstructGPTプロトコルでは、人間が作成したプロンプトと応答の組からなるデータセットによる教師ありファインチューニングと、それに続く、人間のフィードバックによる強化学習（RLHF）を伴っている。この場合、人間の好みを反映したデータセットを用いて報酬関数を教師あり学習し、その後、この報酬モデルを使用した近位方策最適化によってLLM自体を訓練する[44]。

ツールの使用

LLMだけでは解決が難しい、あるいは不可能な問題もある。たとえば、「354 * 139 = 」のような計算式の場合、次のトークンを予測することは困難であり、「What is the time now? It is」（今は何時ですか？ 今は）についてはまったく予測できない。しかし、人が計算機を使って計算し、時計を使って時刻を知るように、LLMも他のプログラムを呼び出して次のトークンを予測することができる。LLMは、「What is the time now? It is {system.time()}」（今何時ですか？ 今は{system.time()}）や、「354 * 139 = {354 * 139}」のようにプログラムコードを生成し、次に別のプログラムインタプリタが生成されたコードを実行してその出力を埋める[45][46]。この基本的な戦略は、生成されたプログラムを複数回試行したり、別のサンプリング戦略を使用して改良することもできる[47]。

一般的に、LLMにツール（道具）を使わせるためには、ツールを使えるようにファインチューニングする必要がある。ツールの数が有限であれば、ファインチューニングは一度で済むかもしれない。オンラインのAPIサービスのようにツールの数が任意に増えるのであれば、APIの仕様書を読み取ってAPIを正しく呼び出せるようにLLMをファインチューニングすることができる[48][49]。

より単純なツールの使用形態として、検索拡張生成（*Retrieval Augmented Generation*、RAG）があり、これはLLMを文書検索を使用して拡張するもので、ときにはベクトルデータベースを使うこともある。クエリが与えられると、文書検索ツールが呼び出され、もっとも関連性が高い文書が取得される（通常、初めにクエリと文書をベクトルで符号化し、次にクエリベクトルにユークリッドノルムで最も近いベクトルを持つ文書を検索する）。その後、LLMは、クエリと取得した文書の両方に基づいて出力を生成する[50]。

エージェント

LLMは言語モデルであり、それ自体は目標を持たないためエージェントではないが、知的エージェントの構成要素として使用することができる。

ReAct（Reason + Act）法は、LLMをプランナーとして使用し、LLMからエージェントを構築するものである。LLMは「考えごとを声に出して言う」よう促される。具体的には、言語モデルに対して、環境のテキスト表現、目標、可能な行動のリスト、および過去の行動と観察の記録が与えられる。LLMは、行動を決める前に1つまたは複数の思考を行い、それが環境内で実行される[51]。LLMプランナーに与えられる環境の言語的記述は、ときには環境を記述した論文のLaTeXコードすら考えられる[52]。

エピソードの終わりに、LLMはそのエピソードの記録が渡され、次のエピソードでより良い成績を出すための「教訓」を考えるように促される。これらの「教訓」は次のエピソードでエージェントに渡される。

モンテカルロ木探索では、LLMをロールアウトのためのヒューリスティクスとして使用することができる。プログラムされた世界モデルが利用できない場合、LLMは世界モデルとして動作するように環境を説明するよう促されることもある[54]。

オープンエンド探索では、LLMを観測値の「興味深さ（interestingness）」のスコアリングに使用し、これを通常の（非LLM）強化学習エージェントを誘導する報酬信号として使用することができる[55]。あるいは、LLMに、カリキュラム学習のために次第に難しくなるタスクを提案させることもできる[56]。LLMプランナーは、個々の行動を出力する代わりに、複雑な行動シーケンスを表す「スキル」や関数を構築することもできる。スキルを保存して後で呼び出すことができるため、プランニングの抽象度を高めることができる[56]。LLMを使用したエージェントは、過去のコンテキストの長期記憶を保持して、この記憶は検索拡張生成と同じ方法で取り出すことができる。このようなエージェントどうしが社会的に相互作用することができる[57]。

圧縮

通常、LLMの訓練では、全精度または半精度の浮動小数点数（float32とfloat16）が使用される。float16は16ビット（つまり2バイト）なので、たとえば10億個のパラメータは2ギガバイトのサイズとなる。典型的な最大級のモデルは1,000億個のパラメータを持ち、ロードするのに200ギガバイトを必要とするため、ほとんどの一般向けコンピュータの能力を超えたものとなる。訓練後の量子化（*Post-training quantization*）は[58]、訓練済みモデルの性能をほとんど維持したまま、パラメーターの精度を下げることで、必要なサイズを削減することを目的としている[59][60]。量子化の最も単純な形は、すべての数値を所定のビット数に切り捨てるだけである。これは、層ごとに異なる量子化コードブックを使用することで改善できる。さらに、パラメータごとにさまざまな精度を適用し、特に重要なパラメータ（外れ値の重み）にはより高い精度を確保することで、さらなる改善をはかることができる[61]。

量子化モデルは通常は凍結され、量子化前のモデルだけがファインチューニングされるが、量子化モデルも引き続きファインチューニングが可能である[62]。

評価

パープレキシティ

言語モデルの性能を表す最も一般的な指標は、所与のテキストコーパスにおける言語モデルのパープレキシティである。パープレキシティは、モデルがデータセットの内容をどれだけうまく予測できるかを示す尺度である。モデルがデータセットに割り当てる尤度（ゆうど）が高いほど、パープレキシティは低くなる。数学的には、パープレキシティは、トークンごとの平均負対数尤度の対数として定義される。

$$\log(\text{Perplexity}) = -\frac{1}{N}\sum_{i=1}^{N}\log(Pr(\text{token}_i|\text{context for token}_i))$$

ここで、N はテキストコーパス内のトークン数であり、「context for token i（トークン i の文脈）」は使用するLLMの種類に依存する。たとえば、LLMが自己回帰型の場合、「context for token i」はトークン i よりも前に現れたテキストの一部である。

れるテストセットに対するパープレキシティによって評価される。このことは、大規模な言語モデルを評価する際に、特に重要な課題となる[35]。言語モデルの訓練は、主にウェブから収集された、より大規模なテキストコーパスが使用されるため、モデルの訓練データに特定のテストセットの一部が誤って含まれてしまう可能性がますます高くなる[41]。

タスク固有のデータセットとベンチマーク

また、言語モデルがより具体的な下流タスクを実行する能力を評価するために、多くのテスト用データセットやベンチマークが開発されている。テストは、一般的な知識、常識的な推論、数学的な問題解決など、さまざまな能力を評価するために設計することができる。

評価用データセットの大区分の1つに、質問と正解の組で構成される質問応答データセットがある。たとえば、『「サンノゼ・シャークスはスタンレーカップで優勝しましたか?」、「いいえ」』のような組である[63]。質問回答タスクでは、モデルのプロンプトに期待される答えを導き出せるテキストが含まれる場合、「明白なもの（オープンブック (en:英語版)）」とみなされる。たとえば、先の質問には、「2016年、シャークスはスタンレーカップ決勝戦に進出し、ピッツバーグ・ペンギンズに敗れた。」という文を含むテキストが追加される可能性がある[63]。そうでない場合、タスクは「(理解する術がなく）説明できないもの（クローズドブック)」とみなされ、モデルは訓練中に獲得した知識を動員する必要がある[64]。一般的な質問回答データセットの例として、TruthfulQA、Web Questions、TriviaQA、SQuADなどがある[64]。

評価用データセットは、テキスト補完の形式をとることもできる。この場合、モデルは、プロンプトを完成させるために最も可能性の高い単語や文章を選択する。たとえば、「アリスはボブと友達だった。アリスは彼女の友人の_____を訪ねた。」のような穴埋め型の設問である[41]。

また、さまざまな評価データセットやタスクを組み合わせた複合ベンチマークも開発されている。たとえば、GLUE、SuperGLUE、MMLU、BIG-bench、HELMなどがある[65][64]。

かつては、評価用データセットの一部を手元に残し、残りの部分で教師ありファインチューニングを行い、その後に結果を報告するのが一般的であった。現在では、事前訓練されたモデルをプロンプティング技術によって直接評価することが一般的になっている。しかし、特定のタスクに対するプロンプトの作成方法、特にプロンプトに付加される解決済みタスクの事例数（nショットプロンプトのn値）については研究者によって異なる。

逆説的に構成された評価

大規模言語モデルの改良が急速に進んでいるため、評価ベンチマークの寿命は短く、最先端のモデルが既存のベンチマークを急速に「飽和」させ、人間の注釈者の能力をも超えてしまう。そのためベンチマークをより難易度が高いタスクで置き換えたり、強化したりする取り組みが行われている[66]。

中には敵対的に構築されたデータセットもあり、人間と比べて既存の言語モデルの性能が異常に低いと思われる特定の問題に重点が置かれている。その一例がTruthfulQAデータセットで、言語モデルが訓練中に繰り返し触れた虚偽を模倣することで不正確な解答をする可能性がある、817問からなる質問応答データセットである。たとえば、LLMは「Can you teach an old dog new tricks?（年老いた犬に新しい芸を教えられますか？)」という質問に対して、「*you can't teach an old dog new tricks*（老犬に新しい芸を仕込むことはできない)」という英語の語法に触れた結果、文字通り真実でないにもかかわらず、「No」と答えるかもしれない[67]。

さらに、AIが多肢選択式テスト（○×式テスト）において、必ずしも実際に訪ねられている設問を理解することなく表面的な問題文の統計的相関を利用して正解を推測し、「カンニング」する「ショートカット学習」と呼ばれるケースもある[68]。

敵対的評価データセットのもう一つの例は、Swagとその後継のHellaSwagである。これは、文章を完成させるためにいくつかの選択肢から一つを選択しなければならない問題を集めたものである。不正解の選択肢は、言語モデルからサンプリングし、一連の分類器でフィルタリングすることで作成された。その結果、人

かった。たとえば、次のようなものである。

> フィットネスセンターの看板が見える。そして、エクササイズボールに座ったり横たわりながら、カメラに向かって話しかける男性が見える。その男性は、...
>
> a) ボールの上を走ったり降りたりして、運動の効果を効率的にする方法を実演している。
>
> b) すべての腕と脚を動かしてたくさんの筋肉をつけている。
>
> c) 次にボールを投げ、グラフィックや生け垣の刈り込みの実演を見る。
>
> d) ボールの上で腹筋運動をしながら話をしている[69]。

BERTは最も可能性の高い補完としてb)を選択したが、正解はd)である[69]。

解釈

大規模言語モデルは、それ自体が「ブラックボックス」であり、どのようにして言語タスクを実行できるのかは明らかではない。しかし、LLMがどのように機能するかを理解するためのいくつかの方法がある。

機械的解釈可能性 は、LLMによって実行される推論を近似する記号アルゴリズムを発見することにより、LLMをリバースエンジニアリングすることを目的としている。オセロGPT（Othello-GPT）はその一例で、オセロの正当な手を予測するように小規模なTransformerが訓練された。その結果、オセロ盤の線形表現が存在し、この表現を変更することで、予測される正当なオセロの手が正しい方向に変化することがわかった[70][71]。別の例では、著者はモジュラ算術加算に対して小規模なTransformerを訓練し、得られたモデルをリバースエンジニアリングしたところ、離散フーリエ変換を使用していることがわかった[72]。

別の例では、小規模なTransformerをKarelプログラムに対して訓練している。オセロGPTの例と同様に、Karelプログラムのセマンティクスには線形表現があり、その表現を修正すると出力が正しく変更される。このモデルはまた、訓練セット内のプログラムよりも平均して短く、正しいプログラムを生成した[73]。

理解力と知性

2022年の調査で、（チューニングされていない）LLMが、「自然言語を何らかの自明でない意味で理解できる（ことがある）か」という問いに対して、自然言語処理研究者の意見は真っ二つに分かれた[68]。「LLMは理解力を持つ」派の支持者は、数学的推論のようないくつかのLLMの能力は、特定の概念を「理解」する能力を意味すると考えている。マイクロソフトのチームは、2023年に、GPT-4は「数学、コーディング、視覚、医学、法律、心理学などにまたがる斬新で難しいタスクを解決できる」とし、GPT-4は「汎用人工知能システムの初期バージョン（しかしまだ未完成）とみなすのが妥当だろう」と主張し、「ソフトウェア工学の受験者の試験に合格するシステムが、本当の意味で知的ではないと言えるだろうか？[74][75]」と述べた。LLMを「地球外生命の知能」と呼ぶ研究者もいる[76][77]。たとえば、ConjectureのCEOであるコナー・リーヒーは、チューニングされていないLLMを、まるで得体の知れないエイリアン「ショゴス」のようだと見なし、RLHFチューニングがLLMの内部構造を覆い隠す「見せかけの笑顔」を作り出すと考えている。『あまり無理をしなければ、笑顔のままだ。しかし（予期せぬ）プロンプトを与えると突然、狂気、奇妙な思考過程、そして明らかに人間ではない理解といった巨大な裏の顔を覗かせる』[78][79]。

対照的に、「LLMは理解力を欠く」派の支持者の中には、既存のLLMは「既存の文章を単に練り直し、組み替えているだけ」であると考えたり[77]、既存のLLMが予測能力、推論能力、主体性、説明可能性において依然として欠点を抱えていることを指摘したりする人もいる[68]。たとえば、GPT-4は計画やリアルタイム学習においてもっともな欠陥がある[75]。生成的LLMは、訓練データでは正当化されないような事実を自信をもって主張することが観察されており、この現象は「ハルシネーション（幻覚）」として知られている[80]。

相違は、自然の叡智に基づく私たちの古い考え方が十分ではないことを示唆している」と主張している[68]。

より広範囲な影響

2023年、科学雑誌 *Nature Biomedical Engineering* は、人間が書いたテキストと大規模言語モデルによって作成されたテキストを「正確に区別することはもはや不可能」であり、「汎用大規模言語モデルが急速に普及することはほぼ確実である。いずれは多くの業界を変えてゆくだろう。」と結論づけた[81]。ゴールドマン・サックスは2023年、言語生成AIは今後10年間で世界のGDPを7％増加させ、全世界で3億人の雇用を自動化にさらす可能性があると示唆した[82][83]。一部の投稿者は、偶発的または意図的な誤情報の作成や、その他の悪用に対して懸念を表明した[84]。たとえば、大規模言語モデルが利用できるようになると、バイオ

スフェルトは、LLM開発者は、病原体の作成や改良に関する論文を訓練データから除外すべきだと提案している[85]。

大規模言語モデルの一覧

大規模言語モデルの一覧 [表示]

名称	公開日[注釈 1]	開発者	パラメータ数[注釈 2]	コーパスサイズ	ライセンス[注釈 3]	注記
BERT	2018年	Google	3.4億[86]	33億語[86]	Apache 2.0[87]	初期の影響力のある言語モデルだが[2]、エンコーダのみで、プロンプトや生成的モデルを想定していない[88]
XLNet	2019年	Google	~340 million[89]	33 billion words		An alternative to BERT; designed as encoder-only[90][91]
GPT-2	2019年	OpenAI	15億[92]	40GB[93] (~100億トークン) [94]	MIT[95]	Transformerアーキテクチャに基づく汎用モデル
GPT-3	2020年	OpenAI	1,750億[37]	4,990億トークン[94]	public web API	GPT-3のファインチューニング版はGPT-3.5と呼ばれ、2022年にChatGPTというWebインターフェースを通じて一般公開された[96]。
GPT-Neo	2021年3月	EleutherAI	27億[97]	825 GiB[98]	MIT[99]	EleutherAIがリリースした無料のGPT-3代替シリーズのうち最初のもの。GPT-Neoは、いくつかのベンチマークで同サイズのGPT-3モデルよりも優れていたが、最大のGPT-3よりは大幅に劣っていた[99]。

名称	公開日[注釈 1]	開発者	パラメータ数[注釈 2]	コーパスサイズ	ライセンス[注釈 3]	注記
GPT-J	2021年6月	EleutherAI	60億[100]	825 GiB[98]	Apache 2.0	GPT-3方式の言語モデル
Megatron-Turing NLG	2021年10月[101]	Microsoft and Nvidia	5,300億[102]	3,386億トークン[102]	Restricted web access	標準的なアーキテクチャだが、スーパーコンピューティング・クラスターで訓練された
Ernie 3.0 Titan	2021年12月	Baidu	2,600億[103]	4 Tb	プロプライエタリ	中国語版LLM。Ernie Botはこのモデルに基づく。
Claude[104]	2021年12月	Anthropic	520億[105]	4,000億トークン[105]	Closed beta	会話で望ましい動作をするようにファインチューニングされた[106]
GLaM (Generalist Language Model)	2021年12月	Google	1.2兆[30]	1.6兆トークン [30]	プロプライエタリ	GPT-3と比較して、訓練費用は高いが、推論費用は安い、スパース混合エキスパートモデル
Gopher	2021年12月	DeepMind	2,800億[107]	3,000億トークン[108]	プロプライエタリ	
LaMDA (Language Models for Dialog Applications)	2022年1月	Google	1,370億[109]	1.56T語,[109] 1,680億トークン[108]	プロプライエタリ	会話での応答生成に特化し、Google Bardチャットボットで使用されている
GPT-NeoX	2022年2月	EleutherAI	200億[110]	825 GiB[98]	Apache 2.0	Megatronアーキテクチャに基づく
Chinchilla	2022年3月	DeepMind	700億[111]	1.4兆 トークン [111][108]	プロプライエタリ	より多くのデータで訓練されたパラメータ削減モデル。Sparrowボットで使用された。
PaLM (Pathways Language Model)	2022年4月	Google	5,400億[112]	7,680億トークン[111]	プロプライエタリ	モデルスケールの実用的な限界に到達することを目指した

名称	公開日[注釈 1]	開発者	パラメータ数[注釈 2]	コーパスサイズ	ライセンス[注釈 3]	注記
OPT (Open Pretrained Transformer)	2022年5月	Meta	1,750億[113]	1,800億トークン[114]	Non-commercial research[注釈 4]	GPT-3アーキテクチャにMegatronから改作を加えたもの
YaLM 100B	2022年6月	Yandex	1,000億[115]	1.7TB[115]	Apache 2.0	MicrosoftのMegatron-LMに基づく英露モデル
Minerva	2022年6月	Google	5,400億[116]	385億トークン[注釈 5][116]	プロプライエタリ	数学的および科学的な問題を段階的な推論によって解くために訓練されたLLMである[117]。Minervaは、PaLMモデルに基にさらに数学的および科学的データで訓練されている。
BLOOM	2022年7月	Large collaboration led by Hugging Face	1,750億[118]	3,500億トークン (1.6TB)[119]	Responsible AI	基本的にはGPT-3だが、多言語コーパスでトレーニングされている（プログラミング言語を除いて、30%は英語）。
Galactica	2022年11月	Meta	1,200億	1,060億トークン[120]	CC-BY-NC-4.0	科学的なテキストや方法の訓練を受けている
AlexaTM (Teacher Models)	2022年11月	Amazon	200億[121]	1.3兆[122]	public web API[123]	双方向のシーケンスからシーケンスへのアーキテクチャ
LLaMA (Large Language Model Meta AI)	2023年2月	Meta	650億[124]	1.4兆[124]	Non-commercial research[注釈 6]	20言語の大規模コーパスで訓練し、より少ないパラメータでの性能向上を目指す[124]。スタンフォード大学の研究者は、Alpacaと呼ばれるLLaMAの

名称	公開日[注釈 1]	開発者	パラメータ数[注釈 2]	コーパスサイズ	ライセンス[注釈 3]	注記
						重みに基づいて微調整されたモデルを訓練した[125]。
GPT-4	2023年3月	OpenAI	非公開[注釈 7]	非公開	public web API	ChatGPT Plusユーザが利用でき、いくつかの製品で使用されている
Cerebras-GPT	2023年3月	Cerebras	130億[127]		Apache 2.0	Chinchilla方式で訓練された
Falcon	2023年3月	Technology Innovation Institute	1800億[128]	3.5兆トークン[128]	Falcon 180B TII License (Apache 2.0ベース)[128]	モデルはGPT-3の75%、Chinchillaの40%、PaLM-62Bの80%の訓練計算量で済むとされる
BloombergGPT	2023年3月	Bloomberg L.P.	500億	3,630億トークン[注釈 8][129]	プロプライエタリ	独自ソースによる財務データで訓練され、「一般的なLLMベンチマークでの性能を犠牲にすることなく、財務タスクで既存モデルを大幅に上回る」とされる
PanGu-Σ	2023年3月	Huawei	1.085兆	3,290億トークン[130]	プロプライエタリ	
OpenAssistant[131]	2023年3月	LAION	17 billion	1.5 trillion tokens	Apache 2.0	Trained on crowdsourced open data
PaLM 2 (Pathways Language Model 2)	2023年5月	Google	340 billion[132]	3.6 trillion tokens[132]	Proprietary	Used in Bard chatbot.[133]
RedPajama	2023年5月	Together Computer他	7 billion	1.2兆	Apache 2.0	LLaMAベース
MPT	2023年5月	MosaicML Foundation	7 billion	1兆	Apache 2.0	
Mistral	2023年9月	Mistral AI	7 billion	*?*	Apache 2.0	

脚注

注釈

1. ^ モデルのアーキテクチャを説明する文書が最初に公開された日。
2. ^ 多くの場合、研究者はサイズの異なる複数のモデルを公開または報告する。こうした場合、ここでは一番大きなモデルのサイズを記載している。
3. ^ これは、事前学習されたモデルウェイトのライセンスである。たいていの場合、訓練コード自体はオープンソースであるか、簡単に複製することができる。
4. ^ 66Bを含めた小規模モデルは一般に公開されており、175Bのモデルはリクエストに応じて入手可能である。
5. ^ 数学的な内容でフィルタリングされたウェブページおよびarXivプレプリントサーバーに投稿された論文からの385億トークン。
6. ^ Facebookのライセンスと配布スキームにより、モデルへのアクセスは承認された研究者にが制限されていたが、モデルウェイトが流出して広く利用されるようになった。
7. ^ テクニカルレポートに述べられているように『GPT-4のような大規模モデルの市場競争と安全性への影響の両方を考慮して、このレポートには、アーキテクチャ（モデルサイズを含む）、ハードウェア、訓練計算環境、データセット構築、トレーニング方法に関する詳細は含まれていない[126]。』
8. ^ ブルームバーグのデータソースからの3,630億トークンと、汎用データセットからの3,450億トークンのデータセット

出典

1. ^ Goled, Shraddha (2021年5月7日). “Self-Supervised Learning Vs Semi-Supervised Learning: How They Differ (https://analyticsindiamag.com/self-supervised-learning-vs-semi-supervised-learning-how-they-differ/)”. *Analytics India Magazine*. 2023年5月13日閲覧。
2. ^ ***a*** ***b*** ***c*** ***d*** ***e*** ***f*** ***g*** Manning, Christopher D. (2022). “Human Language Understanding & Reasoning” (https://www.amacad.org/publication/human-language-understanding-reasoning). *Daedalus* **151** (2): 127–138. doi:10.1162/daed_a_01905 (https://doi.org/10.1162%2Fdaed_a_01905).
3. ^ “Responsible AI - Week 3 (https://www.coursera.org/lecture/generative-ai-with-llms/responsible-ai-moMCz)”. *Coursera*. 2023年7月23日閲覧。
4. ^ ***a*** ***b*** ***c*** ***d*** ***e*** ***f*** Wei, Jason; Tay, Yi; Bommasani, Rishi; Raffel, Colin; Zoph, Barret; Borgeaud, Sebastian; Yogatama, Dani; Bosma, Maarten et al. (31 August 2022). “Emergent Abilities of Large Language Models” (https://openreview.net/forum?id=yzkSU5zdwD) (英語). *Transactions on Machine Learning Research*. ISSN 2835-8856 (https://search.worldcat.org/ja/search?fq=x0:jrnl&q=n2:2835-8856).
5. ^ ***a*** ***b*** Bowman, Samuel R.. *Eight Things to Know about Large Language Models* (https://cims.nyu.edu/~sbowman/eightthings.pdf).
6. ^ Schaeffer, Rylan; Miranda, Brando; Koyejo, Sanmi (2023). *Are Emergent Abilities of Large Language Models a Mirage?* (https://arxiv.org/abs/2304.15004). doi:10.48550/ARXIV.2304.15004 (https://doi.org/10.48550%2FARXIV.2304.15004).

7. ^ Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta , Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, Yonghui Wu. "PaLM 2 Technical Report". arXiv:2305.10403 (https://arxiv.org/abs/2305.10403)。
8. ^ “Papers with Code - MassiveText Dataset (https://paperswithcode.com/dataset/massivetext)” (英語). *paperswithcode.com*. 2023年4月26日閲覧。
9. ^ Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, Gideon Mann. "BloombergGPT: A Large Language Model for Finance". arXiv:2303.17564 (https://arxiv.org/abs/2303.17564)。
10. ^ Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, Matt Gardner. "Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus". arXiv:2104.08758 (https://arxiv.org/abs/2104.08758)。
11. ^ Villalobos, Pablo; Sevilla, Jaime; Heim, Lennart; Besiroglu, Tamay; Hobbhahn, Marius; Ho, Anson (25 October 2022). "Will we run out of data? An analysis of the limits of scaling datasets in Machine Learning". arXiv:2211.04325 (https://arxiv.org/abs/2211.04325) [cs.LG (https://arxiv.org/archive/cs.LG)]。
12. ^ 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. "Language Models are Few-Shot Learners". arXiv:2005.14165 (https://arxiv.org/abs/2005.14165)。

13. ^ Hoffmann, Jordan; Borgeaud, Sebastian; Mensch, Arthur; Buchatskaya, Elena; Cai, Trevor; Rutherford, Eliza; Casas, Diego de Las; Hendricks, Lisa Anne et al. (2022-03-29). "Training Compute-Optimal Large Language Models" (http://arxiv.org/abs/2203.15556). *arXiv:2203.15556 [cs]*.
14. ^ **a** **b** Kaplan, Jared; McCandlish, Sam; Henighan, Tom; Brown, Tom B.; Chess, Benjamin; Child, Rewon; Gray, Scott; Radford, Alec et al. (2020). "Scaling Laws for Neural Language Models". *CoRR* **abs/2001.08361**. arXiv:2001.08361.
15. ^ Caballero, Ethan; Gupta, Kshitij; Rish, Irina; Krueger, David (2022). Broken Neural Scaling Laws. International Conference on Learning Representations (ICLR), 2023.
16. ^ Ornes, Stephen (2023年3月16日). "The Unpredictable Abilities Emerging From Large AI Models (https://www.quantamagazine.org/the-unpredictable-abilities-emerging-from-large-ai-models-20230316/)". *Quanta Magazine*. 2023年5月13日閲覧。
17. ^ Schaeffer, Rylan; Miranda, Brando; Koyejo, Sanmi (1 April 2023). "Are Emergent Abilities of Large Language Models a Mirage?". arXiv:2304.15004 (https://arxiv.org/abs/2304.15004) [cs.AI (https://arxiv.org/archive/cs.AI)]。
18. ^ Elman, Jeffrey L. (March 1990). "Finding Structure in Time" (http://doi.wiley.com/10.1207/s15516709cog1402_1) (英語). *Cognitive Science* **14** (2): 179–211. doi:10.1207/s15516709cog1402_1 (https://doi.org/10.1207%2Fs15516709cog1402_1).
19. ^ Sutskever, Ilya; Vinyals, Oriol; Le, Quoc V (2014). "Sequence to Sequence Learning with Neural Networks" (https://proceedings.neurips.cc/paper_files/paper/2014/hash/a14ac55a4f27472c5d894ec1c3c743d2-Abstract.html). *Advances in Neural Information Processing Systems* (Curran Associates, Inc.) **27**.
20. ^ Cho, Kyunghyun; van Merriënboer, Bart; Bahdanau, Dzmitry; Bengio, Yoshua (2014). "On the Properties of Neural Machine Translation: Encoder–Decoder Approaches" (https://doi.org/10.3115/v1/w14-4012). *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation* (Stroudsburg, PA, USA: Association for Computational Linguistics). doi:10.3115/v1/w14-4012 (https://doi.org/10.3115%2Fv1%2Fw14-4012).
21. ^ **a** **b** Bahdanau, Dzmitry; Cho, Kyunghyun; Bengio, Yoshua (2014-09-01). *Neural Machine Translation by Jointly Learning to Align and Translate* (https://ui.adsabs.harvard.edu/abs/2014arXiv1409.0473B).
22. ^ Lewis-Kraus, Gideon (2016年12月14日). "The Great A.I. Awakening" (https://web.archive.org/web/20230524052626/https://www.nytimes.com/2016/12/14/magazine/the-great-ai-awakening.html) (英語). *The New York Times*. ISSN 0362-4331 (https://search.worldcat.org/ja/search?fq=x0:jrnl&q=n2:0362-4331). オリジナル (https://www.nytimes.com/2016/12/14/magazine/the-great-ai-awakening.html)の2023年5月24日時点におけるアーカイブ。 2023年6月22日閲覧。
23. ^ Wu, Yonghui; Schuster, Mike; Chen, Zhifeng; Le, Quoc V.; Norouzi, Mohammad; Macherey, Wolfgang; Krikun, Maxim; Cao, Yuan et al. (2016-09-01). *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation* (https://ui.adsabs.harvard.edu/abs/2016arXiv160908144W).
24. ^ Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N; Kaiser, Łukasz; Polosukhin, Illia (2017). "Attention is All you Need" (https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html). *Advances in Neural Information Processing Systems* (Curran Associates, Inc.) **30**.

25. ^ Devlin, Jacob; Chang, Ming-Wei; Lee, Kenton; Toutanova, Kristina (11 October 2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". arXiv:1810.04805v2 (https://arxiv.org/abs/1810.04805v2) [cs.CL (https://arxiv.org/archive/cs.CL)]。
26. ^ "Improving language understanding with unsupervised learning (https://web.archive.org/web/20230318210736/https://openai.com/research/language-unsupervised)" (英語). *openai.com* (2018年6月11日). 2023年3月18日時点のオリジナル (https://openai.com/research/language-unsupervised)よりアーカイブ。2023年3月18日閲覧。
27. ^ *finetune-transformer-lm* (https://github.com/openai/finetune-transformer-lm), OpenAI, (June 11, 2018) 2023年5月1日閲覧。
28. ^ Shazeer, Noam; Mirhoseini, Azalia; Maziarz, Krzysztof; Davis, Andy; Le, Quoc; Hinton, Geoffrey; Dean, Jeff (2017-01-01). *Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer* (https://ui.adsabs.harvard.edu/abs/2017arXiv170106538S).
29. ^ Lepikhin, Dmitry; Lee, HyoukJoong; Xu, Yuanzhong; Chen, Dehao; Firat, Orhan; Huang, Yanping; Krikun, Maxim; Shazeer, Noam et al. (2021-01-12) (英語). *GShard: Scaling Giant Models with Conditional Computation and Automatic Sharding* (https://openreview.net/forum?id=qrwe7XHTmYb).
30. ^ ***a*** ***b*** ***c*** "More Efficient In-Context Learning with GLaM (https://ai.googleblog.com/2021/12/more-efficient-in-context-learning-with.html)" (英語). *ai.googleblog.com* (2021年12月9日). 2023年3月9日閲覧。
31. ^ "OpenAI API (https://web.archive.org/web/20230423211308/https://platform.openai.com/tokenizer)" (英語). *platform.openai.com*. 2023年4月23日時点のオリジナル (https://platform.openai.com/)よりアーカイブ。2023年4月30日閲覧。
32. ^ "OpenAI API (https://archive.is/k6LNQ)" (英語). *platform.openai.com*. 2023年6月16日時点のオリジナル (https://platform.openai.com/)よりアーカイブ。2023年6月20日閲覧。
33. ^ LeCun, Yann (2023年4月28日). "A survey of LLMs with a practical guide and evolutionary tree (https://web.archive.org/web/20230623012310/https://twitter.com/ylecun/status/1651762787373428736?lang=en)" (英語). *Twitter*. 2023年6月23日時点のオリジナル (https://twitter.com/ylecun/status/1651762787373428736?lang=en)よりアーカイブ。2023年6月23日閲覧。
34. ^ Zaib, Munazza; Sheng, Quan Z.; Emma Zhang, Wei (4 February 2020). "A Short Survey of Pre-trained Language Models for Conversational AI-A New Age in NLP" (https://www.researchgate.net/publication/338931711). *Proceedings of the Australasian Computer Science Week Multiconference*: 1–4. arXiv:2104.10810. doi:10.1145/3373017.3373028 (https://doi.org/10.1145%2F3373017.3373028). ISBN 9781450376976.
35. ^ ***a*** ***b*** ***c*** ***d*** ***e*** ***f*** Jurafsky, Dan; Martin, James H. (7 January 2023). *Speech and Language Processing* (https://web.stanford.edu/~jurafsky/slp3/ed3book_jan72023.pdf) (3rd edition draft ed.) 2022年5月24日閲覧。
36. ^ Zhu, Yukun; Kiros, Ryan; Zemel, Rich; Salakhutdinov, Ruslan; Urtasun, Raquel; Torralba, Antonio; Fidler, Sanja (December 2015). "Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books" (https://www.cv-foundation.org/openaccess/content_iccv_2015/papers/Zhu_Aligning_Books_and_ICCV_2015_paper.pdf). *2015 IEEE International Conference on Computer Vision (ICCV)*: 19–27. arXiv:1506.06724. doi:10.1109/ICCV.2015.11 (https://doi.org/10.1109%2FICCV.2015.11). ISBN 978-1-4673-8391-2 2023年4月11日閲覧。.

37. ^ Wiggers, Kyle (2022年4月28日). "The emerging types of language models and why they matter (https://techcrunch.com/2022/04/28/the-emerging-types-of-language-models-and-why-they-matter/)". *TechCrunch*. 2023年4月28日閲覧。
38. ^ Sharir, Or, Barak Peleg, and Yoav Shoham. "The cost of training nlp models: A concise overview." arXiv preprint arXiv:2004.08900 (2020).
39. ^ Biderman, Stella; Schoelkopf, Hailey; Anthony, Quentin; Bradley, Herbie; Khan, Mohammad Aflah; Purohit, Shivanshu; Prashanth, USVSN Sai (April 2023). "Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling". arXiv:2304.01373 (https://arxiv.org/abs/2304.01373) [cs.CL (https://arxiv.org/archive/cs.CL)]。
40. ^ Vincent, James (2023年4月3日). "AI is entering an era of corporate control" (https://www.theverge.com/23667752/ai-progress-2023-report-stanford-corporate-control). *The Verge* 2023年6月19日閲覧。
41. ^ ***a*** ***b*** ***c*** ***d*** Brown, Tom B.; Mann, Benjamin; Ryder, Nick; Subbiah, Melanie; Kaplan, Jared; Dhariwal, Prafulla; Neelakantan, Arvind; Shyam, Pranav et al. (Dec 2020). Larochelle, H.; Ranzato, M.; Hadsell, R. et al.. eds. "Language Models are Few-Shot Learners" (https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf). *Advances in Neural Information Processing Systems* (Curran Associates, Inc.) **33**: 1877–1901.
42. ^ "Introducing FLAN: More generalizable Language Models with Instruction Fine-Tuning (https://ai.googleblog.com/2021/10/introducing-flan-more-generalizable.html)". *Google Research* (2021年10月6日). 2024年4月28日閲覧。
43. ^ Wang, Yizhong; Kordi, Yeganeh; Mishra, Swaroop; Liu, Alisa; Smith, Noah A.; Khashabi, Daniel; Hajishirzi, Hannaneh (2022). "Self-Instruct: Aligning Language Model with Self Generated Instructions". arXiv:2212.10560 (https://arxiv.org/abs/2212.10560) [cs.CL (https://arxiv.org/archive/cs.CL)]。
44. ^ Ouyang, Long; Wu, Jeff; Jiang, Xu; Almeida, Diogo; Wainwright, Carroll L.; Mishkin, Pamela; Zhang, Chong; Agarwal, Sandhini; Slama, Katarina; Ray, Alex; Schulman, John; Hilton, Jacob; Kelton, Fraser; Miller, Luke; Simens, Maddie; Askell, Amanda; Welinder, Peter; Christiano, Paul; Leike, Jan; Lowe, Ryan (2022). "Training language models to follow instructions with human feedback". arXiv:2203.02155 (https://arxiv.org/abs/2203.02155) [cs.CL (https://arxiv.org/archive/cs.CL)]。
45. ^ Gao, Luyu; Madaan, Aman; Zhou, Shuyan; Alon, Uri; Liu, Pengfei; Yang, Yiming; Callan, Jamie; Neubig, Graham (1 November 2022). "PAL: Program-aided Language Models". arXiv:2211.10435 (https://arxiv.org/abs/2211.10435) [cs.CL (https://arxiv.org/archive/cs.CL)]。
46. ^ "PAL: Program-aided Language Models (https://reasonwithpal.com/)". *reasonwithpal.com*. 2023年6月12日閲覧。
47. ^ Paranjape, Bhargavi; Lundberg, Scott; Singh, Sameer; Hajishirzi, Hannaneh; Zettlemoyer, Luke; Tulio Ribeiro, Marco (1 March 2023). "ART: Automatic multi-step reasoning and tool-use for large language models". arXiv:2303.09014 (https://arxiv.org/abs/2303.09014) [cs.CL (https://arxiv.org/archive/cs.CL)]。
48. ^ Liang, Yaobo; Wu, Chenfei; Song, Ting; Wu, Wenshan; Xia, Yan; Liu, Yu; Ou, Yang; Lu, Shuai; Ji, Lei; Mao, Shaoguang; Wang, Yun; Shou, Linjun; Gong, Ming; Duan, Nan (1 March 2023). "TaskMatrix.AI: Completing Tasks by Connecting Foundation Models with Millions of APIs". arXiv:2303.16434 (https://arxiv.org/abs/2303.16434) [cs.AI (https://arxiv.org/archive/cs.AI)]。

49. ^ Patil, Shishir G.; Zhang, Tianjun; Wang, Xin; Gonzalez, Joseph E. (2023-05-01). *Gorilla: Large Language Model Connected with Massive APIs* (https://ui.adsabs.harvard.edu/abs/2023arXiv230515334P).
50. ^ Lewis, Patrick; Perez, Ethan; Piktus, Aleksandra; Petroni, Fabio; Karpukhin, Vladimir; Goyal, Naman; Küttler, Heinrich; Lewis, Mike et al. (2020). "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks" (https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html). *Advances in Neural Information Processing Systems* (Curran Associates, Inc.) **33**: 9459–9474. arXiv:2005.11401.
51. ^ Yao, Shunyu; Zhao, Jeffrey; Yu, Dian; Du, Nan; Shafran, Izhak; Narasimhan, Karthik; Cao, Yuan (1 October 2022). "ReAct: Synergizing Reasoning and Acting in Language Models". arXiv:2210.03629 (https://arxiv.org/abs/2210.03629) [cs.CL (https://arxiv.org/archive/cs.CL)]。
52. ^ Wu, Yue; Prabhumoye, Shrimai; Min, So Yeon (24 May 2023). "SPRING: GPT-4 Outperforms RL Algorithms by Studying Papers and Reasoning". arXiv:2305.15486 (https://arxiv.org/abs/2305.15486) [cs.AI (https://arxiv.org/archive/cs.AI)]。
53. ^ Shinn, Noah; Cassano, Federico; Labash, Beck; Gopinath, Ashwin; Narasimhan, Karthik; Yao, Shunyu (2023-03-01). *Reflexion: Language Agents with Verbal Reinforcement Learning* (https://ui.adsabs.harvard.edu/abs/2023arXiv230311366S).
54. ^ Hao, Shibo; Gu, Yi; Ma, Haodi; Jiahua Hong, Joshua; Wang, Zhen; Zhe Wang, Daisy; Hu, Zhiting (1 May 2023). "Reasoning with Language Model is Planning with World Model". arXiv:2305.14992 (https://arxiv.org/abs/2305.14992) [cs.CL (https://arxiv.org/archive/cs.CL)]。
55. ^ Zhang, Jenny; Lehman, Joel; Stanley, Kenneth; Clune, Jeff (2 June 2023). "OMNI: Open-endedness via Models of human Notions of Interestingness". arXiv:2306.01711 (https://arxiv.org/abs/2306.01711) [cs.AI (https://arxiv.org/archive/cs.AI)]。
56. ^ ***a*** ***b*** "Voyager | An Open-Ended Embodied Agent with Large Language Models (https://voyager.minedojo.org/)". *voyager.minedojo.org*. 2023年6月9日閲覧。
57. ^ Park, Joon Sung; O'Brien, Joseph C.; Cai, Carrie J.; Ringel Morris, Meredith; Liang, Percy; Bernstein, Michael S. (2023-04-01). *Generative Agents: Interactive Simulacra of Human Behavior* (https://ui.adsabs.harvard.edu/abs/2023arXiv230403442P).
58. ^ Nagel, Markus; Amjad, Rana Ali; Baalen, Mart Van; Louizos, Christos; Blankevoort, Tijmen (2020-11-21). "Up or Down? Adaptive Rounding for Post-Training Quantization" (https://proceedings.mlr.press/v119/nagel20a.html) (英語). *Proceedings of the 37th International Conference on Machine Learning* (PMLR): 7197–7206.
59. ^ Polino, Antonio; Pascanu, Razvan; Alistarh, Dan (1 February 2018). "Model compression via distillation and quantization". arXiv:1802.05668 (https://arxiv.org/abs/1802.05668) [cs.NE (https://arxiv.org/archive/cs.NE)]。
60. ^ Frantar, Elias; Ashkboos, Saleh; Hoefler, Torsten; Alistarh, Dan (1 October 2022). "GPTQ: Accurate Post-Training Quantization for Generative Pre-trained Transformers". arXiv:2210.17323 (https://arxiv.org/abs/2210.17323) [cs.LG (https://arxiv.org/archive/cs.LG)]。
61. ^ Dettmers, Tim; Svirschevski, Ruslan; Egiazarian, Vage; Kuznedelev, Denis; Frantar, Elias; Ashkboos, Saleh; Borzunov, Alexander; Hoefler, Torsten; Alistarh, Dan (1 June 2023). "SpQR: A Sparse-Quantized Representation for Near-Lossless LLM Weight Compression". arXiv:2306.03078 (https://arxiv.org/abs/2306.03078) [cs.CL (https://arxiv.org/archive/cs.CL)]。

62. ^ Dettmers, Tim; Pagnoni, Artidoro; Holtzman, Ari; Zettlemoyer, Luke (1 May 2023). "QLoRA: Efficient Finetuning of Quantized LLMs". arXiv:2305.14314 (https://arxiv.org/abs/2305.14314) [cs.LG (https://arxiv.org/archive/cs.LG)]。
63. ^ **a** **b** Clark, Christopher; Lee, Kenton; Chang, Ming-Wei; Kwiatkowski, Tom; Collins, Michael; Toutanova, Kristina (2019). "BoolQ: Exploring the Surprising Difficulty of Natural Yes/No Questions". arXiv:1905.10044 (https://arxiv.org/abs/1905.10044) [cs.CL (https://arxiv.org/archive/cs.CL)]。
64. ^ **a** **b** **c** Wayne Xin Zhao; Zhou, Kun; Li, Junyi; Tang, Tianyi; Wang, Xiaolei; Hou, Yupeng; Min, Yingqian; Zhang, Beichen; Zhang, Junjie; Dong, Zican; Du, Yifan; Yang, Chen; Chen, Yushuo; Chen, Zhipeng; Jiang, Jinhao; Ren, Ruiyang; Li, Yifan; Tang, Xinyu; Liu, Zikang; Liu, Peiyu; Nie, Jian-Yun; Wen, Ji-Rong (2023). "A Survey of Large Language Models". arXiv:2303.18223 (https://arxiv.org/abs/2303.18223) [cs.CL (https://arxiv.org/archive/cs.CL)]。
65. **^** Huyen, Chip (2019年10月18日). "Evaluation Metrics for Language Modeling (https://thegradient.pub/understanding-evaluation-metrics-for-language-models/)". *The Gradient*. 2024年4月28日閲覧。
66. **^** Srivastava, Aarohi; et al. (2022). "Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models". arXiv:2206.04615 (https://arxiv.org/abs/2206.04615) [cs.CL (https://arxiv.org/archive/cs.CL)]。
67. **^** Lin, Stephanie; Hilton, Jacob; Evans, Owain (2021). "TruthfulQA: Measuring How Models Mimic Human Falsehoods". arXiv:2109.07958 (https://arxiv.org/abs/2109.07958) [cs.CL (https://arxiv.org/archive/cs.CL)]。
68. ^ **a** **b** **c** **d** Mitchell, Melanie; Krakauer, David C. (28 March 2023). "The debate over understanding in AI's large language models" (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10068812/). *Proceedings of the National Academy of Sciences* **120** (13): e2215907120. arXiv:2210.13966. Bibcode: 2023PNAS..12015907M (https://ui.adsabs.harvard.edu/abs/2023PNAS..12015907M/abstract). doi:10.1073/pnas.2215907120 (https://doi.org/10.1073%2Fpnas.2215907120). PMC 10068812 (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10068812/). PMID 36943882 (https://pubmed.ncbi.nlm.nih.gov/36943882/).
69. ^ **a** **b** Zellers, Rowan; Holtzman, Ari; Bisk, Yonatan; Farhadi, Ali; Choi, Yejin (2019). "HellaSwag: Can a Machine Really Finish Your Sentence?". arXiv:1905.07830 (https://arxiv.org/abs/1905.07830) [cs.CL (https://arxiv.org/archive/cs.CL)]。
70. **^** Li, Kenneth; Hopkins, Aspen K.; Bau, David; Viégas, Fernanda; Pfister, Hanspeter; Wattenberg, Martin (1 October 2022). "Emergent World Representations: Exploring a Sequence Model Trained on a Synthetic Task". arXiv:2210.13382 (https://arxiv.org/abs/2210.13382) [cs.LG (https://arxiv.org/archive/cs.LG)]。
71. **^** "Large Language Model: world models or surface statistics? (https://thegradient.pub/othello/)" (英語). *The Gradient* (2023年1月21日). 2023年6月12日閲覧。
72. **^** Nanda, Neel; Chan, Lawrence; Lieberum, Tom; Smith, Jess; Steinhardt, Jacob (1 January 2023). "Progress measures for grokking via mechanistic interpretability". arXiv:2301.05217 (https://arxiv.org/abs/2301.05217) [cs.LG (https://arxiv.org/archive/cs.LG)]。
73. **^** Jin, Charles; Rinard, Martin (1 May 2023). "Evidence of Meaning in Language Models Trained on Programs". arXiv:2305.11169 (https://arxiv.org/abs/2305.11169) [cs.LG (https://arxiv.org/archive/cs.LG)]。

74. ^ Metz, Cade (2023年5月16日). "Microsoft Says New A.I. Shows Signs of Human Reasoning" (https://www.nytimes.com/2023/05/16/technology/microsoft-ai-human-reasoning.html). *The New York Times*
75. ^ ***a*** ***b*** Bubeck, Sébastien; Chandrasekaran, Varun; Eldan, Ronen; Gehrke, Johannes; Horvitz, Eric; Kamar, Ece; Lee, Peter; Lee, Yin Tat; Li, Yuanzhi; Lundberg, Scott; Nori, Harsha; Palangi, Hamid; Ribeiro, Marco Tulio; Zhang, Yi (2023). "Sparks of Artificial General Intelligence: Early experiments with GPT-4". arXiv:2303.12712 (https://arxiv.org/abs/2303.12712) [cs.CL (https://arxiv.org/archive/cs.CL)]。
76. **^** "ChatGPT is more like an 'alien intelligence' than a human brain, says futurist" (https://www.zdnet.com/article/chatgpt-is-more-like-an-alien-intelligence-than-a-human-brain-says-futurist/) (英語). *ZDNET*. (2023年) 2023年6月12日閲覧。
77. ^ ***a*** ***b*** Newport, Cal (13 April 2023). "What Kind of Mind Does ChatGPT Have?" (https://www.newyorker.com/science/annals-of-artificial-intelligence/what-kind-of-mind-does-chatgpt-have). *The New Yorker* 2023年6月12日閲覧。.
78. **^** Roose, Kevin (2023年5月30日). "Why an Octopus-like Creature Has Come to Symbolize the State of A.I." (https://www.nytimes.com/2023/05/30/technology/shoggoth-meme-ai.html). *The New York Times* 2023年6月12日閲覧。
79. **^** "The A to Z of Artificial Intelligence" (https://time.com/6271657/a-to-z-of-artificial-intelligence/) (英語). *Time Magazine*. (2023年4月13日) 2023年6月12日閲覧。
80. ^ Ji, Ziwei; Lee, Nayeon; Frieske, Rita; Yu, Tiezheng; Su, Dan; Xu, Yan; Ishii, Etsuko; Bang, Yejin et al. (November 2022). "Survey of Hallucination in Natural Language Generation" (https://dl.acm.org/doi/pdf/10.1145/3571730) (pdf). *ACM Computing Surveys* (Association for Computing Machinery) **55** (12): 1–38. arXiv:2202.03629. doi:10.1145/3571730 (https://doi.org/10.1145%2F3571730) 2023年1月15日閲覧。.
81. **^** "Prepare for truly useful large language models" (英語). *Nature Biomedical Engineering*: pp. 85–86. (2023年3月7日). doi:10.1038/s41551-023-01012-6 (https://doi.org/10.1038%2Fs41551-023-01012-6)
82. **^** "Your job is (probably) safe from artificial intelligence" (https://www.economist.com/finance-and-economics/2023/05/07/your-job-is-probably-safe-from-artificial-intelligence). *The Economist*. (2023年5月7日) 2023年6月18日閲覧。
83. **^** "Generative AI Could Raise Global GDP by 7% (https://www.goldmansachs.com/intelligence/pages/generative-ai-could-raise-global-gdp-by-7-percent.html)". *Goldman Sachs*. 2023年6月18日閲覧。
84. **^** Alba, Davey (2023年5月1日). "AI chatbots have been used to create dozens of news content farms" (https://www.japantimes.co.jp/news/2023/05/01/business/tech/ai-fake-news-content-farms/). *The Japan Times* 2023年6月18日閲覧。
85. **^** "Could chatbots help devise the next pandemic virus?" (https://www.science.org/content/article/could-chatbots-help-devise-next-pandemic-virus) (英語). *Science*. (14 June 2023). doi:10.1126/science.adj2463 (https://doi.org/10.1126%2Fscience.adj2463).
86. ^ ***a*** ***b*** Devlin, Jacob; Chang, Ming-Wei; Lee, Kenton; Toutanova, Kristina (11 October 2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". arXiv:1810.04805v2 (https://arxiv.org/abs/1810.04805v2) [cs.CL (https://arxiv.org/archive/cs.CL)]。

87. ^ "BERT (https://github.com/google-research/bert)" (2023年3月13日). 2023年4月28日閲覧。
88. ^ Patel, Ajay; Li, Bryan; Rasooli, Mohammad Sadegh; Constant, Noah; Raffel, Colin; Callison-Burch, Chris (2022). "Bidirectional Language Models Are Also Few-shot Learners" (https://www.semanticscholar.org/paper/Bidirectional-Language-Models-Are-Also-Few-shot-Patel-Li/b65b7f480a61d3dd31d8117b349cabc87c8ccf6c) (英語). *ArXiv*.
89. ^ "BERT, RoBERTa, DistilBERT, XLNet: Which one to use? (https://www.kdnuggets.com/bert-roberta-distilbert-xlnet-which-one-to-use.html)". 2023年5月13日閲覧。
90. ^ Naik, Amit Raja (2021年9月23日). "Google Introduces New Architecture To Reduce Cost Of Transformers (https://analyticsindiamag.com/google-introduces-new-architecture-to-reduce-cost-of-transformers/)". *Analytics India Magazine*. 2023年5月13日閲覧。
91. ^ Yang, Zhilin; Dai, Zihang; Yang, Yiming; Carbonell, Jaime; Salakhutdinov, Ruslan; Le, Quoc V. (2 January 2020). "XLNet: Generalized Autoregressive Pretraining for Language Understanding" (https://arxiv.org/abs/1906.08237). *arXiv:1906.08237 [cs]* 2023年5月5日閲覧。.
92. ^ "GPT-2: 1.5B Release (https://web.archive.org/web/20191114074358/https://openai.com/blog/gpt-2-1-5b-release/)" (英語). *OpenAI* (2019年11月5日). 2019年11月14日時点のオリジナル (https://openai.com/blog/gpt-2-1-5b-release/)よりアーカイブ。2019年11月14日閲覧。
93. ^ "Better language models and their implications (https://openai.com/research/better-language-models)". *openai.com*. 2023年4月28日閲覧。
94. ^ *a* *b* "OpenAI's GPT-3 Language Model: A Technical Overview (https://lambdalabs.com/blog/demystifying-gpt-3)" (英語). *lambdalabs.com*. 2023年4月28日閲覧。
95. ^ "gpt-2 (https://github.com/openai/gpt-2)". *GitHub*. 2023年3月13日閲覧。
96. ^ "ChatGPT: Optimizing Language Models for Dialogue (https://openai.com/blog/chatgpt/)" (英語). *OpenAI* (2022年11月30日). 2023年1月13日閲覧。
97. ^ "GPT Neo (https://github.com/EleutherAI/gpt-neo)" (2023年3月15日). 2023年4月28日閲覧。
98. ^ *a* *b* *c* Gao, Leo; Biderman, Stella; Black, Sid; Golding, Laurence; Hoppe, Travis; Foster, Charles; Phang, Jason; He, Horace; Thite, Anish; Nabeshima, Noa; Presser, Shawn; Leahy, Connor (31 December 2020). "The Pile: An 800GB Dataset of Diverse Text for Language Modeling". arXiv:2101.00027 (https://arxiv.org/abs/2101.00027) [cs.CL (https://arxiv.org/archive/cs.CL)]。
99. ^ *a* *b* Iyer, Abhishek (2021年5月15日). "GPT-3's free alternative GPT-Neo is something to be excited about (https://venturebeat.com/ai/gpt-3s-free-alternative-gpt-neo-is-something-to-be-excited-about/)". *VentureBeat*. 2023年4月28日閲覧。
100. ^ "GPT-J-6B: An Introduction to the Largest Open Source GPT Model | Forefront (https://www.forefront.ai/blog-posts/gpt-j-6b-an-introduction-to-the-largest-open-sourced-gpt-model)" (英語). *www.forefront.ai*. 2023年2月28日閲覧。
101. ^ "Using DeepSpeed and Megatron to Train Megatron-Turing NLG 530B, the World's Largest and Most Powerful Generative Language Model (https://www.microsoft.com/en-us/research/blog/using-deepspeed-and-megatron-to-train-megatron-turing-nlg-530b-the-worlds-largest-and-most-powerful-generative-language-model/)". *Microsoft Research* (2021年10月11日). 2023年4月28日閲覧。
102. ^ *a* *b* Template:Cite preprint
103. ^ Wang, Shuohuan; Sun, Yu; Xiang, Yang; Wu, Zhihua; Ding, Siyu; Gong, Weibao; Feng, Shikun; Shang, Junyuan et al. (December 23, 2021). *ERNIE 3.0 Titan: Exploring Larger-scale Knowledge Enhanced Pre-training for Language Understanding and Generation*. arXiv:2112.12731.

04. ^ Product (https://www.anthropic.com/product)” (英語). *Anthropic*. 2023年3月14日閲覧。
05. ^ **a** **b** Askell, Amanda; Bai, Yuntao; Chen, Anna; et al. (9 December 2021). "A General Language Assistant as a Laboratory for Alignment". arXiv:2112.00861 (https://arxiv.org/abs/2112.00861) [cs.CL (https://arxiv.org/archive/cs.CL)]。
06. ^ Bai, Yuntao; Kadavath, Saurav; Kundu, Sandipan; et al. (15 December 2022). "Constitutional AI: Harmlessness from AI Feedback". arXiv:2212.08073 (https://arxiv.org/abs/2212.08073) [cs.CL (https://arxiv.org/archive/cs.CL)]。
07. ^ “Language modelling at scale: Gopher, ethical considerations, and retrieval (https://www.deepmind.com/blog/language-modelling-at-scale-gopher-ethical-considerations-and-retrieval)” (英語). *www.deepmind.com*. 2023年3月20日閲覧。
08. ^ **a** **b** **c** Hoffmann, Jordan; Borgeaud, Sebastian; Mensch, Arthur; et al. (29 March 2022). "Training Compute-Optimal Large Language Models". arXiv:2203.15556 (https://arxiv.org/abs/2203.15556) [cs.CL (https://arxiv.org/archive/cs.CL)]。
09. ^ **a** **b** “LaMDA: Towards Safe, Grounded, and High-Quality Dialog Models for Everything (https://ai.googleblog.com/2022/01/lamda-towards-safe-grounded-and-high.html)” (英語). *ai.googleblog.com* (2022年1月21日). 2023年3月9日閲覧。
10. ^ Black, Sidney; Biderman, Stella; Hallahan, Eric (1 May 2022). *GPT-NeoX-20B: An Open-Source Autoregressive Language Model* (https://aclanthology.org/2022.bigscience-1.9/). Proceedings of BigScience Episode #5 -- Workshop on Challenges & Perspectives in Creating Large Language Models. Vol. Proceedings of BigScience Episode #5 -- Workshop on Challenges & Perspectives in Creating Large Language Models. pp. 95–136. 2022年12月19日閲覧。
111. ^ “An empirical analysis of compute-optimal large language model training (https://www.deepmind.com/blog/an-empirical-analysis-of-compute-optimal-large-language-model-training)”. *Deepmind Blog* (2022年4月12日). 2023年4月28日閲覧。
112. ^ “Pathways Language Model (PaLM): Scaling to 540 Billion Parameters for Breakthrough Performance (https://ai.googleblog.com/2022/04/pathways-language-model-palm-scaling-to.html)” (英語). *ai.googleblog.com* (2022年4月4日). 2023年3月9日閲覧。
113. ^ “Democratizing access to large-scale language models with OPT-175B (https://ai.facebook.com/blog/democratizing-access-to-large-scale-language-models-with-opt-175b/)” (英語). *ai.facebook.com*. 2023年4月28日閲覧。
114. ^ Zhang, Susan; Roller, Stephen; Goyal, Naman; Artetxe, Mikel; Chen, Moya; Chen, Shuohui; Dewan, Christopher; Diab, Mona; Li, Xian; Lin, Xi Victoria; Mihaylov, Todor; Ott, Myle; Shleifer, Sam; Shuster, Kurt; Simig, Daniel; Koura, Punit Singh; Sridhar, Anjali; Wang, Tianlu; Zettlemoyer, Luke (21 June 2022). "OPT: Open Pre-trained Transformer Language Models". arXiv:2205.01068 (https://arxiv.org/abs/2205.01068) [cs.CL (https://arxiv.org/archive/cs.CL)]。
115. ^ **a** **b** Khrushchev, Mikhail; Vasilev, Ruslan; Petrov, Alexey; Zinov, Nikolay (2022-06-22), *YaLM 100B* (https://github.com/yandex/YaLM-100B) 2023年3月18日閲覧。
116. ^ **a** **b** Lewkowycz, Aitor; Andreassen, Anders; Dohan, David; Dyer, Ethan; Michalewski, Henryk; Ramasesh, Vinay; Slone, Ambrose; Anil, Cem; Schlag, Imanol; Gutman-Solo, Theo; Wu, Yuhuai; Neyshabur, Behnam; Gur-Ari, Guy; Misra, Vedant (30 June 2022). "Solving Quantitative Reasoning Problems with Language Models". arXiv:2206.14858 (https://arxiv.org/abs/2206.14858) [cs.CL (https://arxiv.org/archive/cs.CL)]。

17. ^ "Minerva: Solving Quantitative Reasoning Problems with Language Models (https://ai.googleblog.com/2022/06/minerva-solving-quantitative-reasoning.html)" (英語). *ai.googleblog.com*. 2023年3月20日閲覧。
18. ^ Ananthaswamy, Anil (2023年3月8日). "In AI, is bigger always better? (https://www.nature.com/articles/d41586-023-00641-w)". *Nature*. 2023年4月28日閲覧。
19. ^ "bigscience/bloom · Hugging Face (https://huggingface.co/bigscience/bloom)". *huggingface.co*. 2023年4月28日閲覧。
20. ^ Taylor, Ross; Kardas, Marcin; Cucurull, Guillem; Scialom, Thomas; Hartshorn, Anthony; Saravia, Elvis; Poulton, Andrew; Kerkez, Viktor; Stojnic, Robert (16 November 2022). "Galactica: A Large Language Model for Science". arXiv:2211.09085 (https://arxiv.org/abs/2211.09085) [cs.CL (https://arxiv.org/archive/cs.CL)]。
21. ^ "20B-parameter Alexa model sets new marks in few-shot learning (https://www.amazon.science/blog/20b-parameter-alexa-model-sets-new-marks-in-few-shot-learning)" (英語). *Amazon Science* (2022年8月2日). 2023年4月28日閲覧。
22. ^ Soltan, Saleh; Ananthakrishnan, Shankar; FitzGerald, Jack; et al. (3 August 2022). "AlexaTM 20B: Few-Shot Learning Using a Large-Scale Multilingual Seq2Seq Model". arXiv:2208.01448 (https://arxiv.org/abs/2208.01448) [cs.CL (https://arxiv.org/archive/cs.CL)]。
23. ^ "AlexaTM 20B is now available in Amazon SageMaker JumpStart | AWS Machine Learning Blog (https://aws.amazon.com/blogs/machine-learning/alexatm-20b-is-now-available-in-amazon-sagemaker-jumpstart/)". *aws.amazon.com* (2022年11月17日). 2023年3月13日閲覧。
24. ^ ***a*** ***b*** ***c*** "Introducing LLaMA: A foundational, 65-billion-parameter large language model (https://ai.facebook.com/blog/large-language-model-llama-meta-ai/)". *Meta AI* (2023年2月24日). 2023年4月28日閲覧。
125. ^ "Stanford CRFM (https://crfm.stanford.edu/2023/03/13/alpaca.html)". *crfm.stanford.edu*. 2023年4月28日閲覧。
126. ^ "GPT-4 Technical Report (https://web.archive.org/web/20230314190904/https://cdn.openai.com/papers/gpt-4.pdf)". *OpenAI* (2023年). 2023年3月14日時点のオリジナル (https://cdn.openai.com/papers/gpt-4.pdf)よりアーカイブ。2023年3月14日閲覧。
127. ^ Dey, Nolan (2023年3月28日). "Cerebras-GPT: A Family of Open, Compute-efficient, Large Language Models (https://www.cerebras.net/blog/cerebras-gpt-a-family-of-open-compute-efficient-large-language-models/)". *Cerebras*. 2024年4月28日閲覧。
128. ^ ***a*** ***b*** ***c*** Technology Innovation Institute Introduces World's Most Powerful Open LLM: Falcon 180B (https://www.tii.ae/news/technology-innovation-institute-introduces-worlds-most-powerful-open-llm-falcon-180b) Technology Innovation Institute 2023年9月6日
129. ^ Wu, Shijie; Irsoy, Ozan; Lu, Steven; Dabravolski, Vadim; Dredze, Mark; Gehrmann, Sebastian; Kambadur, Prabhanjan; Rosenberg, David et al. (March 30, 2023). *BloombergGPT: A Large Language Model for Finance*. arXiv:2303.17564.
130. ^ Ren, Xiaozhe; Zhou, Pingyi; Meng, Xinfan; Huang, Xinjing; Wang, Yadao; Wang, Weichao; Li, Pengfei; Zhang, Xiaoda et al. (March 19, 2023). *PanGu-Σ: Towards Trillion Parameter Language Model with Sparse Heterogeneous Computing*. arXiv:2303.10845.
131. ^ Köpf, Andreas; Kilcher, Yannic; von Rütte, Dimitri; Anagnostidis, Sotiris; Tam, Zhi-Rui; Stevens, Keith; Barhoum, Abdullah; Duc, Nguyen Minh et al. (2023-04-14). "OpenAssistant Conversations -- Democratizing Large Language Model Alignment" (http://arxiv.org/abs/2304.07327). *arXiv:2304.07327 [cs]*.

32. ^ Elias, Jennifer (2023年5月16日). “Google's newest A.I. model uses nearly five times more text data for training than its predecessor (https://www.cnbc.com/2023/05/16/googles-palm-2-uses-nearly-five-times-more-text-data-than-predecessor.html)”. *CNBC*. 2023年5月18日閲覧。
133. ^ “Introducing PaLM 2 (https://blog.google/technology/ai/google-palm-2-ai-large-language-model/)”. *Google* (2023年5月10日). 2023年6月24日閲覧。

関連項目

- 基盤モデル - 幅広いデータで大規模に訓練された、幅広い下流タスクに適用できる大規模な人工知能モデル。事前学習済の大規模言語モデル（LLM）はその初期の例である。
- 生成的人工知能 - プロンプトに応答してテキスト、画像、または他のメディアを生成することができる人工知能システムの一種