Language Models are Few-Shot Learners - a review of GPT-3 model[1]

CS410 Technology review

by Dmitry Villevald (dmitryv2@illinois.edu)

Introduction

There are three paradigms in machine learning. First is to build the model from scratch for a specific use case. Such models usually require a lot of data to train and do not generalize well. Second is a transfer learning approach where a pre-trained model is fine-tuned for performance on a specific task. Lastly, one can build a general-purpose model trained on a very large corpus which would need only a few examples (a few shots) to learn how to perform on a new task. This paper explores the model from the latter paradigm - a Generative Pre-trained Transformer 3 or GPT-3 which is an autoregressive language prediction model introduced by OpenAI in May 2020. In this technical review I explore the model approach, its architecture, data, training process and performance. While the paper describes multiple applications of GPT-3, in my review I would like to focus on a model's ability to generate short news stories and its possible impact on a society.

Approach

The main focus of this study is to systematically explore the different settings for learning with the context, i.e. how much task-specific data the model has to rely on. In particular, the following three tasks were explored:

- **Few-shot** describes a case when a model was shown 10-100 examples-demonstrations (depending on how many would fit into the context window) and then a final example of context with a model expecting to give a prediction. No weight updates were allowed.
- **One-shot** is a task similar to the few-shot described above except that (1) only one example-demonstration was allowed and (2) natural language description of the task was provided.
- **Zero-shot** is a case where no demonstration was allowed and only the natural language description of the task was given to the model (i.e. entered in the context window).

Training data

The Common Crawl dataset with nearly 1 trillion records was curated by filtering based on similarity to a few high-quality reference corpora, performing fuzzy deduplication and, finally, by augmenting the cleaned data with a few high-quality reference corpora to increase data

[1] This is a technical review of the paper “Language Models are Few-Shot Learners” by Brown et al. (https://arxiv.org/abs/2005.14165)

diversity. Because of the size of the data, significant efforts were made to reduce data contamination, i.e. to identify and remove the development and test data inadvertently seen during pre-training. The final 570GB training set consisted of the filtered Common Crawl data (410B byte-pair-encoded tokens), WebText (19B), Books1 (12B), Books2 (55B) and English-language Wikipedia (3B).

Architecture

8 different model versions were tested. The largest, GPT-3, had 175B trainable parameters and 96 layers. The model architecture for all models (shown on Figure 1) was similar to the one of GPT-2[2] and included the following[3]:

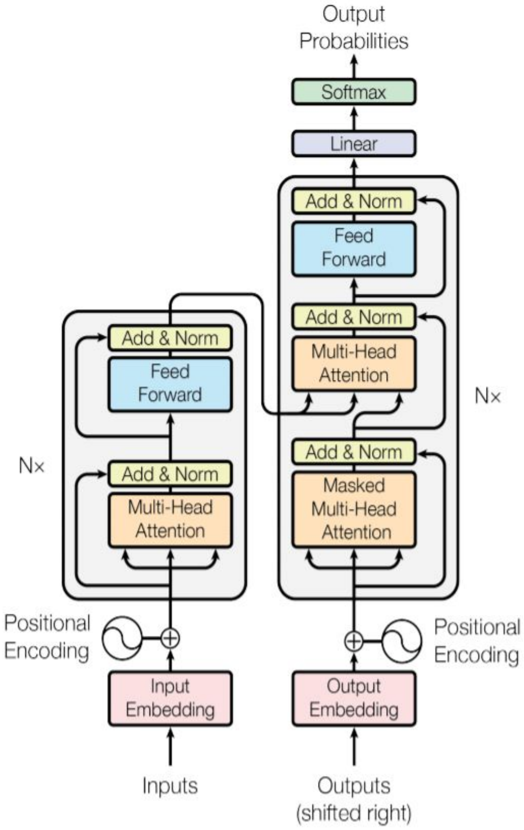


Figure 1: The Transformer - model architecture.

Input (context window) is an array which can fit 2,048 tokens. (The sequence of input words/characters is encoded using the model's vocabulary of 50,257 terms.)

Embedding layer projects one-hot encoded tokens representation to 12,288 dimension embedding vectors creating a 2,048 x 12,288 embedding matrix.

Positional encoding layer mapps a position of each token in the input sequence to 12,288 sinusoidal functions creating a 2,048 x 12,288 positional embedding matrix (which is then added to the embedding matrix mentioned above).

Multi-head attention layer learns the importance of each token in a sequence to each other token. The first attention layer (head) learns three linear projections (called *queries*, *keys* and *values*) and transforms the sequence embeddings matrix into 2,048 x 128 matrix. Then the second attention layer takes this matrix as input and produces another 2,048 x 128 matrix. This process continues for all 96 heads and then all output 96 matrices (output of head #1, #2, etc.) are concatenated into a single 2,048 x 12,288 output matrix.

Feed forward layer is a fully connected hidden layer of size 4*12,288 which takes a 2,048 x 12,288 matrix and returns a matrix of the same size.

Add & Norm layer takes input and output of the previous layer (Multi-head attention or Feed forward), adds them up and normalizes the result. All inputs and outputs of this layer are 2,048 x 12,288 matrices.

[2] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multi task learners, 2019

[3] http://dugas.ch/artificial_curiosity/GPT_architecture.html

Linear and softmax layers implement decoding. The linear layer does a reversing encoding by transforming the input matrix 2,048 x 12,288 back into a 2,048 x 50,257 (number of tokens x vocabulary size) word encoding matrix. Softmax layer is then applied to produce the probabilities for each of 2.048 tokens.

Output is a sequence of 2,048 probabilities mapped to specific tokens (in practice most often only the first token with the highest probability is used as a model prediction).

Training process

The model was trained on 300B tokens on V100 GPU's with Adam optimizer.

- **Learning rate** of 0.6 x 10^(-4) was used. A cosine decay down to 10% was used for the learning rate over the first 260 billion tokens (after 260 billion tokens the training continued at 10% of the original learning rate).

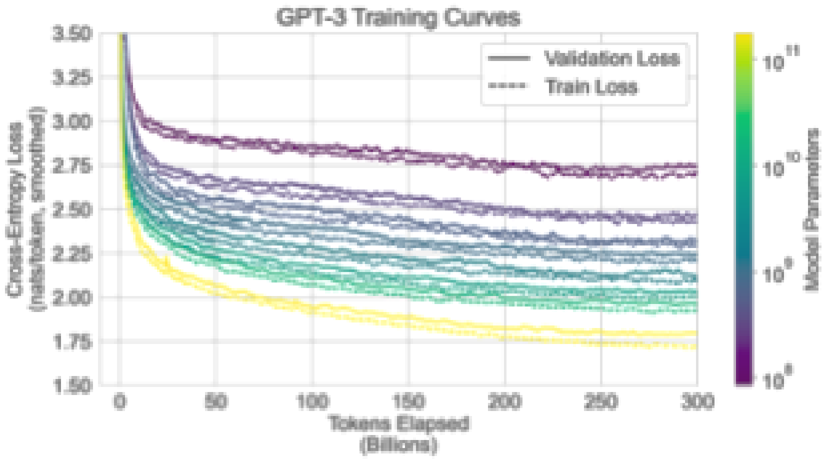
- **Batch size** of 3.2M was used. The batch size was linearly increased from initial small value (32k tokens) to the full value over the first 4-12 billion tokens of training.

- **Data**. The mix of training data included Common Crawl (60% of the training set), WebText2 (22%), Books1 (8%), Books2 (8%) and Wikipedia (3%).

- **Risk of overfitting**. To minimize the risk of overfitting, the data were sampled without replacement during the training (within each epoch). Also, a weight decay of 0.1 was applied for regularization.

- **Other details.** Training was always performed on sequences of the full context window (2,048 tokens). Multiple documents shorter than 2,048 were packed into a single sequence for increased computational efficiency. Sequences with multiple documents were not masked in any special way but documents within a sequence were delimited with a special end-of-text token. This gave the language model the information necessary to infer that the context separated by the end-of-text token is unrelated.

The following chart shows training curves for 8 models (the curve for GPT-3 with 175B parameters is in yellow.)



Performance

In this review I would like to focus on one task which GTP-3 performs well - a generation of synthetic news articles. In particular, the model was given three news articles in a context window to learn (i.e. “few-shot” learning ability was exploited) followed by the title and subtitle of the fourth proposed article which the model was expected to complete.

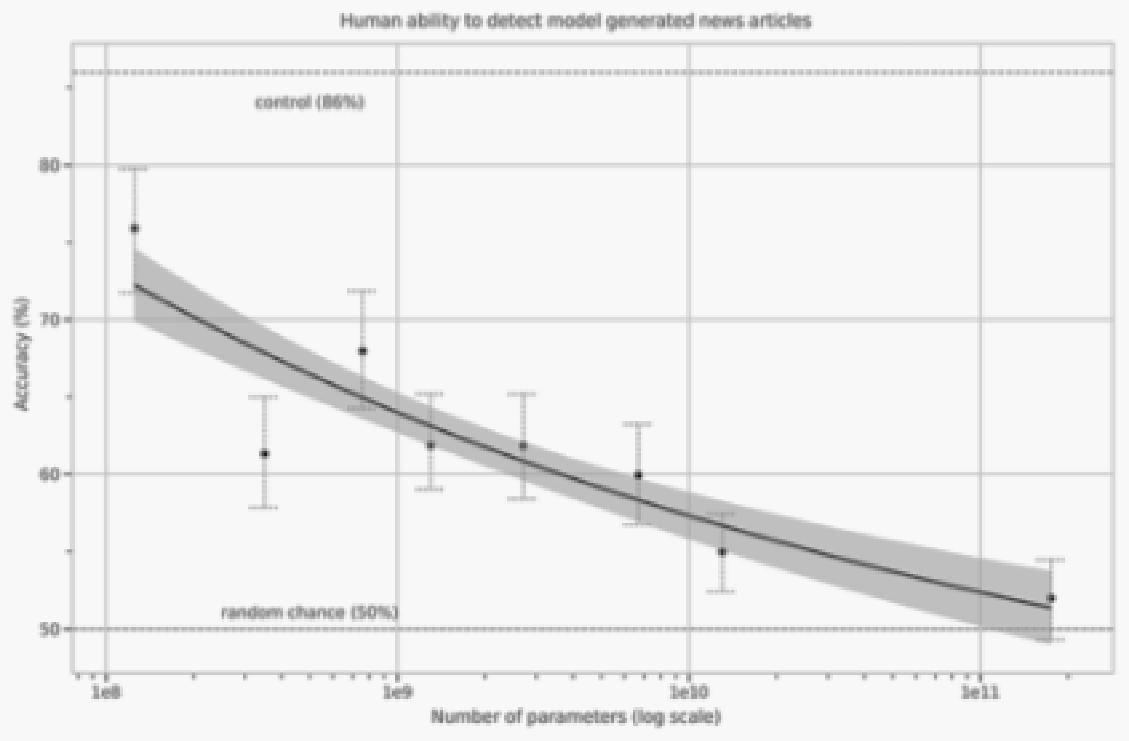
To evaluate GPT-3’s ability to generate plausible news articles, authors set up an experiment testing human’s ability to distinguish between the real news and the articles generated by the model. In particular, they selected 25 article titles and subtitles from newser.com with an average length of 215 words. Then they generated completions of these titles and subtitles from GPT-3 (with 175B parameters) with an average length of 200 words. 80 US-based participants were presented with a quiz consisting of these titles and subtitles followed by either the model generated or human written article. Participants were asked to select where the article was “very likely written by a machine”, “more likely written by a machine”, “I don’t know”, “more likely written by a human”, or “very likely written by a human”. Note that the selected articles were not in the models’ training data and the model outputs were formatted and selected programmatically to prevent human cherry-picking. To control for participant effort and attention the intentionally bad model generated articles in the same format were included in the quiz. This was done by generating articles from an intentionally bad “control model” with 160M parameters, no context and increased output randomness.

The share of correct assignments in all non-neutral assignments per participant - a mean human accuracy - at detecting that the intentionally bad articles were model generated was 86% (where 50% is a chance level performance.) By contrast, **mean human accuracy at detecting articles generated by GPT-3 model was 52% which is barely above the chance level performance**.

Evaluators indicated that much of model-generated text was difficult to distinguish from human content. Some factual inaccuracies were an indicator that an article is model-generated because, unlike humans, the models do not have access to the specific facts that the article titles refer to. Other indicators included repetitions, non sequiturs (i.e. logical inconsistencies), and unusual phrasings, although these were often subtle enough that they were not noticed.

Authors also tested a hypothesis that human accuracy at detecting model-generated text increases as humans observe more tokens. If this is true then humans are expected to do better at detecting longer machine-generated articles. To test this the authors selected 12 world news articles from Reuters with an average length of 569 words and then generated completions of these articles from GPT-3 with an average length of 498 words (i.e. 298 words longer than the first experiment.) Following the methodology above, authors found that mean human accuracy at detecting the intentionally bad longer articles from the control model was indeed a bit higher - 88% compared to 86% for 200-word-long articles. However, mean human accuracy at detecting the longer articles produced by GPT-3 175B model was still barely above chance at about 52% which indicates that, for news articles that are around 500 words long, GPT-3 continues to produce articles that humans find difficult to distinguish from human-written news articles.

Finally, the authors showed that model size is important and the ability of humans to detect model-generated text decreases with the increase of model complexity. As the chart below shows, mean human accuracy for GTP-3 Small model with only 125M parameters was about 72% compared to 52% for GPT-3 175B-parameter model.



Impact on society

Advanced language models like GPT-3 have a wide range of applications beneficial to society. Writing code, converting natural language commands into shell commands, improving search engine responses, writing auto-completion and generating game narrative are just a few useful applications. A model's ability to generate short news articles practically indistinguishable from human-generated text suggests that in the near future the advanced language models will be helping people to write anything from legal contracts to movie scripts and poetry. Because the model does not need to be tuned for a particular task, there are many other applications which have not been discovered yet.

However, the power of these models can also be used to advance potentially harmful applications. Examples include phishing, spam, misinformation, abuse of legal and governmental processes, fraudulent academic essay writing, fake news generation and many others. While the ability to use model-generated high quality text to deploy, for example, chat bots (to impersonate humans) for phishing on a massive scale is scary enough, a prospect of converting this text into speech and engaging bots into phone conversations with humans brings this threat to a new level.

Many of these malicious applications rely on human beings' ability to write sufficiently high quality text, and the language models that can generate it could lower existing barriers to carrying out these malicious activities. While the authors of the reviewed paper state that at the moment there is no evidence that language models are used on scale by bad actors, it is very likely that AI researchers will eventually develop language models that are sufficiently consistent and easy to manage so that they will be of greater interest to malicious actors. This will introduce unique challenges for the regulators, the general public and the research community.

Conclusion

In this paper I reviewed GPT-3 - a state-of-the-art autoregressive language prediction model introduced by OpenAI in May 2020 - and analyzed the model approach, architecture, training data and process and, finally, the model performance on a task of generating short news articles. While the model architecture is similar to the one of its predecessor GPT-2, a main advantage of GTP-3 seems to be its size (width in particular) and the size of data it was trained on which allow the model, after seeing only a few examples, to reliably generate short articles in the “news” genre which humans found difficult to distinguish from human-generated ones. I also explored a potential impact of this innovation on a society which could bring both new opportunities as well as unseen before challenges.