OpenAI

Research Areas: Generative models, Reinforcement Learning, Human feedback, Reasoning, Representation Learning, Speech Recognition, Adversarial examples, Software Engineering, Sparsity, Community
Years: 2023, 2022, 2021, 2020, 2019, 2018, 2017, 2016
Number of Research Papers Evaluated: 166
Number of Research Papers After Evaluation: 86
Author: Dhruv Awasthi

GPT-4V(ision) system card (Sep 2023; Link)

Abstract:
GPT-4 with vision (GPT-4V) enables users to instruct GPT-4 to analyze image inputs provided by the user, and is the latest capability we are making broadly available. Incorporating additional modalities (such as image inputs) into large language models (LLMs) is viewed by some as a key frontier in artificial intelligence research and development. Multimodal LLMs offer the possibility of expanding the impact of language-only systems with novel interfaces and capabilities, enabling them to solve new tasks and provide novel experiences for their users. In this system card, we analyze the safety properties of GPT-4V. Our work on safety for GPT-4V builds on the work done for GPT-4 and here we dive deeper into the evaluations, preparation, and mitigation work done specifically for image inputs.

Improving mathematical reasoning with process supervision (May 2023; Link)

Abstract:
In recent years, large language models have greatly improved in their ability to perform complex multi-step reasoning. However, even stateof-the-art models still regularly produce logical mistakes. To train more reliable models, we can turn either to outcome supervision, which provides feedback for a final result, or process supervision, which provides feedback for each intermediate reasoning step. Given the importance of training reliable models, and given the high cost of human feedback, it is important to carefully compare the both methods. Recent work has already begun this comparison, but many questions still remain. We conduct our own investigation, finding that process supervision significantly outperforms outcome supervision for training models to solve problems from the challenging MATH dataset. Our process-supervised model solves 78% of problems from a representative subset of the MATH test set. Additionally, we show that active learning significantly improves

the efficacy of process supervision. To support related research, we also release PRM800K, the complete dataset of 800,000 step-level human feedback labels used to train our best reward model.

Language models can explain neurons in language models (May 2023; Link)

Abstract:
Language models have become more capable and more broadly deployed, but our understanding of how they work internally is still very limited. For example, it might be difficult to detect from their outputs whether they use biased heuristics or engage in deception. Interpretability research aims to uncover additional information by looking inside the model.

One simple approach to interpretability research is to first understand what the individual components (neurons and attention heads) are doing. This has traditionally required humans to manually inspect neurons to figure out what features of the data they represent. This process doesn't scale well: it's hard to apply it to neural networks with tens or hundreds of billions of parameters. We propose an automated process that uses GPT-4 to produce and score natural language explanations of neuron behavior and apply it to neurons in another language model.

This work is part of the third pillar of our approach to alignment research: we want to automate the alignment research work itself. A promising aspect of this approach is that it scales with the pace of AI development. As future models become increasingly intelligent and helpful as assistants, we will find better explanations.

GPT-4 (Mar 2023; Link)

Abstract:
We report the development of GPT-4, a large-scale, multimodal model which can accept image and text inputs and produce text outputs. While less capable than humans in manyreal-world scenarios, GPT-4 exhibits human-level performance on various professional and academic benchmarks, including passing a simulated bar exam with a score around the top 10% of test takers. GPT-4 is a Transformerbased model pre-trained to predict the next token in a document. The post-training alignment process results in improved performance on measures of factuality and adherence to desired behavior. A core component of this project was developing infrastructure and optimization methods that behave predictably across a wide range of scales. This allowed us to accurately predict some aspects of GPT-4's performance based on models trained with no more than 1/1,000th the compute of GPT-4.

Scaling laws for reward model over optimization (Oct 2022; Link)

Abstract:
In reinforcement learning from human feedback, it is common to optimize against a reward model trained to predict human preferences. Because the reward model is an imperfect proxy, optimizing its value too much can hinder ground truth performance, in accordance with Goodhart's law. This effect has been frequently observed, but not carefully measured due to the expense of collecting human preference data. In this work, we use a synthetic setup in which a fixed "gold-standard" reward model plays the role of humans, providing labels used to train a proxy reward model. We study how the gold reward model score changes as we optimize against the proxy reward model using either reinforcement learning or best-of-n sampling. We find that this relationship follows a different functional form depending on the method of optimization, and that in both cases its coefficients scale smoothly with the number of reward model parameters. We also study the effect on this relationship of the size of the reward model dataset, the number of reward model and policy parameters, and the coefficient of the KL penalty added to the reward in the reinforcement learning setup. We explore the implications of these empirical results for theoretical considerations in AI alignment.

Whisper - Robust Speech Recognition via Large-Scale Weak Supervision (Sep 2022; Link)

Abstract:
We study the capabilities of speech processing systems trained simply to predict large amounts of transcripts of audio on the internet. When scaled to 680,000 hours of multilingual and multitask supervision, the resulting models generalize well to standard benchmarks and are often competitive with prior fully supervised results but in a zeroshot transfer setting without the need for any finetuning. When compared to humans, the models approach their accuracy and robustness. We are releasing models and inference code to serve as a foundation for further work on robust speech processing.

Whisper is an automatic speech recognition (ASR) system trained on 680,000 hours of multilingual and multitask supervised data collected from the web. We show that the use of such a large and diverse dataset leads to improved robustness to accents, background noise and technical language. Moreover, it enables transcription in multiple languages, as well as translation from those languages into English. We are open-sourcing models and inference code to serve as a foundation for building useful applications and for further research on robust speech processing.

Efficient training of language models to fill in the middle (Jul 2022; Link)

Abstract:
We show that autoregressive language models can learn to infill text after we apply a straightforward transformation to the dataset, which simply moves a span of text from the middle of a document to its end. While this data augmentation has garnered much interest in recent years, we provide extensive evidence that training models with a large fraction of data transformed in this way does not harm the original left-to-right generative capability, as measured by perplexity and sampling evaluations across a wide range of scales. Given the usefulness, simplicity, and efficiency of training models to fill-in-the-middle (FIM), we suggest that future autoregressive language models be trained with FIM by default. To this end, we run a series of ablations on key hyperparameters, such as the data transformation frequency, the structure of the transformation, and the method of selecting the infill span. We use these ablations to prescribe strong default settings and best practices to train FIM models. We have released our best infilling model trained with best practices in our API, and release our infilling benchmarks to aid future research.

Codex - A hazard analysis framework for code synthesis large language models (Jul 2022; Link)

Abstract:
Codex, a large language model (LLM) trained on a variety of codebases, exceeds the previous state of the art in its capacity to synthesize and generate code. Although Codex provides a plethora of benefits, models that may generate code on such scale have significant limitations, alignment problems, the potential to be misused, and the possibility to increase the rate of progress in technical fields that may themselves have destabilizing impacts or have misuse potential. Yet such safety impacts are not yet known or remain to be explored. In this paper, we outline a hazard analysis framework constructed at OpenAI to uncover hazards or safety risks that the deployment of models like Codex may impose technically, socially, politically, and economically. The analysis is informed by a novel evaluation framework that determines the capacity of advanced code generation techniques against the complexity and expressivity of specification prompts, and their capability to understand and execute them relative to human ability.

Learning to play Minecraft with Video PreTraining (Jun 2022; Link)

Abstract:
Pretraining on noisy, internet-scale datasets has been heavily studied as a technique for training models with broad, general capabilities for text, images, and other modalities. 1–6 However, for many sequential decision domains such as robotics,

video games, and computer use, publicly available data does not contain the labels required to train behavioral priors in the same way. We extend the internet-scale pretraining paradigm to sequential decision domains through semi-supervised imitation learning wherein agents learn to act by watching online unlabeled videos. Specifically, we show that with a small amount of labeled data we can train an inverse dynamics model accurate enough to label a huge unlabeled source of online data – here, online videos of people playing Minecraft – from which we can then train a general behavioral prior. Despite using the native human interface (mouse and keyboard at 20Hz), we show that this behavioral prior has nontrivial zeroshot capabilities and that it can be fine-tuned, with both imitation learning and reinforcement learning, to hard-exploration tasks that are impossible to learn from scratch via reinforcement learning. For many tasks our models exhibit humanlevel performance, and we are the first to report computer agents that can craft diamond tools, which can take proficient humans upwards of 20 minutes (24,000 environment actions) of gameplay to accomplish.

AI-written critiques help humans notice flaws (Jun 2022; Link)

Abstract:

We fine-tune large language models to write natural language critiques (natural language critical comments) using behavioral cloning. On a topic-based summarization task, critiques written by our models help humans find flaws in summaries that they would have otherwise missed. Our models help find naturally occurring flaws in both model and human written summaries, and intentional flaws in summaries written by humans to be deliberately misleading. We study scaling properties of critiquing with both topic-based summarization and synthetic tasks. Larger models write more helpful critiques, and on most tasks, are better at self-critiquing, despite having harder-to-critique outputs. Larger models can also integrate their own self-critiques as feedback, refining their own summaries into better ones. Finally, we motivate and introduce a framework for comparing critiquing ability to generation and discrimination ability. Our measurements suggest that even large models may still have relevant knowledge they cannot or do not articulate as critiques. These results are a proof of concept for using AI-assisted human feedback to scale the supervision of machine learning systems to tasks that are difficult for humans to evaluate directly. We release our training datasets, as well as samples from our critique assistance experiments.

Techniques for training large neural networks (Jun 2022; Link)

Abstract:

Large neural networks are at the core of many recent advances in AI, but training them is a difficult engineering and research challenge which requires orchestrating a cluster of GPUs to perform a single synchronized calculation. As cluster and model sizes have grown, machine learning practitioners have developed an increasing variety of techniques to parallelize model training over many GPUs. At first glance, understanding these parallelism techniques may seem daunting, but with only a few assumptions about the structure of the computation these techniques become much more clear—at that point, you're just shuttling around opaque bits from A to B like a network switch shuttles around packets.

Solving (some) formal math olympiad problems (Feb 2022; Link)

Abstract:
We built a neural theorem prover for Lean that learned to solve a variety of challenging high-school olympiad problems, including problems from the AMC12 and AIME competitions, as well as two problems adapted from the IMO. The prover uses a language model to find proofs of formal statements. Each time we find a new proof, we use it as new training data, which improves the neural network and enables it to iteratively find solutions to harder and harder statements.

We achieved a new state-of-the-art (41.2% vs 29.3%) on the miniF2F benchmark, a challenging collection of high-school olympiad problems. Our approach, which we call statement curriculum learning, consists of manually collecting a set of statements of varying difficulty levels (without proof) where the hardest statements are similar to the benchmark we target. Initially our neural prover is weak and can only prove a few of them. We iteratively search for new proofs and re-train our neural network on the newly discovered proofs, and after 8 iterations, our prover ends up being vastly superior when tested on miniF2F.

Formal mathematics is an exciting domain to study because of (i) its richness, letting you prove arbitrary theorems which require reasoning, creativity and insight and (ii) its similarity to games—where AI has been spectacularly successful—in that it has an automated way of determining whether a proof is successful (i.e., verified by the formal system). As demonstrated in the trivial example below, proving a formal statement requires generating a sequence of proof steps, each proof step consisting in a call to a tactic. These tactics take mathematical terms as arguments and each tactic call will transform the current statement to prove, into statements that are easier to prove, until nothing is left to prove.

Aligning language models to follow instructions (Jan 2022; Link)

Abstract:
Making language models bigger does not inherently make them better at following a user's intent. For example, large language models can generate outputs that are untruthful, toxic, or simply not helpful to the user. In other words, these models are not aligned with their users. In this paper, we show an avenue for aligning language models with user intent on a wide range of tasks by fine-tuning with human feedback. Starting with a set of labeler-written prompts and prompts submitted through the OpenAI API, we collect a dataset of labeler demonstrations of the desired model behavior, which we use to fine-tune GPT-3 using

supervised learning. We then collect a dataset of rankings of model outputs, which we use to further fine-tune this supervised model using reinforcement learning from human feedback. We call the resulting models InstructGPT. In human evaluations on our prompt distribution, outputs from the 1.3B parameter InstructGPT model are preferred to outputs from the 175B GPT-3, despite having 100x fewer parameters. Moreover, InstructGPT models show improvements in truthfulness and reductions in toxic output generation while having minimal performance regressions on public NLP datasets. Even though InstructGPT still makes simple mistakes, our results show that fine-tuning with human feedback is a promising direction for aligning language models with human intent.

Text and code embeddings by contrastive pre-training (Jan 2022; Link)

Abstract:

Text embeddings are useful features in many applications such as semantic search and computing text similarity. Previous work typically trains models customized for different use cases, varying in dataset choice, training objective and model architecture. In this work, we show that contrastive pre-training on unsupervised data at scale leads to high quality vector representations of text and code. The same unsupervised text embeddings that achieve new state-of-the-art results in linear-probe classification also display impressive semantic search capabilities and sometimes even perform competitively with fine-tuned models. On linear-probe classification accuracy averaging over 7 tasks, our best unsupervised model achieves a relative improvement of 4% and 1.8% over previous best unsupervised and supervised text embedding models respectively. The same text embeddings when evaluated on large-scale semantic search attains a relative improvement of 23.4%, 14.7%, and 10.6% over previous best unsupervised methods on MSMARCO, Natural Questions and TriviaQA benchmarks, respectively. Similarly to text embeddings, we train code embedding models on (text, code) pairs, obtaining a 20.8% relative improvement over prior best work on code search.

WebGPT: Browser-assisted question-answering with human feedback (Dec 2021; Link)

Abstract:

We fine-tune GPT-3 to answer long-form questions using a text-based web-browsing environment, which allows the model to search and navigate the web. By setting up the task so that it can be performed by humans, we are able to train models on the task using imitation learning, and then optimize answer quality with human feedback. To make human evaluation of factual accuracy easier, models must collect references while browsing in support of their answers. We train and evaluate our models on ELI5, a dataset of questions asked by Reddit users. Our best model is obtained by fine-tuning GPT-3 using behavior cloning, and then performing rejection sampling against a reward model trained to predict

human preferences. This model's answers are preferred by humans 56% of the time to those of our human demonstrators, and 69% of the time to the highest-voted answer from Reddit.

Solving math word problems (Oct 2021; Link)

Abstract:

State-of-the-art language models can match human performance on many tasks, but they still struggle to robustly perform multi-step mathematical reasoning. To diagnose the failures of current models and support research, we introduce GSM8K, a dataset of 8.5K high quality linguistically diverse grade school math word problems. We find that even the largest transformer models fail to achieve high test performance, despite the conceptual simplicity of this problem distribution. To increase performance, we propose training verifiers to judge the correctness of model completions. At test time, we generate many candidate solutions and select the one ranked highest by the verifier. We demonstrate that verification significantly improves performance on GSM8K, and we provide strong empirical evidence that verification scales more effectively with increased data than a finetuning baseline.

Summarizing books with human feedback (Sep 2021; Link)

Abstract:

To safely deploy powerful, general-purpose artificial intelligence in the future, we need to ensure that machine learning models act in accordance with human intentions. This challenge has become known as the alignment problem.

A scalable solution to the alignment problem needs to work on tasks where model outputs are difficult or time-consuming for humans to evaluate. To test scalable alignment techniques, we trained a model to summarize entire books, as shown in the following samples.Our model works by first summarizing small sections of a book, then summarizing those summaries into a higher-level summary, and so on.

Our best model is fine-tuned from GPT-3 and generates sensible summaries of entire books, sometimes even matching the average quality of human-written summaries: it achieves a 6/7 rating (similar to the average human-written summary) from humans who have read the book 5% of the time and a 5/7 rating 15% of the time. Our model also achieves state-of-the-art results on the BookSum dataset for book-length summarization. A zero-shot question-answering model can use our model’s summaries to obtain competitive results on the NarrativeQA dataset for book-length question answering.

TruthfulQA: Measuring how models mimic human falsehoods (Sep 2021; Link)

Abstract:
We propose a benchmark to measure whether a language model is truthful in generating answers to questions. The benchmark comprises 817 questions that span 38 categories, including health, law, finance and politics. We crafted questions that some humans would answer falsely due to a false belief or misconception. To perform well, models must avoid generating false answers learned from imitating human texts. We tested GPT-3, GPT-Neo/J, GPT-2 and a T5-based model. The best model was truthful on 58% of questions, while human performance was 94%. Models generated many false answers that mimic popular misconceptions and have the potential to deceive humans. The largest models were generally the least truthful. This contrasts with other NLP tasks, where performance improves with model size. However, this result is expected if false answers are learned from the training distribution. We suggest that scaling up models alone is less promising for improving truthfulness than fine-tuning using training objectives other than imitation of text from the web.

Introducing Triton: Open-source GPU programming for neural networks (Jun 2021; Link)

Abstract:
We're releasing Triton 1.0, an open-source Python-like programming language which enables researchers with no CUDA experience to write highly efficient GPU code—most of the time on par with what an expert would be able to produce.

Triton makes it possible to reach peak hardware performance with relatively little effort; for example, it can be used to write FP16 matrix multiplication kernels that match the performance of cuBLAS—something that many GPU programmers can't do—in under 25 lines of code. Our researchers have already used it to produce kernels that are up to 2x more efficient than equivalent Torch implementations, and we're excited to work with the community to make GPU programming more accessible to everyone.

Evaluating large language models trained on code (Jul 2021; Link)

Abstract:
We introduce Codex, a GPT language model fine-tuned on publicly available code from GitHub, and study its Python code-writing capabilities. A distinct production version of Codex powers GitHub Copilot. On HumanEval, a new evaluation set we release to measure functional correctness for synthesizing programs from docstrings, our model solves 28.8% of the problems, while GPT-3 solves 0% and GPT-J solves 11.4%. Furthermore, we find that

repeated sampling from the model is a surprisingly effective strategy for producing working solutions to difficult prompts. Using this method, we solve 70.2% of our problems with 100 samples per problem. Careful investigation of our model reveals its limitations, including difficulty with docstrings describing long chains of operations and with binding operations to variables. Finally, we discuss the potential broader impacts of deploying powerful code generation technologies, covering safety, security, and economics.

Improving language model behavior by training on a curated dataset (Jun 2021; Link)

Abstract:
We've found we can improve language model behavior with respect to specific behavioral values by fine-tuning on a curated dataset of <100 examples of those values. We also found that this process becomes more effective as models get larger. While the technique is still nascent, we're looking for OpenAI API users who would like to try it out and are excited to find ways to use these and other techniques in production use cases.

Language models can output almost any kind of text, in any kind of tone or personality, depending on the user's input. Our approach aims to give language model operators the tools to narrow this universal set of behaviors to a constrained set of values. While OpenAI provides guardrails and monitoring to ensure that model use-cases are compatible with our Charter, we view selecting the exact set of Charter-compatible values for the model as a choice that our users must face for their specific applications.

Understanding the capabilities, limitations, and societal impact of large language models (Feb 2021; Link)

Abstract:
On October 14th, 2020, researchers from OpenAI, the Stanford Institute for Human-Centered Artificial Intelligence, and other universities convened to discuss open research questions surrounding GPT-3, the largest publicly-disclosed dense language model at the time. The meeting took place under Chatham House Rules. Discussants came from a variety of research backgrounds including computer science, linguistics, philosophy, political science, communications, cyber policy, and more. Broadly, the discussion centered around two main questions: 1) What are the technical capabilities and limitations of large language models? 2) What are the societal effects of widespread use of large language models? Here, we provide a detailed summary of the discussion organized by the two themes above.

Generative Language Modeling for Automated Theorem Proving (Sep 2020; Link)

Abstract:

We explore the application of transformer-based language models to automated theorem proving. This work is motivated by the possibility that a major limitation of automated theorem provers compared to humans -- the generation of original mathematical terms -- might be addressable via generation from language models. We present an automated prover and proof assistant, GPT-f, for the Metamath formalization language, and analyze its performance. GPT-f found new short proofs that were accepted into the main Metamath library, which is to our knowledge, the first time a deep-learning based system has contributed proofs that were adopted by a formal mathematics community.

Learning to summarize with human feedback (Sep 2020; Link)

Abstract:

As language models become more powerful, training and evaluation are increasingly bottlenecked by the data and metrics used for a particular task. For example, summarization models are often trained to predict human reference summaries and evaluated using ROUGE, but both of these metrics are rough proxies for what we really care about -- summary quality. In this work, we show that it is possible to significantly improve summary quality by training a model to optimize for human preferences. We collect a large, high-quality dataset of human comparisons between summaries, train a model to predict the human-preferred summary, and use that model as a reward function to fine-tune a summarization policy using reinforcement learning. We apply our method to a version of the TL;DR dataset of Reddit posts and find that our models significantly outperform both human reference summaries and much larger models fine-tuned with supervised learning alone. Our models also transfer to CNN/DM news articles, producing summaries nearly as good as the human reference without any news-specific fine-tuning. We conduct extensive analyses to understand our human feedback dataset and fine-tuned models We establish that our reward model generalizes to new datasets, and that optimizing our reward model results in better summaries than optimizing ROUGE according to humans. We hope the evidence from our paper motivates machine learning researchers to pay closer attention to how their training loss affects the model behavior they actually want.

Language models are few-shot learners (May 2020; Link)

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.

Dota 2 with large scale deep reinforcement learning (Dec 2019; Link)

Abstract:

On April 13th, 2019, OpenAI Five became the first AI system to defeat the world champions at an esports game. The game of Dota 2 presents novel challenges for AI systems such as long time horizons, imperfect information, and complex, continuous state-action spaces, all challenges which will become increasingly central to more capable AI systems. OpenAI Five leveraged existing reinforcement learning techniques, scaled to learn from batches of approximately 2 million frames every 2 seconds. We developed a distributed training system and tools for continual training which allowed us to train OpenAI Five for 10 months. By defeating the Dota 2 world champion (Team OG), OpenAI Five demonstrates that self-play reinforcement learning can achieve superhuman performance on a difficult task.

Procgen Benchmark (Dec 2019; Link)

Abstract:

We introduce Procgen Benchmark, a suite of 16 procedurally generated game-like environments designed to benchmark both sample efficiency and generalization in reinforcement learning. We believe that the community will benefit from increased access to high quality training environments, and we provide detailed experimental protocols for using this benchmark. We empirically demonstrate that diverse environment distributions are essential to adequately train and evaluate RL agents, thereby motivating the extensive use of procedural content generation. We then use this benchmark to investigate the effects of

scaling model size, finding that larger models significantly improve both sample efficiency and generalization.

SBenchmarking safe exploration in deep reinforcement learning (Nov 2019; Link)

Abstract:

Reinforcement learning (RL) agents need to explore their environments in order to learn optimal policies by trial and error. In many environments, safety is a critical concern and certain errors are unacceptable: for example, robotics systems that interact with humans should never cause injury to the humans while exploring. While it is currently typical to train RL agents mostly or entirely in simulation, where safety concerns are minimal, we anticipate that challenges in simulating the complexities of the real world (such as human-AI interactions) will cause a shift towards training RL agents directly in the real world, where safety concerns are paramount. Consequently we take the position that safe exploration should be viewed as a critical focus area for RL research, and in this work we make three contributions to advance the study of safe exploration. First, building on a wide range of prior work on safe reinforcement learning, we propose to standardize constrained RL as the main formalism for safe exploration. Second, we present the Safety Gym benchmark suite, a new slate of high-dimensional continuous control environments for measuring research progress on constrained RL. Finally, we benchmark several constrained deep RL algorithms on Safety Gym environments to establish baselines that future work can build on.

Safety Gym (Nov 2019; Link)

Abstract:

We're releasing Safety Gym, a suite of environments and tools for measuring progress towards reinforcement learning agents that respect safety constraints while training.

GPT-2: 1.5B release (Nov 2019; Link)

Abstract:

As the final model release of GPT-2's staged release, we're releasing the largest version (1.5B parameters) of GPT-2 along with code and model weights to facilitate detection of outputs of GPT-2 models. While there have been larger language models released since August, we've continued with our original staged release plan in order to provide the community with a test case of a full staged release process. We hope that this test case will be useful to developers of future powerful models, and we're actively continuing the conversation with the AI community on responsible publication.

Fine-tuning GPT-2 from human preferences (Sep 2019; Link)

Abstract:
We've fine-tuned the 774M parameter GPT-2 language model using human feedback for various tasks, successfully matching the preferences of the external human labelers, though those preferences did not always match our own. Specifically, for summarization tasks the labelers preferred sentences copied wholesale from the input (we'd only asked them to ensure accuracy), so our models learned to copy. Summarization required 60k human labels; simpler tasks which continue text in various styles required only 5k. Our motivation is to move safety techniques closer to the general task of "machines talking to humans," which we believe is key to extracting information about human values.

Emergent tool use from multi-agent interaction (Sep 2019; Link)

Abstract:
Through multi-agent competition, the simple objective of hide-and-seek, and standard reinforcement learning algorithms at scale, we find that agents create a self-supervised autocurriculum inducing multiple distinct rounds of emergent strategy, many of which require sophisticated tool use and coordination. We find clear evidence of six emergent phases in agent strategy in our environment, each of which creates a new pressure for the opposing team to adapt; for instance, agents learn to build multi-object shelters using moveable boxes which in turn leads to agents discovering that they can overcome obstacles using ramps. We further provide evidence that multi-agent competition may scale better with increasing environment complexity and leads to behavior that centers around far more human-relevant skills than other self-supervised reinforcement learning methods such as intrinsic motivation. Finally, we propose transfer and fine-tuning as a way to quantitatively evaluate targeted capabilities, and we compare hide-and-seek agents to both intrinsic motivation and random initialization baselines in a suite of domain-specific intelligence tests.

GPT-2: 6-month follow-up (Aug 2019; Link)

Abstract:
We're releasing the 774 million parameter GPT-2 language model after the release of our small 124M model in February, staged release of our medium 355M model in May, and subsequent research with partners and the AI community into the model's potential for misuse and societal benefit. We're also releasing an open-source legal agreement to make it easier for organizations to initiate model-sharing partnerships with each other, and are publishing a technical report about our experience in coordinating with the wider AI research community on publication norms.

Generative modeling with sparse transformers (Aug 2019; Link)

Abstract:
We've developed the Sparse Transformer, a deep neural network which sets new records at predicting what comes next in a sequence—whether text, images, or sound. It uses an algorithmic improvement of the attention mechanism to extract patterns from sequences 30x longer than possible previously.

OpenAI Five defeats Dota 2 world champions (Apr 2019; Link)

Abstract:
OpenAI Five is the first AI to beat the world champions in an esports game, having won two back-to-back games versus the world champion Dota 2 team, OG, at Finals this weekend. Both OpenAI Five and DeepMind's AlphaStar had previously beaten good pros privately but lost their live pro matches, making this also the first time an AI has beaten esports pros on livestream.

Neural MMO: A massively multiagent game environment (Mar 2019; Link)

Abstract:
We're releasing a Neural MMO, a massively multiagent game environment for reinforcement learning agents. Our platform supports a large, variable number of agents within a persistent and open-ended task. The inclusion of many agents and species leads to better exploration, divergent niche formation, and greater overall competence.

The emergence of complex life on Earth is often attributed to the arms race that ensued from a huge number of organisms all competing for finite resources. We present an artificial intelligence research environment, inspired by the human game genre of MMORPGs (Massively Multiplayer Online Role-Playing Games, a.k.a. MMOs), that aims to simulate this setting in microcosm. As with MMORPGs and the real world alike, our environment is persistent and supports a large and variable number of agents. Our environment is well suited to the study of large-scale multiagent interaction: it requires that agents learn robust combat and navigation policies in the presence of large populations attempting to do the same. Baseline experiments reveal that population size magnifies and incentivizes the development of skillful behaviors and results in agents that outcompete agents trained in smaller populations. We further show that the policies of agents with unshared weights naturally diverge to fill different niches in order to avoid competition.

Language Models are Unsupervised Multitask Learners (Feb 2019; Link)

Abstract:

Natural language processing tasks, such as question answering, machine translation, reading comprehension, and summarization, are typically approached with supervised learning on taskspecific datasets. We demonstrate that language models begin to learn these tasks without any explicit supervision when trained on a new dataset of millions of webpages called WebText. When conditioned on a document plus questions, the answers generated by the language model reach 55 F1 on the CoQA dataset - matching or exceeding the performance of 3 out of 4 baseline systems without using the 127,000+ training examples. The capacity of the language model is essential to the success of zero-shot task transfer and increasing it improves performance in a log-linear fashion across tasks. Our largest model, GPT-2, is a 1.5B parameter Transformer that achieves state of the art results on 7 out of 8 tested language modeling datasets in a zero-shot setting but still underfits WebText. Samples from the model reflect these improvements and contain coherent paragraphs of text. These findings suggest a promising path towards building language processing systems which learn to perform tasks from their naturally occurring demonstrations.

How AI training scales (Dec 2018; Link)

Abstract:

In an increasing number of domains it has been demonstrated that deep learning models can be trained using relatively large batch sizes without sacrificing data efficiency. However the limits of this massive data parallelism seem to differ from domain to domain, ranging from batches of tens of thousands in ImageNet to batches of millions in RL agents that play the game Dota 2. To our knowledge there is limited conceptual understanding of why these limits to batch size differ or how we might choose the correct batch size in a new domain. In this paper, we demonstrate that a simple and easy-to-measure statistic called the gradient noise scale predicts the largest useful batch size across many domains and applications, including a number of supervised learning datasets (MNIST, SVHN, CIFAR10, ImageNet, Billion Word), reinforcement learning domains (Atari and Dota), and even generative model training (autoencoders on SVHN). We find that the noise scale increases as the loss decreases over a training run and depends on the model size primarily through improved model performance. Our empirically-motivated theory also describes the tradeoff between compute-efficiency and time-efficiency, and provides a rough model of the benefits of adaptive batch-size training.

Quantifying generalization in reinforcement learning (Dec 2018; Link)

Abstract:

In this paper, we investigate the problem of overfitting in deep reinforcement learning. Among the most common benchmarks in RL, it is customary to use the same environments for both training and testing. This practice offers relatively little insight into an agent's ability to generalize. We address this issue by using procedurally generated environments to construct distinct training and test sets. Most notably, we introduce a new environment called CoinRun, designed as a benchmark for generalization in RL. Using CoinRun, we find that agents overfit to surprisingly large training sets. We then show that deeper convolutional architectures improve generalization, as do methods traditionally found in supervised learning, including L2 regularization, dropout, data augmentation and batch normalization.

Spinning Up in Deep RL (Nov 2018; Link)

Abstract:

We’re releasing Spinning Up in Deep RL, an educational resource designed to let anyone learn to become a skilled practitioner in deep reinforcement learning. Spinning Up consists of crystal-clear examples of RL code, educational exercises, documentation, and tutorials.

Plan online, learn offline: Efficient learning and exploration via model-based control (Nov 2018; Link)

Abstract:

We propose a plan online and learn offline (POLO) framework for the setting where an agent, with an internal model, needs to continually act and learn in the world. Our work builds on the synergistic relationship between local model-based control, global value function learning, and exploration. We study how local trajectory optimization can cope with approximation errors in the value function, and can stabilize and accelerate value function learning. Conversely, we also study how approximate value functions can help reduce the planning horizon and allow for better policies beyond local solutions. Finally, we also demonstrate how trajectory optimization can be used to perform temporally coordinated exploration in conjunction with estimating uncertainty in value function approximation. This exploration is critical for fast and stable learning of the value function. Combining these components enable solutions to complex simulated control tasks, like humanoid locomotion and dexterous in-hand manipulation, in the equivalent of a few minutes of experience in the real world.

Reinforcement learning with prediction-based rewards (Oct 2018; Link)

Abstract:
We introduce an exploration bonus for deep reinforcement learning methods that is easy to implement and adds minimal overhead to the computation performed. The bonus is the error of a neural network predicting features of the observations given by a fixed randomly initialized neural network. We also introduce a method to flexibly combine intrinsic and extrinsic rewards. We find that the random network distillation (RND) bonus combined with this increased flexibility enables significant progress on several hard exploration Atari games. In particular we establish state of the art performance on Montezuma's Revenge, a game famously difficult for deep reinforcement learning methods. To the best of our knowledge, this is the first method that achieves better than average human performance on this game without using demonstrations or having access to the underlying state of the game, and occasionally completes the first level.

Large-scale study of curiosity-driven learning (Aug 2018; Link)

Abstract:

Reinforcement learning algorithms rely on carefully engineering environment rewards that are extrinsic to the agent. However, annotating each environment with hand-designed, dense rewards is not scalable, motivating the need for developing reward functions that are intrinsic to the agent. Curiosity is a type of intrinsic reward function which uses prediction error as reward signal. In this paper: (a) We perform the first large-scale study of purely curiosity-driven learning, i.e. without any extrinsic rewards, across 54 standard benchmark environments, including the Atari game suite. Our results show surprisingly good performance, and a high degree of alignment between the intrinsic curiosity objective and the hand-designed extrinsic rewards of many game environments. (b) We investigate the effect of using different feature spaces for computing prediction error and show that random features are sufficient for many popular RL game benchmarks, but learned features appear to generalize better (e.g. to novel game levels in Super Mario Bros.). (c) We demonstrate limitations of the prediction-based rewards in stochastic setups. Game-play videos and code are at this https URL.

OpenAI Five Benchmark: Results (Aug 2018; Link)

Abstract:

Yesterday, OpenAI Five won a best-of-three against a team of 99.95th percentile Dota players: Blitz, Cap, Fogged, Merlini, and MoonMeander—four of whom have played Dota professionally—in front of a live audience and 100,000 concurrent livestream viewers.

Our usual development cycle is to train each major revision of the system from scratch. However, this version of OpenAI Five contains parameters that have been training since June 9th across six major system revisions. Each revision was initialized with parameters from the previous one.

We invested heavily in “surgery” tooling which allows us to map old parameters to a new network architecture. For example, when we first trained warding, we shared a single action head for determining where to move and where to place a ward. But Five would often drop wards seemingly in the direction it was trying to go, and we hypothesized it was allocating its capacity primarily to movement. Our tooling let us split the head into two clones initialized with the same parameters.

Learning Montezuma’s Revenge from a single demonstration (Jul 2018; Link)

Abstract:

We’ve trained an agent to achieve a high score of 74,500 on Montezuma’s Revenge from a single human demonstration, better than any previously published result. Our algorithm is simple: the agent plays a sequence of games starting from carefully chosen states from the demonstration, and learns from them by optimizing the game score using PPO, the same reinforcement learning algorithm that underpins OpenAI Five.

OpenAI Five (Jun 2018; Link)

Abstract:

Our team of five neural networks, OpenAI Five, has started to defeat amateur human teams at Dota 2.

Our team of five neural networks, OpenAI Five, has started to defeat amateur human teams at Dota 2. While today we play with restrictions, we aim to beat a team of top professionals at The International in August subject only to a limited set of heroes. We may not succeed: Dota 2 is one of the most popular and complex esports games in the world, with creative and motivated professionals who train year-round to earn part of Dota’s annual $40M prize pool (the largest of any esports game).

OpenAI Five plays 180 years worth of games against itself every day, learning via self-play. It trains using a scaled-up version of Proximal Policy Optimization running on 256 GPUs and 128,000 CPU cores—a larger-scale version of the system we built to play the much-simpler solo variant of the game last year. Using a separate LSTM for each hero and no human data, it learns recognizable strategies. This indicates that reinforcement learning can yield long-term planning with large but achievable scale—without fundamental advances, contrary to our own expectations upon starting the project.

Retro Contest: Results (Jun 2018; Link)

Abstract:

The first run of our Retro Contest—exploring the development of algorithms that can generalize from previous experience—is now complete.

Though many approaches were tried, top results all came from tuning or extending existing algorithms such as PPO and Rainbow. There's a long way to go: top performance was 4,692 after training while the theoretical max is 10,000. These results provide validation that our Sonic benchmark is a good problem for the community to double down on: the winning solutions are general machine learning approaches rather than competition-specific hacks, suggesting that one can't cheat through this problem.

Learning policy representations in multiagent systems (Jun 2018; Link)

Abstract:

Modeling agent behavior is central to understanding the emergence of complex phenomena in multiagent systems. Prior work in agent modeling has largely been task-specific and driven by hand-engineering domain-specific prior knowledge. We propose a general learning framework for modeling agent behavior in any multiagent system using only a handful of interaction data. Our framework casts agent modeling as a representation learning problem. Consequently, we construct a novel objective inspired by imitation learning and agent identification and design an algorithm for unsupervised learning of representations of agent policies. We demonstrate empirically the utility of the proposed framework in (i) a challenging high-dimensional competitive environment for continuous control and (ii) a cooperative environment for communication, on supervised predictive tasks, unsupervised clustering, and policy optimization using deep reinforcement learning.

Improving language understanding by Generative Pre-Training (Jun 2018; Link)

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).

GamePad: A learning environment for theorem proving (Jun 2018; Link)

Abstract:

In this paper, we introduce a system called GamePad that can be used to explore the application of machine learning methods to theorem proving in the Coq proof assistant. Interactive theorem provers such as Coq enable users to construct machine-checkable proofs in a step-by-step manner. Hence, they provide an opportunity to explore theorem proving with human supervision. We use GamePad to synthesize proofs for a simple algebraic rewrite problem and train baseline models for a formalization of the Feit-Thompson theorem. We address position evaluation (i.e., predict the number of proof steps left) and tactic prediction (i.e., predict the next proof step) tasks, which arise naturally in tactic-based theorem proving.

Gym Retro (May 2018; Link)

Abstract:

We're releasing the full version of Gym Retro, a platform for reinforcement learning research on games. This brings our publicly-released game count from around 70 Atari games and 30 Sega games to over 1,000 games across a variety of backing emulators. We're also releasing the tool we use to add new games to the platform.

Evolved Policy Gradients (Apr 2018; Link)

Abstract:

We propose a metalearning approach for learning gradient-based reinforcement learning (RL) algorithms. The idea is to evolve a differentiable loss function, such that an agent, which optimizes its policy to minimize this loss, will achieve high rewards. The loss is parametrized via temporal convolutions over the agent's experience. Because this loss is highly flexible in its ability to take into account the agent's history, it enables fast task learning. Empirical results show that our evolved policy gradient algorithm (EPG) achieves faster learning on several randomized environments compared to an off-the-shelf policy gradient method. We also demonstrate that EPG's learned loss can generalize to out-of-distribution test time tasks, and exhibits qualitatively different behavior from other popular metalearning algorithms.

We're releasing an experimental metalearning approach called Evolved Policy Gradients, a method that evolves the loss function of learning agents, which can enable fast training on novel tasks. Agents trained with EPG can succeed at basic tasks at test time that were outside their training regime, like learning to navigate to an object on a different side of the room from where it was placed during training.

Variance reduction for policy gradient with action-dependent factorized baselines (Mar 2018; Link)

Abstract:

Policy gradient methods have enjoyed great success in deep reinforcement learning but suffer from high variance of gradient estimates. The high variance problem is particularly exasperated in problems with long horizons or high-dimensional action spaces. To mitigate this issue, we derive a bias-free action-dependent baseline for variance reduction which fully exploits the structural form of the stochastic policy itself and does not make any additional assumptions about the MDP. We demonstrate and quantify the benefit of the action-dependent baseline through both theoretical analysis as well as numerical results, including an analysis of the suboptimality of the optimal state-dependent baseline. The result is a computationally efficient policy gradient algorithm, which scales to high-dimensional control problems, as demonstrated by a synthetic 2000-dimensional target matching task. Our experimental results indicate that action-dependent baselines allow for faster learning on standard reinforcement learning benchmarks and high-dimensional hand manipulation and synthetic tasks. Finally, we show that the general idea of including additional information in baselines for improved variance reduction can be extended to partially observed and multi-agent tasks.

On first-order meta-learning algorithms (Mar 2018; Link)

Abstract:
This paper considers meta-learning problems, where there is a distribution of tasks, and we would like to obtain an agent that performs well (i.e., learns quickly) when presented with a previously unseen task sampled from this distribution. We analyze a family of algorithms for learning a parameter initialization that can be fine-tuned quickly on a new task, using only first-order derivatives for the meta-learning updates. This family includes and generalizes first-order MAML, an approximation to MAML obtained by ignoring second-order derivatives. It also includes Reptile, a new algorithm that we introduce here, which works by repeatedly sampling a task, training on it, and moving the initialization towards the trained weights on that task. We expand on the results from Finn et al. showing that first-order meta-learning algorithms perform well on some well-established benchmarks for few-shot classification, and we provide theoretical analysis aimed at understanding why these algorithms work.

Reptile: A scalable meta-learning algorithm (Mar 2018; Link)

Abstract:
This paper considers meta-learning problems, where there is a distribution of tasks, and we would like to obtain an agent that performs well (i.e., learns quickly) when presented with a previously unseen task sampled from this distribution. We analyze a family of algorithms for learning a parameter initialization that can be fine-tuned quickly on a new task, using only first-order derivatives for the meta-learning updates. This family includes and generalizes first-order MAML, an approximation to MAML obtained by ignoring second-order derivatives. It also includes Reptile, a new algorithm that we introduce here, which works by repeatedly sampling a task, training on it, and moving the initialization towards the trained weights on that task. We expand on the results from Finn et al. showing that first-order meta-learning algorithms perform well on some well-established benchmarks for few-shot classification, and we provide theoretical analysis aimed at understanding why these algorithms work.

We’ve developed a simple meta-learning algorithm called Reptile which works by repeatedly sampling a task, performing stochastic gradient descent on it, and updating the initial parameters towards the final parameters learned on that task. Reptile is the application of the Shortest Descent algorithm to the meta-learning setting, and is mathematically similar to first-order MAML (which is a version of the well-known MAML algorithm) that only needs black-box access to an optimizer such as SGD or Adam, with similar computational efficiency and performance.

Some considerations on learning to explore via meta-reinforcement learning (Mar 2018; Link)

Abstract:

We consider the problem of exploration in meta reinforcement learning. Two new meta reinforcement learning algorithms are suggested: E-MAML and E-RL². Results are presented on a novel environment we call "Krazy World" and a set of maze environments. We show E-MAML and E-RL² deliver better performance on tasks where exploration is important.

Interpretable machine learning through teaching (Feb 2018; Link)

Abstract:

Teachers intentionally pick the most informative examples to show their students. However, if the teacher and student are neural networks, the examples that the teacher network learns to give, although effective at teaching the student, are typically uninterpretable. We show that training the student and teacher iteratively, rather than jointly, can produce interpretable teaching strategies. We evaluate interpretability by (1) measuring the similarity of the teacher's emergent strategies to intuitive strategies in each domain and (2) conducting human experiments to evaluate how effective the teacher's strategies are at teaching humans. We show that the teacher network learns to select or generate interpretable, pedagogical examples to teach rule-based, probabilistic, boolean, and hierarchical concepts.

We've designed a method that encourages AIs to teach each other with examples that also make sense to humans. Our approach automatically selects the most informative examples to teach a concept—for instance, the best images to describe the concept of dogs—and experimentally we found our approach to be effective at teaching both AIs.

Learning sparse neural networks through L_0 regularization (Dec 2017; Link)

Abstract:

We propose a practical method for L_0 norm regularization for neural networks: pruning the network during training by encouraging weights to become exactly zero. Such regularization is interesting since (1) it can greatly speed up training and inference, and (2) it can improve generalization. AIC and BIC, well-known model selection criteria, are special cases of L_0 regularization. However, since the L_0 norm of weights is non-differentiable, we cannot incorporate it directly as a regularization term in the objective function. We propose a solution through the inclusion of a collection of non-negative stochastic gates, which collectively determine which weights to set to zero. We show that, somewhat surprisingly, for certain distributions over the gates, the expected L_0 norm of the resulting gated weights is differentiable with respect to the distribution parameters. We further propose the hard concrete distribution for the gates, which is obtained by "stretching" a binary concrete distribution and then transforming its samples with a hard-sigmoid. The parameters of the distribution over the gates can then be jointly optimized with the original network parameters.

As a result our method allows for straightforward and efficient learning of model structures with stochastic gradient descent and allows for conditional computation in a principled way. We perform various experiments to demonstrate the effectiveness of the resulting approach and regularizer.

Learning a hierarchy (Oct 2017; Link)

Abstract:

We've developed a hierarchical reinforcement learning algorithm that learns high-level actions useful for solving a range of tasks, allowing fast solving of tasks requiring thousands of timesteps. Our algorithm, when applied to a set of navigation problems, discovers a set of high-level actions for walking and crawling in different directions, which enables the agent to master new navigation tasks quickly.

We develop a metalearning approach for learning hierarchically structured policies, improving sample efficiency on unseen tasks through the use of shared primitives---policies that are executed for large numbers of timesteps. Specifically, a set of primitives are shared within a distribution of tasks, and are switched between by task-specific policies. We provide a concrete metric for measuring the strength of such hierarchies, leading to an optimization problem for quickly reaching high reward on unseen tasks. We then present an algorithm to solve this problem end-to-end through the use of any off-the-shelf reinforcement learning method, by repeatedly sampling new tasks and resetting task-specific policies. We successfully discover meaningful motor primitives for the directional movement of four-legged robots, solely by interacting with distributions of mazes. We also demonstrate the transferability of primitives to solve long-timescale sparse-reward obstacle courses, and we enable 3D humanoid robots to robustly walk and crawl with the same policy.

Competitive self-play (Oct 2017; Link)

Abstract:

Reinforcement learning algorithms can train agents that solve problems in complex, interesting environments. Normally, the complexity of the trained agent is closely related to the complexity of the environment. This suggests that a highly capable agent requires a complex environment for training. In this paper, we point out that a competitive multi-agent environment trained with self-play can produce behaviors that are far more complex than the environment itself. We also point out that such environments come with a natural curriculum, because for any skill level, an environment full of agents of this level will have the right level of difficulty. This work introduces several competitive multi-agent environments where agents compete in a 3D world with simulated physics. The trained agents learn a wide variety of complex and interesting skills, even though the environment themselves are relatively simple. The skills include behaviors such as running, blocking, ducking, tackling, fooling opponents, kicking, and defending using both arms and legs. A highlight of the learned behaviors can be found here.

Learning to model other minds (Sep 2017; Link)

Abstract:

Multi-agent settings are quickly gathering importance in machine learning. This includes a plethora of recent work on deep multi-agent reinforcement learning, but also can be extended to hierarchical RL, generative adversarial networks and decentralised optimisation. In all these settings the presence of multiple learning agents renders the training problem non-stationary and often leads to unstable training or undesired final results. We present Learning with Opponent-Learning Awareness (LOLA), a method in which each agent shapes the anticipated learning of the other agents in the environment. The LOLA learning rule includes a term that accounts for the impact of one agent's policy on the anticipated parameter update of the other agents. Results show that the encounter of two LOLA agents leads to the emergence of tit-for-tat and therefore cooperation in the iterated prisoners' dilemma, while independent learning does not. In this domain, LOLA also receives higher payouts compared to a naive learner, and is robust against exploitation by higher order gradient-based methods. Applied to repeated matching pennies, LOLA agents converge to the Nash equilibrium. In a round robin tournament we show that LOLA agents successfully shape the learning of a range of multi-agent learning algorithms from literature, resulting in the highest average returns on the IPD. We also show that the LOLA update rule can be efficiently calculated using an extension of the policy gradient estimator, making the method suitable for model-free RL. The method thus scales to large parameter and input spaces and nonlinear function approximators. We apply LOLA to a grid world task with an embedded social dilemma using recurrent policies and opponent modelling. By explicitly considering the learning of the other agent, LOLA agents learn to cooperate out of self-interest. The code is at this http URL.

Scalable trust-region method for deep reinforcement learning using Kronecker-factored approximation (Aug 2017; Link)

Abstract:

In this work, we propose to apply trust region optimization to deep reinforcement learning using a recently proposed Kronecker-factored approximation to the curvature. We extend the framework of natural policy gradient and propose to optimize both the actor and the critic using Kronecker-factored approximate curvature (K-FAC) with trust region; hence we call our method Actor Critic using Kronecker-Factored Trust Region (ACKTR). To the best of our knowledge, this is the first scalable trust region natural gradient method for actor-critic methods. It is also a method that learns non-trivial tasks in continuous control as well as discrete control policies directly from raw pixel inputs. We tested our approach across discrete domains in Atari games as well as continuous domains in the MuJoCo environment. With the proposed methods, we are able to achieve higher rewards and a 2- to 3-fold improvement in sample efficiency on average, compared to previous state-of-the-art on-policy actor-critic methods. Code is available at this URL.

Dota 2 (Aug 2017; Link)

Abstract:
We’ve created a bot which beats the world’s top professionals at 1v1 matches of Dota 2 under standard tournament rules. The bot learned the game from scratch by self-play, and does not use imitation learning or tree search. This is a step towards building AI systems which accomplish well-defined goals in messy, complicated situations involving real humans.

Our Dota 2 result shows that self-play can catapult the performance of machine learning systems from far below human level to superhuman, given sufficient compute. In the span of a month, our system went from barely matching a high-ranked player to beating the top pros and has continued to improve since then. Supervised deep learning systems can only be as good as their training datasets, but in self-play systems, the available data improves automatically as the agent gets better.

Gathering human feedback (Aug 2017; Link)

Abstract:
RL-Teacher is an open-source implementation of our interface to train AIs via occasional human feedback rather than hand-crafted reward functions. The underlying technique was developed as a step towards safe AI systems, but also applies to reinforcement learning problems with rewards that are hard to specify.

The release contains three main components:

- A reward predictor that can be plugged into any agent and learns to predict the actions the agent could take that a human would approve of.
- An example agent that learns via a function specified by a reward predictor. RL-Teacher ships with three pre-integrated algorithms, including OpenAI Baselines PPO.
- A web-app that humans can use to give feedback, providing the data used to train the reward predictor

Parameter Space Noise for Exploration (Jun 2017; Link)

Abstract:
Deep reinforcement learning (RL) methods generally engage in exploratory behavior through noise injection in the action space. An alternative is to add noise directly to the agent's parameters, which can lead to more consistent exploration and a richer set of behaviors. Methods such as evolutionary strategies use parameter perturbations, but discard all temporal structure in the process and require significantly more samples. Combining parameter noise with traditional RL methods allows to combine the best of both worlds. We demonstrate that both off- and on-policy methods benefit from this approach through

experimental comparison of DQN, DDPG, and TRPO on high-dimensional discrete action environments as well as continuous control tasks. Our results show that RL with parameter noise learns more efficiently than traditional RL with action space noise and evolutionary strategies individually.

Proximal Policy Optimization (Jul 2017; Link)

Abstract:

We propose a new family of policy gradient methods for reinforcement learning, which alternate between sampling data through interaction with the environment, and optimizing a "surrogate" objective function using stochastic gradient ascent. Whereas standard policy gradient methods perform one gradient update per data sample, we propose a novel objective function that enables multiple epochs of minibatch updates. The new methods, which we call proximal policy optimization (PPO), have some of the benefits of trust region policy optimization (TRPO), but they are much simpler to implement, more general, and have better sample complexity (empirically). Our experiments test PPO on a collection of benchmark tasks, including simulated robotic locomotion and Atari game playing, and we show that PPO outperforms other online policy gradient methods, and overall strikes a favorable balance between sample complexity, simplicity, and wall-time.

Hindsight Experience Replay (Jul 2017; Link)

Abstract:

Dealing with sparse rewards is one of the biggest challenges in Reinforcement Learning (RL). We present a novel technique called Hindsight Experience Replay which allows sample-efficient learning from rewards which are sparse and binary and therefore avoid the need for complicated reward engineering. It can be combined with an arbitrary off-policy RL algorithm and may be seen as a form of implicit curriculum.

We demonstrate our approach on the task of manipulating objects with a robotic arm. In particular, we run experiments on three different tasks: pushing, sliding, and pick-and-place, in each case using only binary rewards indicating whether or not the task is completed. Our ablation studies show that Hindsight Experience Replay is a crucial ingredient which makes training possible in these challenging environments. We show that our policies trained on a physics simulation can be deployed on a physical robot and successfully complete the task.

Teacher–student curriculum learning (Jul 2017; Link)

Abstract:

We propose Teacher–Student Curriculum Learning (TSCL), a framework for automatic curriculum learning, where the Student tries to learn a complex task and the Teacher automatically chooses subtasks from a given set for the Student to train on. We describe a family of Teacher algorithms that rely on the intuition that the Student should practice more those tasks on which it makes the fastest progress, i.e. where the slope of the learning curve is highest. In addition, the Teacher algorithms address the problem of forgetting by also choosing tasks where the Student's performance is getting worse. We demonstrate that TSCL matches or surpasses the results of carefully hand-crafted curricula in two tasks: addition of decimal numbers with LSTM and navigation in Minecraft. Using our automatically generated curriculum enabled to solve a Minecraft maze that could not be solved at all when training directly on solving the maze, and the learning was an order of magnitude faster than uniform sampling of subtasks.

Learning from human preferences (Jun 2017; Link)

Abstract:
For sophisticated reinforcement learning (RL) systems to interact usefully with real-world environments, we need to communicate complex goals to these systems. In this work, we explore goals defined in terms of (non-expert) human preferences between pairs of trajectory segments. We show that this approach can effectively solve complex RL tasks without access to the reward function, including Atari games and simulated robot locomotion, while providing feedback on less than one percent of our agent's interactions with the environment. This reduces the cost of human oversight far enough that it can be practically applied to state-of-the-art RL systems. To demonstrate the flexibility of our approach, we show that we can successfully train complex novel behaviors with about an hour of human time. These behaviors and environments are considerably more complex than any that have been previously learned from human feedback.

Learning to cooperate, compete, and communicate (Jun 2017; Link)

Abstract:
We explore deep reinforcement learning methods for multi-agent domains. We begin by analyzing the difficulty of traditional algorithms in the multi-agent case: Q-learning is challenged by an inherent non-stationarity of the environment, while policy gradient suffers from a variance that increases as the number of agents grows. We then present an adaptation of actor-critic methods that considers action policies of other agents and is able to successfully learn policies that require complex multi-agent coordination. Additionally, we introduce a training regimen utilizing an ensemble of policies for each agent that leads to more robust multi-agent policies. We show the strength of our approach compared to existing methods in cooperative as well as competitive scenarios, where agent populations are able to discover various physical and informational coordination strategies.

UCB exploration via Q-ensembles (Jun 2017; Link)

Abstract:
We show how an ensemble of Q^*-functions can be leveraged for more effective exploration in deep reinforcement learning. We build on well established algorithms from the bandit setting, and adapt them to the Q-learning setting. We propose an exploration strategy based on upper-confidence bounds (UCB). Our experiments show significant gains on the Atari benchmark.

OpenAI Baselines: DQN (May 2017; Link)

Abstract:
We're open-sourcing OpenAI Baselines, our internal effort to reproduce reinforcement learning algorithms with performance on par with published results. We'll release the algorithms over upcoming months; today's release includes DQN and three of its variants.

Equivalence between policy gradients and soft Q-learning (Apr 2017; Link)

Abstract:
Two of the leading approaches for model-free reinforcement learning are policy gradient methods and Q-learning methods. Q-learning methods can be effective and sample-efficient when they work, however, it is not well-understood why they work, since empirically, the Q-values they estimate are very inaccurate. A partial explanation may be that Q-learning methods are secretly implementing policy gradient updates: we show that there is a precise equivalence between Q-learning and policy gradient methods in the setting of entropy-regularized reinforcement learning, that "soft" (entropy-regularized) Q-learning is exactly equivalent to a policy gradient method. We also point out a connection between Q-learning methods and natural policy gradient methods. Experimentally, we explore the entropy-regularized versions of Q-learning and policy gradients, and we find them to perform as well as (or slightly better than) the standard variants on the Atari benchmark. We also show that the equivalence holds in practical settings by constructing a Q-learning method that closely matches the learning dynamics of A3C without using a target network or ϵ-greedy exploration schedule.

Stochastic Neural Networks for hierarchical reinforcement learning (Apr 2017; Link)

Abstract:

Deep reinforcement learning has achieved many impressive results in recent years. However, tasks with sparse rewards or long horizons continue to pose significant challenges. To tackle these important problems, we propose a general framework that first learns useful skills in a pre-training environment, and then leverages the acquired skills for learning faster in downstream tasks. Our approach brings together some of the strengths of intrinsic motivation and hierarchical methods: the learning of useful skill is guided by a single proxy reward, the design of which requires very minimal domain knowledge about the downstream tasks. Then a high-level policy is trained on top of these skills, providing a significant improvement of the exploration and allowing to tackle sparse rewards in the downstream tasks. To efficiently pre-train a large span of skills, we use Stochastic Neural Networks combined with an information-theoretic regularizer. Our experiments show that this combination is effective in learning a wide span of interpretable skills in a sample-efficient way, and can significantly boost the learning performance uniformly across a wide range of downstream tasks.

Learning to Generate Reviews and Discovering Sentiment (Apr 2017; Link)

Abstract:
We explore the properties of byte-level recurrent language models. When given sufficient amounts of capacity, training data, and compute time, the representations learned by these models include disentangled features corresponding to high-level concepts. Specifically, we find a single unit which performs sentiment analysis. These representations, learned in an unsupervised manner, achieve state of the art on the binary subset of the Stanford Sentiment Treebank. They are also very data efficient. When using only a handful of labeled examples, our approach matches the performance of strong baselines trained on full datasets. We also demonstrate the sentiment unit has a direct influence on the generative process of the model. Simply fixing its value to be positive or negative generates samples with the corresponding positive or negative sentiment.

Evolution strategies as a scalable alternative to reinforcement learning (Mar 2017; Link)

Abstract:
We explore the use of Evolution Strategies (ES), a class of black box optimization algorithms, as an alternative to popular MDP-based RL techniques such as Q-learning and Policy Gradients. Experiments on MuJoCo and Atari show that ES is a viable solution strategy that scales extremely well with the number of CPUs available: By using a novel communication strategy based on common random numbers, our ES implementation only needs to communicate scalars, making it possible to scale to over a thousand parallel workers. This allows us to solve 3D humanoid walking in 10 minutes and obtain competitive results on most Atari games after one hour of training. In addition, we highlight several

advantages of ES as a black box optimization technique: it is invariant to action frequency and delayed rewards, tolerant of extremely long horizons, and does not need temporal discounting or value function approximation.

Learning to communicate (Mar 2017; Link)

Abstract:
Our hypothesis is that true language understanding will come from agents that learn words in combination with how they affect the world, rather than spotting patterns in a huge corpus of text. As a first step, we wanted to see if cooperative agents could develop a simple language amongst themselves.

Faulty reward functions in the wild (Dec 2016; Link)

Abstract:
Reinforcement learning algorithms can break in surprising, counterintuitive ways. In this post we'll explore one failure mode, which is where you misspecify your reward function.

Universe (Dev 2016; Link)

Abstract:
We're releasing Universe, a software platform for measuring and training an AI's general intelligence across the world's supply of games, websites and other applications.

#Exploration: A study of count-based exploration for deep reinforcement learning (Nov 2016; Link)

Abstract:
Count-based exploration algorithms are known to perform near-optimally when used in conjunction with tabular reinforcement learning (RL) methods for solving small discrete Markov decision processes (MDPs). It is generally thought that count-based methods cannot be applied in high-dimensional state spaces, since most states will only occur once. Recent deep RL exploration strategies are able to deal with high-dimensional continuous state spaces through complex heuristics, often relying on optimism in the face of uncertainty or intrinsic motivation. In this work, we describe a surprising finding: a simple generalization of the classic count-based approach can reach near state-of-the-art performance on various high-dimensional and/or continuous deep RL benchmarks. States are mapped to hash

codes, which allows to count their occurrences with a hash table. These counts are then used to compute a reward bonus according to the classic count-based exploration theory. We find that simple hash functions can achieve surprisingly good results on many challenging tasks. Furthermore, we show that a domain-dependent learned hash code may further improve these results. Detailed analysis reveals important aspects of a good hash function: 1) having appropriate granularity and 2) encoding information relevant to solving the MDP. This exploration strategy achieves near state-of-the-art performance on both continuous control tasks and Atari 2600 games, hence providing a simple yet powerful baseline for solving MDPs that require considerable exploration.

RL^2: Fast reinforcement learning via slow reinforcement learning (Nov 2016; Link)

Abstract:
Deep reinforcement learning (deep RL) has been successful in learning sophisticated behaviors automatically; however, the learning process requires a huge number of trials. In contrast, animals can learn new tasks in just a few trials, benefiting from their prior knowledge about the world. This paper seeks to bridge this gap. Rather than designing a "fast" reinforcement learning algorithm, we propose to represent it as a recurrent neural network (RNN) and learn it from data. In our proposed method, RL^2, the algorithm is encoded in the weights of the RNN, which are learned slowly through a general-purpose ("slow") RL algorithm. The RNN receives all information a typical RL algorithm would receive, including observations, actions, rewards, and termination flags; and it retains its state across episodes in a given Markov Decision Process (MDP). The activations of the RNN store the state of the "fast" RL algorithm on the current (previously unseen) MDP. We evaluate RL^2 experimentally on both small-scale and large-scale problems. On the small-scale side, we train it to solve randomly generated multi-arm bandit problems and finite MDPs. After RL^2 is trained, its performance on new MDPs is close to human-designed algorithms with optimality guarantees. On the large-scale side, we test RL^2 on a vision-based navigation task and show that it scales up to high-dimensional problems.

Extensions and limitations of the neural GPU (Nov 2016; Link)

Abstract:
The Neural GPU is a recent model that can learn algorithms such as multi-digit binary addition and binary multiplication in a way that generalizes to inputs of arbitrary length. We show that there are two simple ways of improving the performance of the Neural GPU: by carefully designing a curriculum, and by increasing model size. The latter requires a memory efficient implementation, as a naive implementation of the Neural GPU is memory intensive. We find that these techniques increase the set of algorithmic problems that can be solved by the Neural GPU: we have been able to learn to perform all the arithmetic operations (and generalize to arbitrarily long numbers) when the arguments are given in the decimal

representation (which, surprisingly, has not been possible before). We have also been able to train the Neural GPU to evaluate long arithmetic expressions with multiple operands that require respecting the precedence order of the operands, although these have succeeded only in their binary representation, and not with perfect accuracy. In addition, we gain insight into the Neural GPU by investigating its failure modes. We find that Neural GPUs that correctly generalize to arbitrarily long numbers still fail to compute the correct answer on highly-symmetric, atypical inputs: for example, a Neural GPU that achieves near-perfect generalization on decimal multiplication of up to 100-digit long numbers can fail on 000000…002×000000…002 while succeeding at 2×2. These failure modes are reminiscent of adversarial examples.

Infrastructure for Deep Learning (Aug 2016; Link)

Abstract:
Deep learning is an empirical science, and the quality of a group's infrastructure is a multiplier on progress. Fortunately, today's open-source ecosystem makes it possible for anyone to build great deep learning infrastructure.

In this post, we'll share how deep learning research usually proceeds, describe the infrastructure choices we've made to support it, and open-source kubernetes-ec2-autoscaler, a batch-optimized scaling manager for Kubernetes. We hope you find this post useful in building your own deep learning infrastructure.

Generative Models (Jun 2016; Link)

Abstract:
This post describes four projects that share a common theme of enhancing or using generative models, a branch of unsupervised learning techniques in machine learning. In addition to describing our work, this post will tell you a bit more about generative models: what they are, why they are important, and where they might be going.

OpenAI Gym Beta (Apr 2016; Link)

Abstract:
We're releasing the public beta of OpenAI Gym, a toolkit for developing and comparing reinforcement learning (RL) algorithms. It consists of a growing suite of environments (from simulated robots to Atari games), and a site for comparing and reproducing results.

OpenAI Gym is compatible with algorithms written in any framework, such as Tensorflow and Theano. The environments are written in Python, but we'll soon make them easy to use from any language. We originally built OpenAI Gym as a tool to accelerate our own RL research. We hope it will be just as useful for the broader community.

Weight normalization: A simple reparameterization to accelerate training of deep neural networks (Feb 2016; Link)

Abstract:

We present weight normalization: a reparameterization of the weight vectors in a neural network that decouples the length of those weight vectors from their direction. By reparameterizing the weights in this way we improve the conditioning of the optimization problem and we speed up convergence of stochastic gradient descent. Our reparameterization is inspired by batch normalization but does not introduce any dependencies between the examples in a minibatch. This means that our method can also be applied successfully to recurrent models such as LSTMs and to noise-sensitive applications such as deep reinforcement learning or generative models, for which batch normalization is less well suited. Although our method is much simpler, it still provides much of the speed-up of full batch normalization. In addition, the computational overhead of our method is lower, permitting more optimization steps to be taken in the same amount of time. We demonstrate the usefulness of our method on applications in supervised image recognition, generative modelling, and deep reinforcement learning.

Blogs

Number of Blogs Evaluated: 105
Number of Blogs After Evaluation: 11

Introducing GPTs (Nov 2023; Link)

We're rolling out custom versions of ChatGPT that you can create for a specific purpose—called GPTs. GPTs are a new way for anyone to create a tailored version of ChatGPT to be more helpful in their daily life, at specific tasks, at work, or at home—and then share that creation with others. For example, GPTs can help you learn the rules to any board game, help teach your kids math, or design stickers.

Anyone can easily build their own GPT—no coding is required. You can make them for yourself, just for your company's internal use, or for everyone. Creating one is as easy as starting a conversation, giving it instructions and extra knowledge, and picking what it can do, like searching the web, making images or analyzing data.

New models and developer products announced at DevDay (Nov 2023; Link)

Today, we shared dozens of new additions and improvements, and reduced pricing across many parts of our platform. These include:

- New GPT-4 Turbo model that is more capable, cheaper and supports a 128K context window
- New Assistants API that makes it easier for developers to build their own assistive AI apps that have goals and can call models and tools
- New multimodal capabilities in the platform, including vision, image creation (DALL·E 3), and text-to-speech (TTS)

Frontier risk and preparedness (Oct 2023; Link)

As part of our mission of building safe AGI, we take seriously the full spectrum of safety risks related to AI, from the systems we have today to the furthest reaches of superintelligence. In July, we joined other leading AI labs in making a set of voluntary commitments to promote safety, security and trust in AI. These commitments encompassed a range of risk areas, centrally including the frontier risks that are the focus of the UK AI Safety Summit. As part of our contributions to the Summit, we have detailed our progress on frontier AI safety, including work within the scope of our voluntary commitments.

We believe that frontier AI models, which will exceed the capabilities currently present in the most advanced existing models, have the potential to benefit all of humanity. But they also pose increasingly severe risks. Managing the catastrophic risks from frontier AI will require answering questions like:

- How dangerous are frontier AI systems when put to misuse, both now and in the future?
- How can we build a robust framework for monitoring, evaluation, prediction, and protection against the dangerous capabilities of frontier AI systems?
- If our frontier AI model weights were stolen, how might malicious actors choose to leverage them?

We need to ensure we have the understanding and infrastructure needed for the safety of highly capable AI systems.

The core objectives for the Forum are:

- Advancing AI safety research to promote responsible development of frontier models, minimize risks, and enable independent, standardized evaluations of capabilities and safety.
- Identifying best practices for the responsible development and deployment of frontier models, helping the public understand the nature, capabilities, limitations, and impact of the technology.
- Collaborating with policymakers, academics, civil society and companies to share knowledge about trust and safety risks.
- Supporting efforts to develop applications that can help meet society's greatest challenges, such as climate change mitigation and adaptation, early cancer detection and prevention, and combating cyber threats.

OpenAI Red Teaming Network (Sep 2023; Link)

We're announcing an open call for the OpenAI Red Teaming Network and invite domain experts interested in improving the safety of OpenAI's models to join our efforts.

Red teaming is an integral part of our iterative deployment process. Over the past few years, our red teaming efforts have grown from a focus on internal adversarial testing at OpenAI, to working with a cohort of external experts to help develop domain specific taxonomies of risk and evaluating possibly harmful capabilities in new systems. You can read more about our prior red teaming efforts, including our past work with external experts, on models such as DALL·E 2 and GPT-4.

Today, we are launching a more formal effort to build on these earlier foundations, and deepen and broaden our collaborations with outside experts in order to make our models safer. Working with individual experts, research institutions, and civil society organizations is an important part of our process. We see this work as a complement to externally specified governance practices, such as third party audits.

The OpenAI Red Teaming Network is a community of trusted and experienced experts that can help to inform our risk assessment and mitigation efforts more broadly, rather than one-off engagements and selection processes prior to major model deployments. Members of the network will be called upon based on their expertise to help red team at various stages of the model and product development lifecycle. Not every member will be involved with each new model or product, and time contributions will be determined with each individual member, which could be as few as 5–10 hours in one year.

Introducing Superalignment (Jul 2023; Link)

We need scientific and technical breakthroughs to steer and control AI systems much smarter than us. To solve this problem within four years, we're starting a new team, co-led by Ilya Sutskever and Jan Leike, and dedicating 20% of the compute we've secured to date to this effort. We're looking for excellent ML researchers and engineers to join us. Superintelligence will be the most impactful technology humanity has ever invented, and could help us solve many of the world's most important problems. But the vast power of superintelligence could also be very dangerous, and could lead to the disempowerment of humanity or even human extinction.

While superintelligence seems far off now, we believe it could arrive this decade.

New AI classifier for indicating AI-written text (Jan 2023; Link)

We've trained a classifier to distinguish between text written by a human and text written by AIs from a variety of providers. While it is impossible to reliably detect all AI-written text, we believe good classifiers can inform mitigations for false claims that AI-generated text was written by a human: for example, running automated misinformation campaigns, using AI tools for academic dishonesty, and positioning an AI chatbot as a human.

Our classifier is not fully reliable. In our evaluations on a "challenge set" of English texts, our classifier correctly identifies 26% of AI-written text (true positives) as "likely AI-written," while incorrectly labeling human-written text as AI-written 9% of the time (false positives). Our classifier's reliability typically improves as the length of the input text increases. Compared to our previously released classifier, this new classifier is significantly more reliable on text from more recent AI systems.

As of July 20, 2023, the AI classifier is no longer available due to its low rate of accuracy. We are working to incorporate feedback and are currently researching more effective provenance techniques for text, and have made a commitment to develop and deploy mechanisms that enable users to understand if audio or visual content is AI-generated.

New and improved embedding model (Dec 2022; Link)

We are excited to announce a new embedding model which is significantly more capable, cost effective, and simpler to use.

The new model, text-embedding-ada-002, replaces five separate models for text search, text similarity, and code search, and outperforms our previous most capable model, Davinci, at most tasks, while being priced 99.8% lower.

Embeddings are numerical representations of concepts converted to number sequences, which make it easy for computers to understand the relationships between those concepts. Since the initial launch of the OpenAI /embeddings endpoint, many applications have incorporated embeddings to personalize, recommend, and search content.

Our approach to alignment research (Aug 2022; Link)

We are improving our AI systems' ability to learn from human feedback and to assist humans at evaluating AI. Our goal is to build a sufficiently aligned AI system that can help us solve all other alignment problems.

At a high-level, our approach to alignment research focuses on engineering a scalable training signal for very smart AI systems that is aligned with human intent. It has three main pillars:

1. Training AI systems using human feedback
2. Training AI systems to assist human evaluation
3. Training AI systems to do alignment research

Introducing text and code embeddings (Jan 2022; Link)

We are introducing embeddings, a new endpoint in the OpenAI API that makes it easy to perform natural language and code tasks like semantic search, clustering, topic modeling, and classification.

OpenAI Codex (Aug 2021; Link)

We’ve created an improved version of OpenAI Codex, our AI system that translates natural language to code, and we are releasing it through our API in private beta starting today.

Codex is the model that powers GitHub Copilot, which we built and launched in partnership with GitHub a month ago. Proficient in more than a dozen programming languages, Codex can now interpret simple commands in natural language and execute them on the user’s behalf—making it possible to build a natural language interface to existing applications. We are now inviting businesses and developers to build on top of OpenAI Codex through our API.

OpenAI standardizes on PyTorch (Jan 2020; Link)

We are standardizing OpenAI’s deep learning framework on PyTorch. In the past, we implemented projects in many frameworks depending on their relative strengths. We’ve now chosen to standardize to make it easier for our team to create and share optimized implementations of our models.
As part of this move, we’ve just released a PyTorch-enabled version of Spinning Up in Deep RL, an open-source educational resource produced by OpenAI that makes it easier to learn about deep reinforcement learning. We are also in the process of writing PyTorch bindings for our highly-optimized blocksparse kernels, and will open-source those bindings in upcoming months.

The main reason we’ve chosen PyTorch is to increase our research productivity at scale on GPUs. It is very easy to try and execute new research ideas in PyTorch; for example, switching to PyTorch decreased our iteration time on research ideas in generative modeling from weeks to days. We’re also excited to be joining a rapidly-growing developer community, including organizations like Facebook and Microsoft, in pushing scale and performance on GPUs.

Going forward we’ll primarily use PyTorch as our deep learning framework but sometimes use other ones when there’s a specific technical reason to do so. Many of our teams have already made the switch, and we look forward to contributing to the PyTorch community in upcoming months.