

6 areas of AI and machine learning to watch closely



Distilling a generally-accepted definition of what qualifies as artificial intelligence (AI) has become a revived topic of debate in recent times. Some have rebranded AI as “cognitive computing” or “machine intelligence”, while others incorrectly interchange AI with “machine learning”. This is in part because AI is not *one* technology. It is in fact a broad field constituted of *many* disciplines, ranging from robotics to machine learning. The ultimate goal of AI, most of us affirm, is to build machines capable of performing tasks and cognitive functions that are otherwise only within the scope of human intelligence. In order to get there, machines must be able to learn these capabilities automatically instead of having each of them be explicitly programmed end-to-end.

It’s amazing how much progress the field of AI has achieved over the last 10 years, ranging from self-driving cars to speech recognition and synthesis. Against this backdrop, AI has become a topic of conversation in more and more companies and households who have come to see AI as a technology that isn’t another 20 years away, but as something that is impacting their lives today. Indeed, the popular press reports on AI almost everyday and technology giants, one by one, articulate their significant long-term AI strategies. While several investors and incumbents are eager to understand how to capture value in this new world, the majority are still scratching their heads to figure out what this all means. Meanwhile, governments are grappling with the implications of automation in society (see Obama’s [farewell address](#)).

Given that AI will impact the entire economy, actors in these conversations represent the entire distribution of intents, levels of understanding and degrees of experience with building or using AI systems. As such, it’s crucial for a discussion on AI—including

the questions, conclusions and recommendations derived therefrom—to be grounded in data and reality, not conjecture. It's far too easy (and sometimes exciting!) to wildly extrapolate the implications of results from published research or tech press announcements, speculative commentary and thought experiments.

Here are six areas of AI that are particularly noteworthy in their ability to impact the future of digital products and services. I describe what they are, why they are important, how they are being used today and include a list (by no means exhaustive) of companies and researchers working on these technologies.

1. Reinforcement learning (RL)

RL is a paradigm for learning by trial-and-error inspired by the way humans learn new tasks. In a typical RL setup, an agent is tasked with observing its current state in a digital environment and taking actions that maximise accrual of a long-term reward it has been set. The agent receives feedback from the environment as a result of each action such that it knows whether the action promoted or hindered its progress. An RL agent must therefore balance the exploration of its environment to find optimal strategies of accruing reward with exploiting the best strategy it has found to achieve the desired goal. This approach was made popular by Google DeepMind in their work on [Atari games and Go](#). An example of RL working in the real world is the task of optimising energy efficiency for cooling Google data centers. Here, an RL system achieved a [40% reduction](#) in cooling costs. An important native advantage of using RL agents in environments that can be simulated (e.g. video games) is that training data can be generated in troves and at very low cost. This is in stark contrast to supervised deep learning tasks that often require training data that is expensive and difficult to procure from the real world.

Applications: Multiple agents [learning in their own instance of an environment with a shared model](#) or by [interacting and learning from one another in the same environment](#), learning to [navigate 3D environments like mazes](#) or city streets for [autonomous driving](#), inverse reinforcement learning to recapitulate observed behaviours by learning the goal of a task (e.g. [learning to drive](#) or endowing non-player video game characters with human-like behaviours).

Principal Researchers: Pieter Abbeel (OpenAI), David Silver, Nando de Freitas, Raia Hadsell, Marc Bellemare (Google DeepMind), Carl Rasmussen (Cambridge), Rich Sutton (Alberta), John Shawe-Taylor (UCL) and [others](#).

Companies: Google DeepMind, Prowler.io, Osaro, MicroPSI, Maluuba/Microsoft, NVIDIA, Mobileye, OpenAI.

[2. Generative models](#)

In contrast to discriminative models that are used for classification or regression tasks, generative models learn a probability distribution over training examples. By sampling from this high-dimensional distribution, generative models output new examples that are similar to the training data. This means, for example, that a generative model trained on real images of faces can output new synthetic images of similar faces. For more details on how these models work, see Ian Goodfellow's [awesome NIPS 2016 tutorial write up](#). The architecture he introduced, generative adversarial networks (GANs), are particularly hot right now in the research world because they [offer a path towards unsupervised learning](#). With GANs, there are two neural networks: a *generator*, which takes random noise as input and is tasked with synthesising content (e.g. an image), and a *discriminator*, which has learned what real images look like and is tasked with identifying whether images created by the generator are real or fake. Adversarial training can be thought of as a game where the generator must iteratively learn how to create images from noise such that the discriminator can no longer distinguish generated images from real ones. This framework is being extended to many data modalities and task.

Applications: Simulate possible futures of a time-series (e.g. for planning tasks in reinforcement learning); [super-resolution of images](#); recovering [3D structure from a 2D image](#); [generalising from small labeled datasets](#); tasks where one input can yield multiple correct outputs (e.g. [predicting the next frame in a video](#)); creating natural language in conversational interfaces (e.g. bots); [cryptography](#); semi-supervised learning when not all labels are available; [artistic style transfer](#); [synthesising music and voice](#); image in-painting.

Companies: Twitter Cortex, Adobe, Apple, Prisma, Jukedeeck*, Creative.ai, Gluru*, Mapillary*, Unbabel.

Principal Researchers: Ian Goodfellow (OpenAI), Yann LeCun and [Soumith Chintala](#) (Facebook AI Research), [Shakir Mohamed](#) and [Aäron van den Oord](#) (Google DeepMind), Alyosha Efros (Berkeley) and many [others](#).

3. Networks with memory

In order for AI systems to generalise in diverse real-world environments just as we do, they must be able to continually learn new tasks and remember how to perform all of them into the future. However, traditional neural networks are typically incapable of such sequential task learning without forgetting. This shortcoming is termed *catastrophic forgetting*. It occurs because the weights in a network that are important to solve for task A are changed when the network is subsequently trained to solve for task B.

There are, however, several powerful architectures that can endow neural networks with varying degrees of memory. These include [long-short term memory](#) networks (a

recurrent neural network variant) that are capable of processing and predicting time series, DeepMind's [differentiable neural computer](#) that combines neural networks and memory systems in order to learn from and navigate complex data structures on their own, the [elastic weight consolidation](#) algorithm that slows down learning on certain weights depending on how important they are to previously seen tasks, and [progressive neural networks](#) that learn lateral connections between task-specific models to extract useful features from previously learned networks for a new task.

Applications: Learning agents that can generalise to new environments; robotic arm control tasks; autonomous vehicles; time series prediction (e.g. financial markets, video, IoT); natural language understanding and next word prediction.

Companies: Google DeepMind, NNaisense (?), SwiftKey/Microsoft Research, Facebook AI Research.

Principal Researchers: Alex Graves, Raia Hadsell, Koray Kavukcuoglu (Google DeepMind), Jürgen Schmidhuber (IDSIA), Geoffrey Hinton (Google Brain/Toronto), James Weston, Sumit Chopra, Antoine Bordes (FAIR).

4. Learning from less data and building smaller models

Deep learning models are notable for requiring enormous amounts of training data to reach state-of-the-art performance. For example, the [ImageNet Large Scale Visual Recognition Challenge](#) on which teams challenge their image recognition models, contains 1.2 million training images hand-labeled with 1000 object categories. Without large scale training data, deep learning models won't converge on their optimal settings and won't perform well on complex tasks such as speech recognition or machine translation. This data requirement only grows when a single neural network is used to solve a problem end-to-end; that is, taking raw audio recordings of speech as the input and outputting text transcriptions of the speech. This is in contrast to using multiple networks each providing intermediate representations (e.g. raw speech audio input → phonemes → words → text transcript output; or [raw pixels from a camera mapped directly to steering commands](#)). If we want AI systems to solve tasks where training data is particularly challenging, costly, sensitive, or time-consuming to procure, it's important to develop models that can learn optimal solutions from less examples (i.e. one or zero-shot learning). When training on small data sets, challenges include overfitting, difficulties in handling outliers, differences in the data distribution between training and test. An alternative approach is to improve learning of a new task by transferring knowledge a machine learning model acquired from a previous task using processes collectively referred to as [transfer learning](#).

A related problem is building smaller deep learning architectures with state-of-the-art performance using a similar number or significantly less parameters. [Advantages would include](#) more efficient distributed training because data needs to be

communicated between servers, less bandwidth to export a new model from the cloud to an edge device, and improved feasibility in deploying to hardware with limited memory.

Applications: Training shallow networks by learning to [mimic the performance of deep networks](#) originally trained on large labeled training data; architectures with fewer parameters but equivalent performance to deep models (e.g. [SqueezeNet](#)); [machine translation](#).

Companies: Geometric Intelligence/Uber, DeepScale.ai, Microsoft Research, Curious AI Company, Google, Bloomsbury AI.

Principal Researchers: Zoubin Ghahramani (Cambridge), Yoshua Bengio (Montreal), Josh Tenenbaum (MIT), Brendan Lake (NYU), Oriol Vinyals (Google DeepMind), Sebastian Riedel (UCL).

5. Hardware for training and inference

A major catalyst for progress in AI is the repurposing of graphics processing units (GPUs) for training large neural network models. Unlike central processing unit (CPUs) that compute in a sequential fashion, GPUs offer a [massively parallel architecture](#) that can handle multiple tasks concurrently. Given that neural networks must process enormous amounts of (often high dimensional data), training on GPUs is much faster than with CPUs. This is why GPUs have veritably become the shovels to the gold rush ever since the [publication of AlexNet](#) in 2012—the first neural network implemented on a GPU. NVIDIA continues to lead the charge into 2017, ahead of Intel, Qualcomm, AMD and more recently Google.

However, GPUs were not purpose-built for training or inference; they were created to render graphics for video games. GPUs have high computational precision that is not always needed and suffer memory bandwidth and data throughput issues. This has opened the playing field for a new breed of startups and projects within large companies like Google to design and produce silicon specifically for high dimensional machine learning applications. [Improvements promised by new chip designs](#) include larger memory bandwidth, computation on graphs instead of vectors (GPUs) or scalars (CPUs), higher compute density, efficiency and performance per Watt. This is exciting because of the clear accelerating returns AI systems deliver to their owners and users: Faster and more efficient model training → better user experience → user engages with the product more → creates larger data set → improves model performance through optimisation. Thus, those who are able to train faster and deploy AI models that are computationally and energy efficient are at a significant advantage.

Applications: Faster training of models (especially on graphs); energy and data efficiency when making predictions; running AI systems at the edge (IoT devices);

always-listening IoT devices; cloud infrastructure as a service; autonomous vehicles, drones and robotics.

Companies: Graphcore, [Cerebras](#), Isocline Engineering, Google (TPU), NVIDIA (DGX-1), Nervana Systems (Intel), Movidius (Intel), Scortex

Principal Researchers: ?

6. Simulation environments

As discussed earlier, generating training data for AI systems is often challenging. What's more, AI's must generalise to many situations if they're to be useful to us in the real world. As such, developing digital environments that simulate the physics and behaviour of the real world will provide us with test beds to measure and train an AI's general intelligence. These environments present raw pixels to an AI, which then take actions in order to solve for the goals they have been set (or learned). Training in [these simulation environments](#) can help us [understand how AI systems learn](#), how to improve them, but also provide us with models that can potentially transfer to real-world applications.

Applications: [Learning to drive](#); manufacturing; industrial design; game development; smart cities.

Companies: Improbable, Unity 3D, Microsoft (Minecraft), Google DeepMind/Blizzard, OpenAI, Comma.ai, Unreal Engine, Amazon Lumberyard

Researchers: [Andrea Vedaldi](#) (Oxford)