Big-Tent AI

Stephen Boyd Mykel Kochenderfer

April 2023

These are notes for several roundtable discussions about AI, led by Stephen Boyd and Mykel Kochenderfer. The authors had many discussions with, and much input from, many others.

What is AI? Let's start at the beginning: What is AI? Some people use a narrow definition, restricting it to one particular method, so-called deep learning or neural networks. For image understanding and natural language translation, that particular method works super well. It's often what you see in the press, for example Alpha-Go, an AI system that beat the world's grandmaster Go player.

We prefer a much broader definition of AI, that includes many other techniques and methods, including statistics, data science, optimization, automatic control, sequential decision making under uncertainty, and others. These fields go way back, and their methods are used in lots of applications. Many interesting applications include components from all these technologies. We call this big-tent AI.

Here's one definition of AI. It is a set of methods and techniques that use data, sometimes a lot of data, along with some kind of mathematical model, and algorithms and powerful computers to carry out tasks like creating forecasts or predictions, simulating a system (i.e., carrying out what-if analyses), or suggesting or even automatically making good decisions in real time.

AI systems can be used in different ways. An embedded AI system can take actions by itself, with no human input. An auto-pilot on an airplane, an example of an automatic control system, works that way. In other cases, the AI system is advisory (to people): its predictions or suggestions are sent to humans, who can look into the situation in more detail, and review or approve the suggestions (or not). For example a quantitative hedge fund

might run an AI system that generates a suggested trade list every day, which is reviewed and (usually) approved by a portfolio manager. It can also be more tentative, as in a recommendation system. For example, an AI system might scan data on hundreds of potential deals, and forward a few of what it deems are the most promising ones to (human) analysts who can pore over every detail, and make the final decisions. This last is an example of what is called a Centaur (from Greek Mythology), an arrangement where an AI system and people work together. One example is the green-yellow-red system used for automated fraud detection. An AI system examines various attributes of a proposed transaction, such as what it is for, the location and time, the user's purchasing history, and so on. It then assigns the proposed transaction one of the colors Green, Yellow, or Red. Green means the AI system is quite confident the transaction isn't fraud, so the transaction goes through without human oversight. Red means the AI system is confident that the transaction is fraud, so it's stopped, again without human intervention. Yellow means it isn't sure; in this case the transaction is routed to a person who can look at it in more detail, and make the final decision.

Another example of a sophisticated AI recommendation system is drug discovery. An AI algorithm predicts whether or not a compound will bind or interact with some biological target, based on some gross features of it. It can scan millions of candidate molecules, to identify a few hundred that seem most promising. These are tested in petri dishes; the ones that show promise there might move on to mice, and if we're really lucky, one or two head on to human trials.

Let's look at some of these AI tasks in greater detail.

Forecasting and prediction. A forecaster or predictor guesses future values of some quantity, given the data it has access to now, such as current or past values. Examples in finance include predicting market prices or trade volumes, interest rates, future income or liabilities, or guessing if the spread for lending a security will be trending up or down over the next week. A predictor can predict one number, or a set of related numbers, such as the market volume over the next 5 trading days. It can also be statistical in nature, for example giving a confidence range for its predictions. Roughly speaking, a predictor might report its own confidence in its predictions.

If you've got a lot of data, it's easy to validate a predictor, by holding out some data where you know what actually happened, and testing your

predictor on that. It's much more challenging when you don't have a lot of data, or the data you have is too old to be relevant.

Anomaly detection. An anomaly detector tells us when observed data (potentially consisting of many quantities) does not 'look like' previously observed data, or some quantities look out of place in the context of the others. The simplest method observes previous values, and sounds an alarm when the new data point exceeds 99% of those values; sophisticated methods take into account complex relations among the quantities before deciding if a new data point really is an anomaly. Anomaly detectors can be used to scan incoming data, flagging suspicious values for human review. When applied to the data used to make a prediction, an anomaly detector can warn us that the current values are very much unlike anything it has seen before, which probably means its prediction shouldn't be trusted too much. An anomaly detector can look at the input and output of a predictor, along with the actual value that occurs, and warn us that the relation between them doesn't look like it has in the past. Here it is acting as a run-time monitor. The run-time monitor may not be directly contributing to P&L, but it can save you on the (hopefully) rare occasions when things change or break. In finance, an anomaly can be an opportunity, for example when you find that something is mis-priced.

An anomaly detector can be validated using several methods. You can use held out data, or intentionally create anomalies in data to see if they are caught.

Simulation. Simulation is another big class of AI methods, related to predictions, and super useful. Here the goal is to predict how something will evolve over time, given various things that affect it. A simulation can be deterministic, where it makes a single guess of what is going to happen. For example, we can simulate the path of an airplane over the next minute, given how the control surfaces are manipulated over that time period. Another example is in circuit design, where we predict the behavior of the circuit to various inputs, under various conditions. A simulation can also be statistical, and give multiple plausible trajectories of what might happen, given the inputs.

These days most high-end engineering is done via simulation. Integrated circuits with many billions of parts are designed and checked using simulators

(as opposed to actual testing of prototype circuits); cars and airplanes are designed almost entirely in silico, with sophisticated simulations predicting wind flow, lift and drag, and performance metrics.

Simulation is also widely used in economics and finance. A very simple example is a plot showing you how your 401K balance might grow over the next few decades, with bands indicating uncertainty. In quantitative hedge funds, a sophisticated simulation of how a trading algorithm will perform is called a backtest. The goal is to see how your trading algorithm would do, on real historical, or simulated statistical, inputs. A simulator allows us to turn on and off various effects, or to shock or stress test a system, to see how it might react. An example is simulation of an airplane landing, when the wind direction or speed changes dramatically right as it touches down. Another example is a simulation of what might happen if there were a historically unprecedented jump in the USD/Euro exchange rate next year.

Statistical simulators were recently rebranded as so-called generative models. This means they can generate fake results, but fake results that look like real ones. An example is a simulator that generates long term predictions of how interest rates and yield curves are going to move, possibly up to a decade or more in the future. A very creepy example is a web site called This person does not exist, which generates high quality photo realistic images of people, who don't exist; they are generated photos that, in this case literally, look like real data.

Simulators are super valuable in practice. They can be used to forecast what will happen under various futures. This can be used for situational awareness, and might possibly spur action if you find that a poor outcome is more likely than you might have thought or are comfortable with. Simulation can be used retrospectively to see how we might have done, had we gone back in time and made a different choice three months ago. (This isn't immediately actionable, of course, but it is very useful information.) Retrospective simulation can be used to create an attribution system, which attributes an outcome (say, P&L) to various aspects or parts of your system.

Statistical simulation can be used to assess risk. We carry out a very large number of simulations, and use the fraction of times that an adverse outcome occurs in these simulations as our estimate of the probability of it occuring. (Of course the resulting estimate is only as accurate as the simulation itself.) A technique called importance sampling generates samples that are more challenging, and then corrects for this skewed sampling, to get good estimates of the likelihood of rare events.

Another use of simulation is for sophisticated monitoring of a complex system. To do this we simulate a system in real time, as the system is running. At each point in time we measure or observe what really happens, and also, what the simulator thought would happen. Typically these would be close; if they start to differ considerably, someone gets beeped, because something has either changed, or is broken, or is failing. In this context, a simulator is called a "digital twin". Detection of a problem via simulation with a digital twin can happen well before the anomalous behavior is otherwise noticeable. You can build sophisticated early warning systems using these techniques.

How do you validate or test a simulator? We can of course check them on historical data, where we know what actually happened. We can apply statistical tests that tell us whether the simulated data "looks like" real data (in the statistical sense). A very important additional check is that the simulations should "look right" to experts. We can use what we call the Turing test: We show experts simulated data alongside real historical data, and challenge them to tell us which is which. If you're fooling the experts 50% of the time, that's great. (You get a gold star when the expert not only selects the fake one as real, but also says "Ah yes, I remember exactly when that happened!")

Optimization. Optimization is another set of powerful AI methods. In optimization, often called mathematical optimization, we start with a model of how some action or choice or input might affect the outcome (which can be statistical), and also give a metric or score on outcomes. Then an algorithm is used to find an action that does best, i.e., achieves the highest possible score on the outcome. This is a focus of our research.

Optimization is used for lots of applications, such as scheduling, allocating resources, creating a trajectory or path for a vehicle, or designing a circuit or a bridge. For example in creating a trajectory for a drone, we might use optimization to create a trajectory that is smooth, avoids obstacles, is energy efficient, and achieves other objectives.

Optimization is also used to create predictors and anomaly detectors. It's a core AI technology. In what might appear to be circular, the models used in optimization are often predictors or simulators, which themselves were designed using . . . optimization. (It only appears to be circular; it's not really.)

There are many examples in finance and economics. For example, your

action might be to choose a portfolio (or determine a trade list); the metric would include a combination of (estimated) return, risk, transaction cost, and others, for example, tax consequences. Or how much your endowment should commit to various alternative investment classes this coming quarter. Or how to schedule the purchases of a large quantity of some stock over some period, taking into account transaction cost, risk, and information leakage.

Here's a secret. Optimization methods that were used a decade or two ago, say by the military or large companies, we can now run on our phones. Some people know this, but far fewer than you might imagine.

Here's another secret. The models used for optimization are often much simpler than the models used for simulation. We refer to these as low fidelity and high fidelity models. The optimizer uses the low fidelity model to come up with good decisions; but we check these (using simulation!) with the high fidelity model.

Optimization is a big field, and there are many methods for carrying it out. Some are suitable for general large and complex problems, but can't guarantee that they find the absolute best choice. Nevertheless such methods are quite useful, and very widely used. Some types of optimization problems can be solved very effectively, with special methods that reliably find the best action, quickly. (Stephen's specialty is convex optimization, an example of this.) These problems and methods are suitable for embedded systems, where an optimization algorithm is embedded in a bigger system, and proposes actions that are carried out with minimal or no human supervision. One example is a resource allocator, such as one used in the operation of a data center, that automatically adjusts resource allocations to some processes, as availability, demand, or other parameters change.

Sequential decision making, a.k.a. automatic control. Finally we get to the last AI technology: Sequential decision making, which combines several of the ideas described above. As in optimization, we seek best actions and are judged by some metric; but here, we include the idea that we will be taking a sequence of actions, in each case, based on what we know at that time. This allows us to model recourse or mid-course correction. In sequential decision making we also model uncertainty in the outcome. To emphasize this, it is sometimes called sequential decision making under uncertainty.

There are many examples of sequential decision making. Carrying out daily or quarterly trades in a portfolio is one; landing a rocket is another.

Executing a large trade is yet another.

One version of sequential decision making has been around for hundreds of years under a different name: automatic control. Control systems are used in pretty much everything, such as cars and rockets, motors and generators, robots, manufacturing systems, and many others. Of course the control systems used in the 1950s, or the ones used on current drones, don't look that much like a modern trading algorithm, but in spirit, they are the same.

There are many methods for doing sequential decision making. Some very effective methods rely on solving an optimization problem to decide on the action to take. The optimization problem will include a term that is the current or immediate reward or score for the action; and also, a term that takes into account the longer term effect of the action. For example, in maintenance, our actions (such as pro-actively replacing a part that still works) have an immediate cost; but that may be more than balanced by the second term, which takes into account long term reduction of future costs.

One effective method for sequential decision making is called model predictive control (MPC). The idea is very natural and simple to explain, and it uses many of the ideas described above. To decide on the current action, MPC creates a plan of actions over some period of time, starting from the current time out to some future horizon. The plan is found using optimization, a model of how the system evolves, and forecasts of future quantities. After creating a plan of action, it then executes the first action in the plan. At the next time step, the whole process is repeated, this time with updated forecasts, a new estimate or measurement of where we are now, and so on.

Here's another secret: MPC still works really well when the forecasts of future quantities aren't very accurate. For example a common forecast is that the quantity will be constant over the planning horizon, equal to its current value. For the steering system of a car, this means that you'll maintain a constant speed and turning radius over, say, the next 3 seconds. It's obviously not very accurate, but it works really, really well in practice.

MPC is widely used in many application areas, often under different names. It's used in supply chain optimization, to order new product and determine how much to ship from warehouses to fulfillment centers. It's used in chemical refineries. It's also used in typical control applications like autopilots, self-driving cars, and weapons systems. It's used to land the first stages of SpaceX's rockets. (Indeed, using optimization software from Boyd's research group.) It's used in finance, for example to execute a large trade, or manage a fixed income fund. (In that context, it's called multi-period

portfolio optimization, to distinguish it from the simple method that uses optimization, but does not find a full plan of action.)

The broad types of methods we've mentioned here (prediction, simulation, optimization, and sequential decision making) are very related, and there's no sharp dividing line between any of them. They are practiced by people who mostly don't suspect that what they are doing is actually quite related to what others, in what would seem to be far away fields, are doing. We know bond portfolio managers and we know rocket engineers. Very few of them could speak to each other (beyond about weather, say), and each would probably deny that there is any connection at all between their jobs. But they'd be wrong.

The benefits of AI. When you apply AI, you can take your benefit several ways. The most obvious and dramatic cases are when the AI system does something better than a human expert, for example in chess or Go, or classifying medical radiology images. Or guiding a vehicle. Here you take the benefit in the obvious way: doing better.

But in some cases the AI system doesn't do any better than people, who are often rather good at what they do, especially if they've been doing it for a while. But the AI system is fast and automatic, and can be applied to lots more problems than a team of people could. This opens the door to mass or hyper customization, for example where many many people can have (almost) the benefit of a team of expert advisors. So here you're taking the benefit in scaling.

In some applications, the AI system does not perform better or faster than a person. But it's more reliable, and doesn't get bored or tired or hungry or distracted. Here you're taking the benefit in reliability, safety, and repeatability.

Of course in practice the benefit of AI will be some combination of these.

Risk, pitfalls, and challenges. There are some big and important questions around AI risk, governance, and liability. These are starting to play out with self-driving cars and some medical AI applications. It's reasonable to expect that we'll see it in finance too. We are not experts on these topics, so we'll just make a few general comments about it.

First, we've heard people say that AI works well on average (because that's what it's trained to do), but not necessarily with that once-a-decade

condition. They conclude it's fine to use it to serve ads, but not, say, to guide surgery or control an airplane. This might be correct if you take the narrow view of AI as deep learning; but in big-tent AI, it's not correct. Indeed there are many applications where the main benefit of using AI is safety under unexpected situations. An example is found in electric vehicles, where turning can be accomplished by redundant means: turning the front wheels, or delivering differential torque to the wheels. If you use real-time embedded optimization to determine the actions needed to turn the car, and your right front tire is on ice, the optimizer will just figure another way to accomplish the turn. (Modern aircraft are also a bit redundant, which is a good thing.)

A more extreme example is the recently developed (and adopted) ACAS X system, which advises pilots of impending mid-air collisions. Almost by definition, this AI system is exclusively focussed on extremely rare events. It delivers its benefits in enhanced safety from very rare but very costly outcomes. Mykel was one of the key developers of ACAS X.

Second, there are risks in using AI systems, some obvious and others less so. If the system evolves according to Newtonian mechanics or the laws of electricity and magnetism, predictions are good and are likely to continue being good. In other cases, when the models are fit using data, and especially data that involve people (e.g., all of economics and finance), there's no fundamental reason to believe that a model will continue to work well. It's important to understand this risk, and to have a system in place that closely monitors the health of the AI system.

In some applications, transparency and interpretability are very important. In these applications, it's not enough to just make good predictions, or good suggestions for actions to take; we have to understand why. In some cases there is a strong trade off, with obscure models making better predictions than ones we understand. When you use convex optimization (one of our specialties), even complex decisions can be understood in terms of the objective function and constraints; we can audit a suggestion, and understand why the algorithm is making it. Roughly speaking, such methods, when well designed, never do anything crazy.

One pitfall we are hearing more and more about is how fitting models to data that includes biases unsurprisingly results in models that encapsulate (and propagate) these biases. At the very least we need to be keenly aware of this. There are some new methods on the horizon that are meant to detect this problem, or to correct it.

We'd like to add one more risk we should be aware of. In recent years

many researchers and an even greater number of students have focussed on the shiny new toy: neural networks. For some applications they perform stunningly well, and there is absolutely no doubt that these methods are powerful ones in the AI toolbox, and deserve a healthy allocation in any AI portfolio. But a growing number of these researchers have not studied, and in some cases are not even aware of, other methods that we think of as critical parts of big-tent AI, like optimization, statistics, and control. As a result I see people using very complex and obscure methods to achieve goals that simpler (and more effective) methods could readily handle. That's bad engineering, and increases the risks of using these methods.

Summary. We'll end on a more upbeat note. When AI methods are used appropriately, they can deliver many benefits, and not just the obvious one of carrying out a task better than humans can. They can also deliver much benefit in scaling, or in reliability and robustness. They can deliver benefits as embedded systems, or as human-in-the-loop recommendation systems. They can also deliver benefits as sophisticated system health monitors and early warning systems. It's an exciting time, and we think we've just scratched the surface.

At this point, we'd like to open it up for questions and discussion. We can talk about the various applications of AI that we've encountered or been involved with, details of how the SpaceX first stages land, or anything else on your minds.