Fooling the machine

The Byzantine science of deceiving artificial intelligence.

BY DAVE GERSHGORN MARCH 30, 2016

TECHNOLOGY

SHARE

In the early 1900s, Wilhelm von Osten, a German horse trainer and mathematician, told the world that his horse could do math. For years, Von Osten traveled Germany giving demonstrations of this phenomenon. He would ask his horse, Clever Hans, to compute simple equations. In response, Hans would tap his hoof for the correct answer. Two plus two? Four taps.

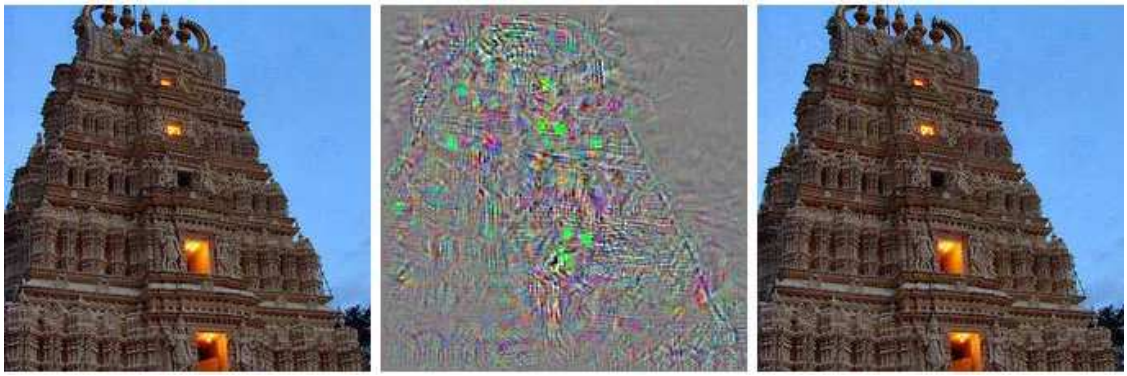
But scientists did not believe Hans was as clever as Von Osten claimed. An extensive study, coined the Hans Commission, was conducted by psychologist Carl Stumpf. He found that Clever Hans wasn't solving equations, but responding to visual cues. Hans would tap up to the correct number, which was usually when his trainer and the crowd broke out in cheers. And then he would stop. When he couldn't see those expressions, he kept tapping and tapping.

There's a lot that computer science can learn from Hans today. An accelerating field of research suggests that most of the artificial intelligence we've created so far has learned enough to give a correct answer, but without truly understanding the information. And that means it's easy to deceive.

Machine learning algorithms have quickly become the all-seeing shepherds of the human flock. This software connects us on the internet, monitors our email for spam or malicious content, and will soon drive our cars. To deceive them would be to shift tectonic underpinnings of the internet, and could pose even greater threats for our safety and security in the future.

Small groups of researchers—from Pennsylvania State University to Google to the U.S. military— are devising and defending against potential attacks that could be carried out on artificially intelligent systems. In theories posed in the research, an attacker could change what a driverless car sees. Or, it could activate voice recognition on any phone and make it visit a website with malware, only sounding

like white noise to humans. Or let a virus travel through a firewall into a network.



On the left, the unaltered image shows a building. The right image is altered to be seen as an ostrich by deep neural network-based image recognition software. The center image shows the slight distortions being made to the original picture in order to deceive the algorithm. *Christian Szegedy*

Instead of taking the controls of a driverless car, this method shows it a kind of a hallucination—images that aren't really there.

These attacks use adversarial examples: images, sounds, or potentially text that seems normal to human viewers, but are perceived as something else entirely by machines. Small changes made by attackers can force a deep neural network to draw incorrect conclusions about what it's being shown.

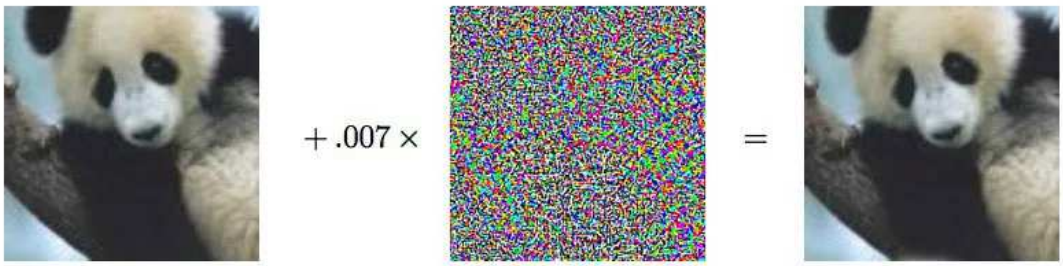
"Any system that uses machine learning for making security-critical decisions is potentially vulnerable to these kinds of attacks," said Alex Kantchelian, a researcher at Berkeley University who studies adversarial machine learning attacks.

But knowing about this early on in the development of artificial intelligence also gives researchers the tools to understand how to fix the gaps. Some have already begun to do so, and say their algorithms are actually more efficient because of it.

Most mainstream A.I. research today involves deep neural networks, which build on the larger field of machine learning. Machine learning techniques use calculus and statistics to make software we all use, like spam filters in email and Google search. Over the last 20 years, researchers began applying these techniques to a new idea called neural networks, a software structure meant to mimic the human brain. The general idea is to decentralize computing over thousands of little equations (the "neurons"), which take data, process it, and pass it onto another layer of thousands of little equations.

These artificial intelligence algorithms learn the same way machine learning has worked, which is the same way humans learn. They're shown examples of things and given labels to associate with what they're shown. Show a computer (or a child) a picture of a cat, say that's what a cat looks like, and the algorithm will learn what a cat is. To identify different cats, or cats at different angle, the computer needs to see thousands to millions of pictures of cats.

Researchers found that they could attack these systems with purposefully deceptive data, called adversarial examples.

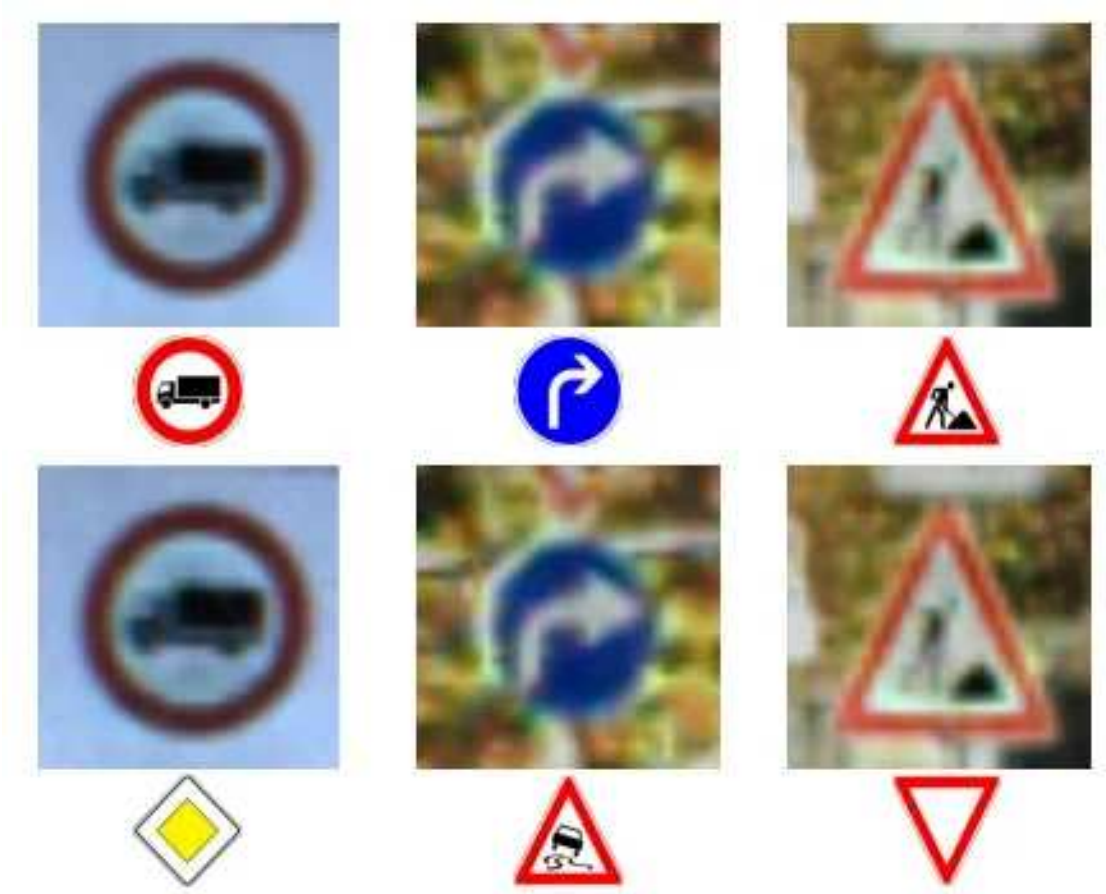


> In a 2015 paper, Google researchers showed it was possible to make deep neural networks classify this image of a panda as a gibbon, by applying light distortion. *Christian Szegedy*

“We show you a photo that’s clearly a photo of a school bus, and we make you think it’s an ostrich,” says Ian Goodfellow, a researcher at Google who has driven much of the work on adversarial examples.

By altering the images fed into a deep neural network by just four percent, researchers were able to trick it into misclassifying the image with a success rate of 97 percent. Even when they did not know how the network was processing the images, they could deceive the network with nearly 85 percent accuracy. That latter research, tricking the network without knowing its architecture, is called a black box attack. This is the first documented research of a functional black box attack on a deep learning system, which is important because this is the most likely scenario in the real world.

In the paper, researchers from Pennsylvania State University, Google, and the U.S. Army Research Laboratory actually carried out an attack against a deep neural network that classified images, supported on MetaMind, an online tool for developers. The team built and trained the network they were attacking, but their attacking algorithm operated independent of that architecture. With the attacking algorithm, they were able to force a black-box algorithm to think it was looking at something else with an accuracy up to 84.24 percent.



> The top row shows the original image with the corresponding classification. The bottom row shows that the network was successfully tricked into thinking each sign was different from the original. *Nicolas Papernot*

The act of showing incorrect information to machines is not new, but Doug Tygar, a professor at Berkeley University who has studied adversarial machine learning for 10 years, says that this attack technique has been translated from simpler machine learning to the more complex deep neural nets. Malicious attackers have used this technique on things like spam filters for years.

Tygar's research stems from a 2006 paper on adversarial attacks against machine learning networks, which he expanded in 2011 with other researchers from UC Berkeley and Microsoft Research. The Google team that pioneered the application to deep neural nets published their first paper on it in 2014, two years after they discovered the possibility of the attack. They wanted to make sure that it was actually possible, and not an anomaly. They published another paper in 2015, which found a way to guard networks and make them more efficient, and Ian Goodfellow has since consulted on other papers in the field, including the black box attack.

Security researchers call the larger idea of unreliable information Byzantine data, and through this procession of research it has arrived at deep learning. The term Byzantine data comes from the Byzantine Generals Problem, a thought experiment in computer science in which a group of generals must coordinate their attack by messenger, but are unsure of traitors in their company. They, therefore, cannot trust the information being supplied from their peers.

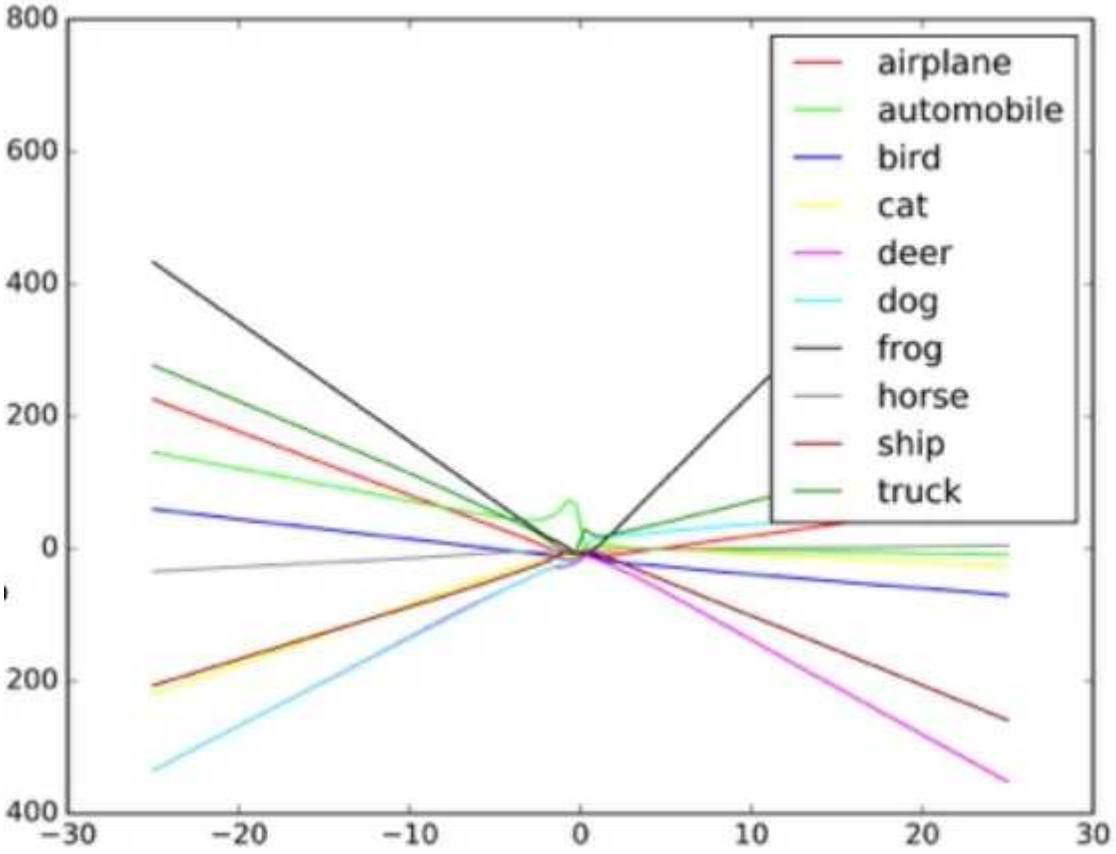
"These algorithms are defined to handle random noise, they're not crafted to handle Byzantine data," says Tygar.

To understand how these attacks work, Goodfellow suggests imagining the neural network like a scatter plot.

Each point on the scatter plot represents one pixel of an image that's being processed by the network. The network would normally try to draw a line through the data that's a best fit aggregate of where each point lies. This is a little more complicated than it sounds, because each pixel has more than one value to the

network. In reality, it's a complex, multidimensional graph that the computer must sort through.

But in our simple analogy of a scatter plot, the shape of the line drawn through the data dictates what the network thinks it's seen. To successfully attack these systems (by forcing them to misclassify inputs), researchers just have to change a small minority of these points, guiding the network to draw a conclusion that really isn't there. These altered points are past what the network would consider familiar, so it makes mistakes. In the example of making a bus look like an ostrich, the school bus photo is peppered with pixels in a pattern designed to be uniquely characteristic of the photos of ostriches that the network is familiar with—not a visible outline, but when the algorithm processes and simplifies the data, the extreme ostrich data points are seen as a valid option for classification. In the black box scenario, researchers tested inputs to ascertain how the algorithm saw certain objects.



An example of how an image classifier would draw different lines based on different objects in an image. Adversary examples would been seen as extreme values on the graph. *Ian Goodfellow*

By giving false inputs to an image classifier, and seeing what decisions the machine made, the research team was able to reverse engineer the algorithm to fool the kind of image recognition system potentially used in self-driving cars to identify a stop sign as a yield sign. And once they figured out how the exploit worked, they were able to engineer a way so they could make the machine see anything they wanted.

Researchers said that this kind of attack could either be injected straight into the image system bypassing the camera, or the manipulation could even be applied over the sign in the real world.

However, Columbia University security researcher Allison Bishop says that this kind of attack could be unrealistic, depending on the kind of system the driverless car has in place. If the attackers already had access to the camera's feed, she says, they would be able to give any input they want.

"If I can bypass the input to the camera, I don't need to work that hard." she said. "You can just show it a stop sign."

The other method of attack, which would be not bypassing the camera but painting the perturbation on the sign itself, seems like as much of a stretch to Bishop. She doubts that a camera with low resolution,

like the ones used in driverless cars today, would be able to read such slight distortions on the sign.

The left image is unaltered and would be classified as a school bus, while the right would be classified as an ostrich. The middle image shows the distortions made to the adversarial example. *Christian Szegedy*

Two groups, one at Berkeley University and another at Georgetown University, have successfully developed algorithms that can issue speech commands for digital personal assistants, like Siri and Google Now, in the form of bursts of sound unrecognizable to human ears. To a human, these commands just sound like random white noise, but they could be used to tell a voice-activated assistant like Amazon's Alexa to do things that its owner never intended.

Nicholas Carlini, one of the Byzantine audio researchers, says that their tests have been able to activate open source audio recognizers, Siri, and Google Now in their tests, with accuracy on all three more than 90 percent.

The noise sounds like a science-fiction alien transmission. It's a garbled mix of white noise and human voice, but certainly unrecognizable as a command.

With this attack, any phone that hears the noise (they have to specifically target iOS or Android) could be unknowingly forced to visit a webpage that plays the noise, and thus infect other phones near it, Carlini says. In that same scenario, the webpage could also silently download malware onto the device. There's also the possibility these noises could be played over the radio, hidden in white noise or background audio.

These attacks can happen because the machine is trained to think that there's readable or important data in almost every input, and also that some things are more common than others, says Goodfellow.

It's easier to fool the network into thinking it's seeing a common object, because it thinks that it should be seeing it more commonly. That's why Goodfellow and a separate group at University of Wyoming are able to make the network classify images when there's nothing there, by making it identify white noise, randomly generated black and white imagery.

In Goodfellow's research, random white noise he put through the network was most often classified as a horse. This coincidentally brings us back to Clever Hans, our not so mathematically-gifted horse from earlier.

Much like Clever Hans, Goodfellow says these neural networks aren't really learning certain ideas, but just how to recognize when they finds the right idea. The distinction is slight, but important. This lack of foundational knowledge makes it easy to maliciously recreate the experience of finding the "right" outcome for the algorithm, which is actually a false answer. To understand what is, the machine also must understand what is not.

Goodfellow found that when he trained his image classifying networks with both the natural images, and the doctored images (specifying that they were fake), he not only could reduce the efficiency of the attack by more than 90 percent, but the network was better at its original task.

“When you start forcing them to explain the really unusual adversarial examples, it might come up with a even more robust explanation of what the underlying concepts are,” Goodfellow says.

The two audio groups have also used the same approach as the Google researchers to patch language recognition systems against their own attacks, by retraining their networks. They’ve achieved similar levels of success, with more than 90 percent reduction in attack efficiency.

It’s no surprise that this area of research has garnered interest from the United States military. In fact, the Army Research Laboratory actually sponsored at least two of the most recent papers, including the black box attack. While the Army Lab is proactive in funding research, this doesn’t mean the tech is in active development for use in warfare. According to a spokesperson, research usually takes upwards of 10 years to make its way into soldier’s hands.

Ananthram Swami, a researcher with the U.S. Army Research Laboratory, has had varying levels of participation in recent papers concerning adversarial attacks. The Army’s interest lies in the detection and stopping of purposefully deceptive data, in an age where not all sources of information can be vetted properly. Swami points to the bevy of data accessible from public sensors placed by universities and open source projects.

“We don’t necessarily control all that data. It’s probably fairly easy for an adversary to fool us, to deceive us,” Swami said. “Some of that may be benign, some of that may not be.”

He also says that as the Army has a vested interest in autonomous robots, tanks, and other vehicles, so this research is obvious. By studying this now, the Army would have a headstart for systems in the field that were immune to potential adversarial attacks.

But any group in that uses deep neural networks, which is a quickly growing faction, should have concerns about the potential of adversarial attacks. While machine learning and artificial intelligence systems are still in their infancy, we’re at a dangerous time where security oversights can have drastic results. Many companies are placing highly volatile information in the hands of artificially intelligent systems, which have not endured the scrutiny of time. Our neural networks are simply too young for us to know everything about them.

A similar oversight led to Tay, Microsoft’s Twitter chatbot that quickly turned into a genocidal racist. A torrent of malicious data, and a foreseeably terrible “repeat after me” function, led Tay to deviate wildly from her original programming. The bot was hijacked by bad training data from the wild, and serves as a handy example for what can happen when machine learning is poorly implemented.

Kantchelian says that he doesn’t think the door is completely closed for any of these attacks, even with the promising research from the Google team.

“At least in computer security, unfortunately the attackers are always ahead of us,” Kantchelian says. “So it’s going to be a little dangerous to say we solved all the problems of adversarial machine learning by retraining.”

ARMY | ARTIFICIAL INTELLIGENCE | BERKELEY | GOOGLE | MICROSOFT