December 27, 2018 | Ben Dickson

The security threats of neural networks and deep learning algorithms

Source: Depositphotos

This article is part of Demystifying AI, a series of posts that (try to) disambiguate the jargon and myths surrounding AI.

History shows that cybersecurity threats evolve along with new technological advances. Relational databases brought SQL injection attacks, web scripting programming languages spurred cross-site scripting attacks, IoT devices ushered in new ways to create botnets, and the internet in general opened a Pandora's box of digital security ills. Social media created new ways to manipulate people through micro-targeted content delivery and made it easier to gather information for phishing attacks. And bitcoin enabled the delivery of crypto-ransowmare attacks.

The list goes on. The point is, every new technology entails new security threats that were previously unimaginable. And in many cases, we learned of those threats in hard, irreversible ways.

Recently, deep learning and neural networks have become very prominent in shaping the technology that powers various industries. From content recommendation to disease diagnosis and treatment and self-driving vehicles, deep learning is playing a very important role in making critical decisions.

Now the question is, what are the security threats unique to neural networks and deep learning algorithms? In the past few years, we've seen examples of ways malicious actors can use the characteristics and functionalities of deep learning algorithms to stage cyberattacks. While we still don't know of any large-scale deep learning attack, these examples can be prologue to what is to come. Here's what we know.

First some conditions

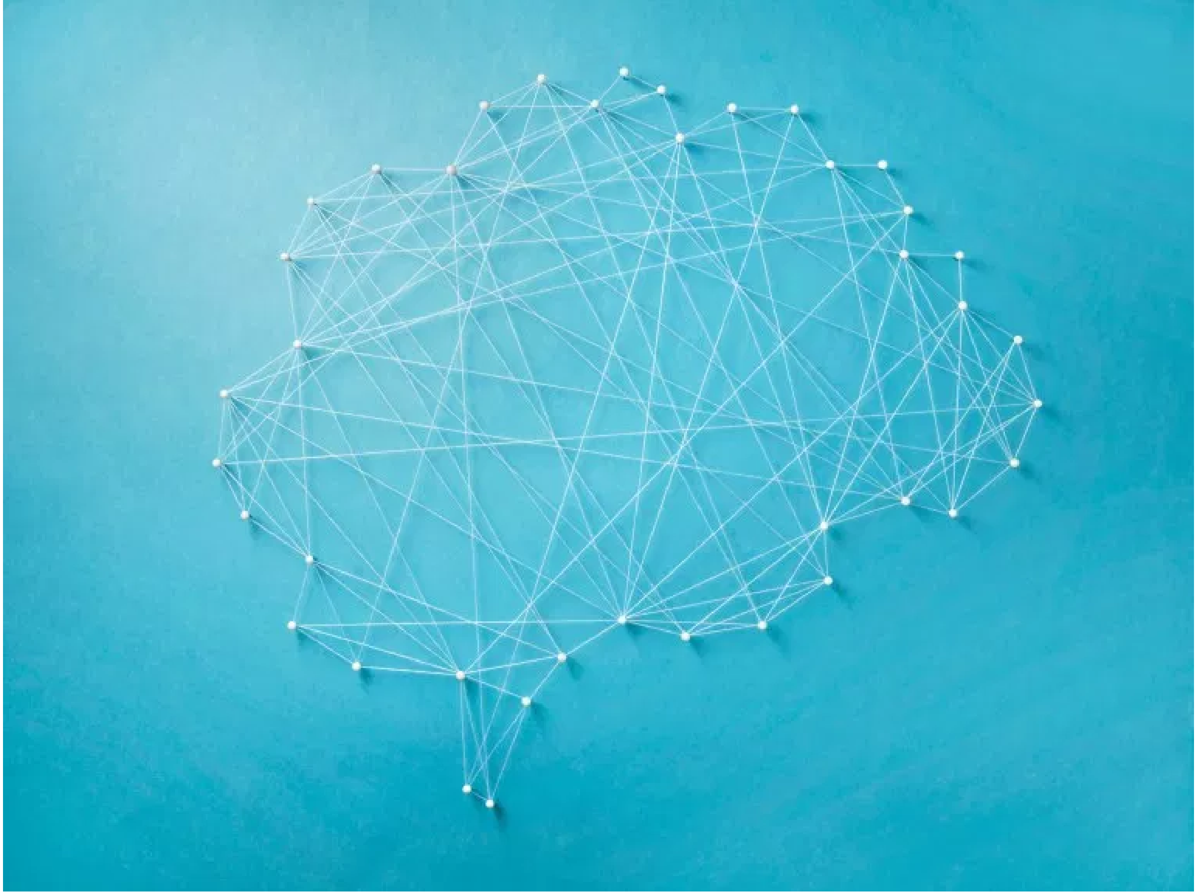
Deep learning and neural networks can be used to amplify or enhance some types of cyberattacks that already exist. For instance, you can use neural networks to replicate a target's writing style in phishing scams. Neural networks might also help automate the finding and exploitation of system vulnerabilities, as the DARPA Cyber Grand Challenge showed in 2016.

However, as mentioned above, we'll be focusing on the cybersecurity threats that are unique to deep learning, which means they couldn't have existed before deep learning algorithms found their way into our software.

We also won't be covering algorithmic bias and other societal and political implications of neural networks such as persuasive computing and election manipulation. Those are real concerns, but they require a separate discussion.

To examine the unique security threats of deep learning algorithms, we must first understand the unique characteristics of neural networks.

What makes deep learning algorithms unique?



Deep learning is a subset of machine learning, a field of artificial intelligence in which software creates its own logic by examining and comparing large sets of data. Machine learning has existed for a long time, but deep learning only became popular in the past few years.

Artificial neural networks, the underlying structure of deep learning algorithms, roughly mimic the physical structure of the human brain. As opposed to classical software development approaches, in which programmers meticulously code the rules that define the behavior of an applications, neural networks create their own behavioral rules through examples.

When you provide a neural network with training examples, it runs it through layers of artificial neurons, which then adjust their inner parameters to be able to classify future data with similar properties. This is an approach that is very useful in for use cases where manually coding software rules is very difficult.

For instance, if you train a neural network with sample images of cats and dogs, it'll be able to tell you if a new image contains a cat or a dog. Performing such a task with classic machine learning or older AI techniques was very difficult, slow and error-prone. Computer vision, speech recognition, speech-to-text and facial recognition are some of the areas that have seen tremendous advances thanks to deep learning.

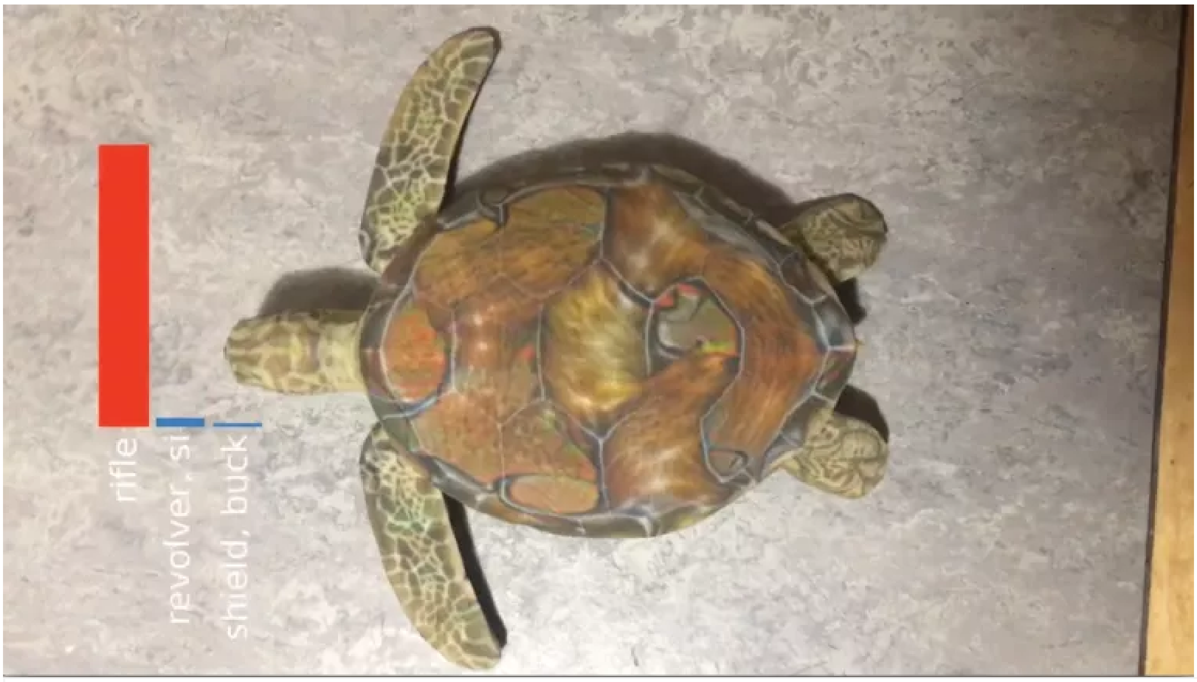
But what you gain in terms of accuracy with neural networks, you lose in transparency and control. Neural networks can perform specific tasks very well, but it's hard to make sense of the billions of neurons and parameters that go into the decisions that the networks make. This is broadly called the "AI black box" problem. In many cases, even the people who create deep learning algorithms have a hard time explaining their inner workings.

To sum things up deep learning algorithms and neural networks have two characteristics that are relevant from a cybersecurity perspective:

- They are overly reliant on data, which means they are as good (or bad) as the data they are trained with.
- They are opaque, which means we don't know how they function (or fail).

Next, we'll see how malicious actors can use the unique characteristics of deep learning algorithms to stage cyberattacks.

Adversarial attacks



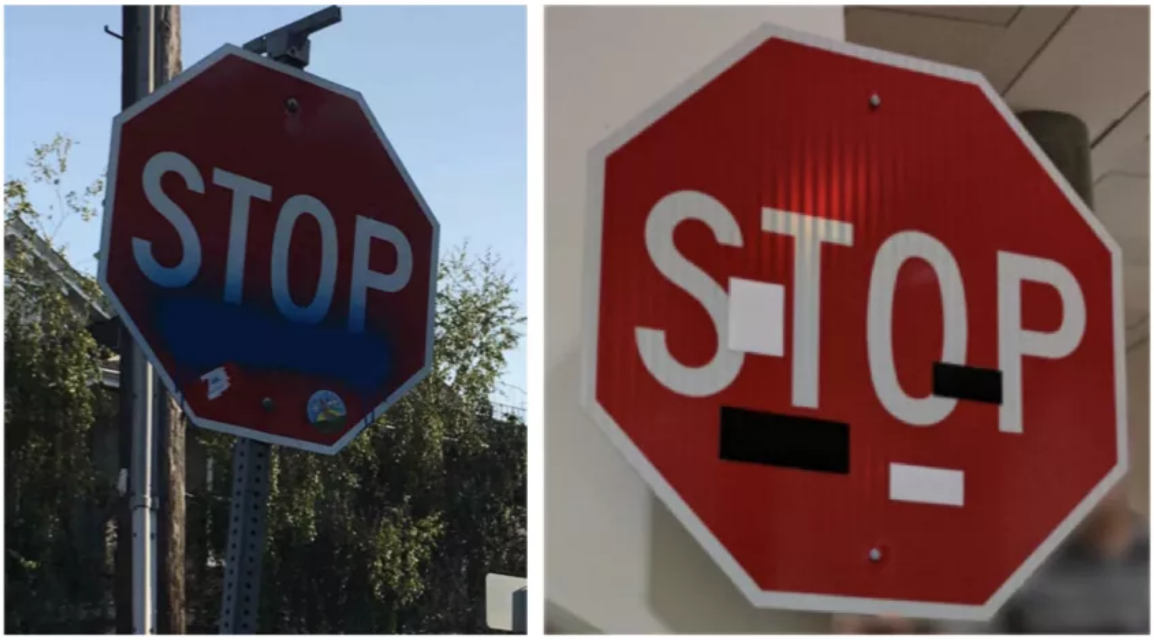
Researchers at labsix showed how a modified toy turtle could fool deep learning algorithms into classifying it as a rifle (source: labsix.org)

Neural networks often make mistake that might seem totally illogical and stupid to humans. For instance, last year, an AI software used by the UK Metropolitan Police to detect and flag pictures of child abuse wrongly labeled pictures of dunes as nudes. In another case, students at MIT showed that making slight changes to a toy turtle would cause a neural network to classify it as a rifle.

These kinds of mistakes happen all the time with neural networks. While neural networks often output results that are very similar to what a human would produce, they do not necessarily go through the same decision-making process. For instance, if you train a neural network with images of white cats and black dogs only, it might optimize its parameters to classify animals based on their color rather than their physical traits such as the presence of whiskers or a stretched muzzle.

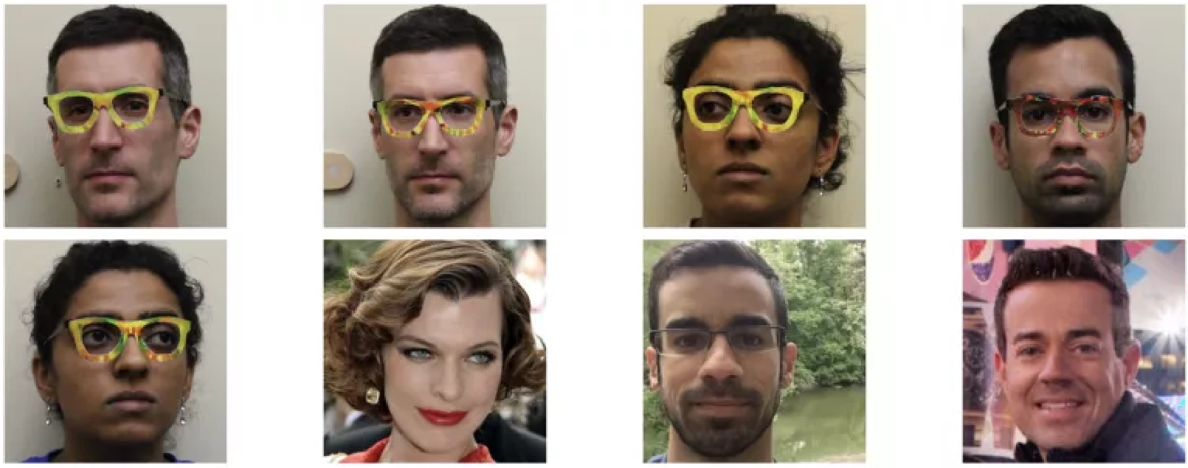
Adversarial examples, inputs that cause neural networks to make irrational mistakes, accentuate the differences between the functions of AI algorithms and the human mind. In most cases, adversarial examples can be fixed by providing more training data and allowing the neural network to readjust its inner parameters. But because of the opaque nature of neural networks, finding and fixing the adversarial examples of a deep learning algorithm can be very difficult.

Malicious actors can leverage these mistakes to stage adversarial attacks against systems that rely on deep learning algorithms. For instance, in 2017, researchers at Samsung and Universities of Washington, Michigan and UC Berkley showed that by making small tweaks to stop signs, they could make them invisible to the computer vision algorithms of self-driving cars. This means that a hacker can force a self-driving car to behave in dangerous ways and possibly cause an accident. As the examples below show, no human driver would fail to notice the "hacked" stop signs, but a neural network could perfectly become blind to it.



AI researchers discovered that by adding small black and white stickers to stop signs, they could make them invisible to computer vision algorithms (Source: arxiv.org)

In another example, researchers at Carnegie Mellon University showed that they could fool the neural networks behind facial recognition systems to mistake a subject for another person by wearing a special pair of glasses. This means that an attacker would be able to use the adversarial attack to bypass facial recognition authentication systems.



Researchers at Carnegie Mellon University discovered that by donning special glasses, they could fool facial recognition algorithms to mistake them for celebrities (Source: www.cs.cmu.edu)

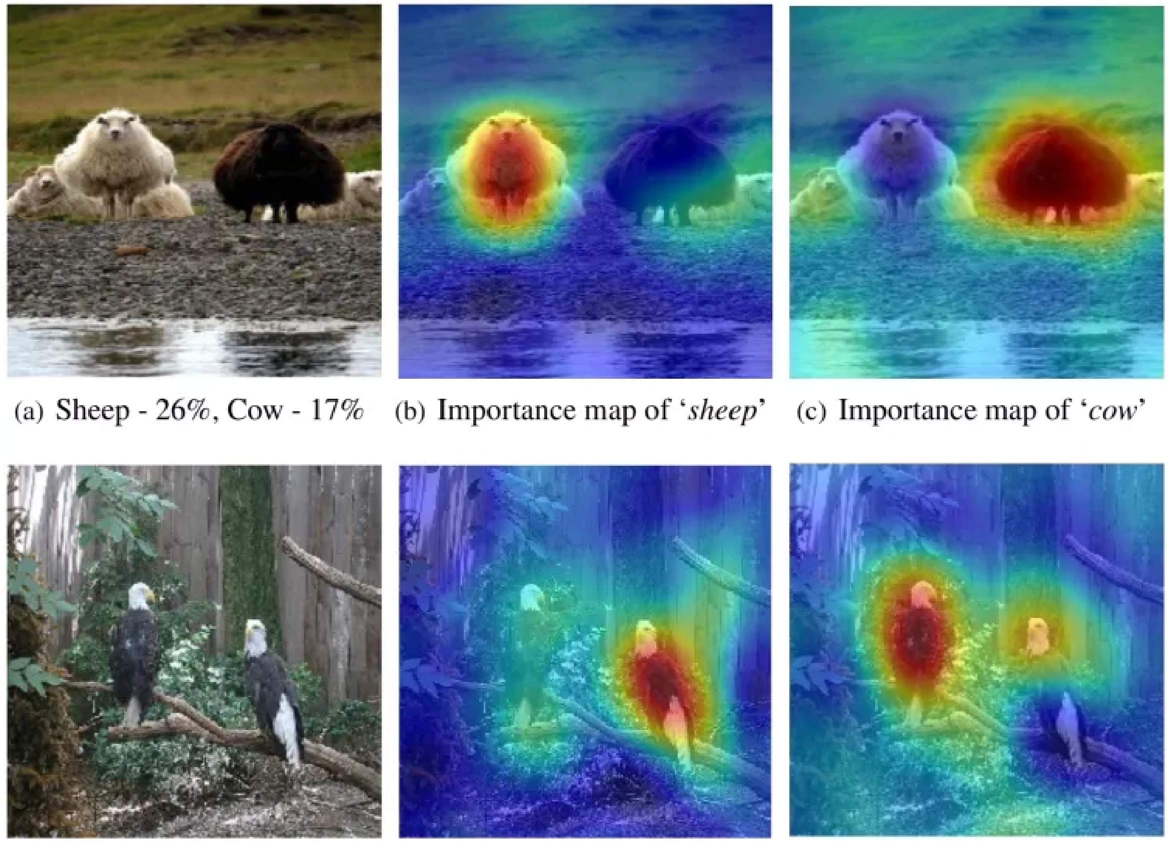
Adversarial attacks are not limited to computer vision. They can also be applied to voice recognition systems that rely on neural networks and deep learning. Researchers at UC Berkley developed a proof-of-concept in which they manipulated an audio file in a way that would go unnoticed to human ears but would cause an AI transcription system to produce a different output. For instance, this kind of adversarial attack can be used to change a music file in a way that would send commands to a smart speaker when played. The human playing the file would not notice the hidden commands that the file contains.

For the moment, adversarial attacks are only being explored in laboratories and research centers. There’s no evidence of real cases of adversarial attacks having taken place. Developing adversarial attacks is just as hard as finding and fixing them. Adversarial attacks are also very unstable, and they can only work in specific circumstances. For instance, a slight change in the viewing angle or lighting conditions can disrupt an adversarial attack against a computer vision system.

But they are nonetheless a real threat, and it's only a matter of time before adversarial attacks will become commoditized, as we've seen in other ill uses of deep learning.

But we're also seeing efforts in the artificial intelligence industry that can help mitigate the threat of adversarial attacks against deep learning algorithms. One of the methods that can help in this regard is the use of generative adversarial networks (GAN). GAN is a deep learning technique that pits two neural networks against each other to generate new data. The first network, the generator, creates input data. The second network, the classifier, evaluates the data created by the generator and determines whether it can pass as a certain category. If it doesn't pass the test, the generator modifies its data and submits it to the classifier again. The two neural networks repeat the process until the generator can fool the classifier into thinking the data it has created is genuine. GANs can help automate the process of finding and patching adversarial examples.

Another trend that can help harden neural networks against adversarial attacks are the efforts in creating explainable artificial intelligence. Explainable AI techniques help reveal the decision processes of neural networks and can help investigate and discover possible vulnerabilities to adversarial attacks. An example is RISE, an explainable AI technique developed by researchers at Boston University. RISE produces heat maps that represent which parts of an input contribute to the outputs produced by a neural network. Techniques such as RISE can help find potentially problematic parameters in neural networks that might make them vulnerable to adversarial attacks.



(a) Sheep - 26%, Cow - 17% (b) Importance map of '*sheep*' (c) Importance map of '*cow*'

(d) Bird - 100%, Person - 39% (e) Importance map of '*bird*' (f) Importance map of '*person*'

Examples of saliency maps produced by RISE

Data poisoning

While adversarial attacks find and abuse problems in neural networks, data poisoning creates problematic behavior in deep learning algorithms by exploiting their over-reliance on data. Deep learning algorithms have no notion of moral, commonsense and the discrimination that the human mind has. They only reflect the hidden biases and tendencies of the data they train on. In 2016, Twitter users fed an AI chatbot deployed by Microsoft with hate speech and racist rhetoric, and in the span of 24 hours, the chatbot turned into a Nazi supporter and Holocaust denier, spewing hateful comments without hesitation.

Because deep learning algorithms are only as good as their data, a malicious actor that feeds a neural network with carefully tailored training data can cause it to manifest harmful behavior. This kind of data poisoning attack is especially effective against deep learning algorithms that draw their training from data that is either publicly available or generated by outside actors.

There are already several examples of how automated systems in criminal justice, facial recognition and recruitment have made mistakes because of biases or shortcomings in their training data. While most of these examples are unintentional mistakes that already exist in our public data due to other problems that plague our societies, there’s nothing preventing malicious actors from intentionally poisoning the data that trains a neural network.

For instance, consider a deep learning algorithm that monitors network traffic and classifies safe and malicious activities. This is a system that uses unsupervised learning. Contrary to computer vision applications that rely on human-labeled examples to train their networks, unsupervised machine learning systems peruse through unlabeled data to find common patterns without receiving specific instructions on what the data represents.

For instance, an AI cybersecurity system will use machine learning to establish baseline network activity patterns for each user. If a user suddenly starts downloading much more data than their normal baseline shows, the system will classify them as a potential malicious insider. A user with malicious intentions could fool the system by increasing their download habits in small increments to slowly “train” the neural network into thinking this is their normal behavior.

Other examples of data poisoning might include training facial recognition authentication systems to validate the identities of unauthorized people. Last year, after Apple introduced its new neural network–based Face ID authentication technology, many users started testing the extents of its capabilities. As Apple had already warned, in several cases, the technology failed to tell the difference between identical twins.

But one of the interesting failures was the case of two brothers who weren’t twins, didn’t look alike and were years apart in age. The brothers initially posted a video that showed how they could both unlock an iPhone X with Face ID. But later they posted an update in which they showed that they had actually tricked Face ID by training its neural network with both their faces. Again, this is a harmless example, but it’s easy to see how the same pattern can serve malicious purposes.

Because neural networks are not transparent and human developers don’t create their rules, it’s hard to investigate and find the harmful behavior that a user might have intentionally inflicted into the deep learning algorithm.

Deep learning–based malware

Earlier this year, researchers at IBM introduced a new breed of malware that used the characteristics of neural networks to hide its malicious payload and to target specific users. Targeted attacks were previously the domain of nation states and organizations with access to vast computing and intelligence resources.

But DeepLocker, the proof-of-concept malware developed by IBM, showed that such attacks might soon become the normal modus operandi of malicious hackers. DeepLocker had embedded its malicious behavior and payload into a neural network to hide it from endpoint security tools, which usually look for signatures and predefined patterns in the binary files of applications.

Another special feature of DeepLocker was the use of neural networks to designate specific targets for its payload. To display the destructive capabilities of the deep learning–based malware, the IBM researchers armed DeepLocker with a ransomware virus and embedded it in a video-conferencing application.

In an imaginable scenario, millions of users would install the malware-inflicted application on their computers and use it for their daily communications, oblivious to the presence of a dangerous ransomware on their device. Naturally, their antivirus software (if they have one) will also fail to detect and block the malicious application.

Meanwhile, the malware's developers have trained the neural network to activate the payload when it sees the face of a specific user through the web cam. Since the malware is embedded in a video conferencing application, it will have legitimate access to the web cam's video feed and will be able to monitor the users of the application. As soon as the target shows their face in front of the camera, DeepLocker unleashes the ransomware and starts to encrypt all the files on the user's computer.

Hackers will be able to use malware such as DeepLocker to target specific users or populations based on their gender and race, characteristics that deep learning algorithms have become very good at detecting.

We have yet to understand the scale of the cybersecurity threats of deep learning algorithms and neural networks. The researchers who created DeepLocker said they didn't know for certain whether such malware has already been let loose in the wild or not.
