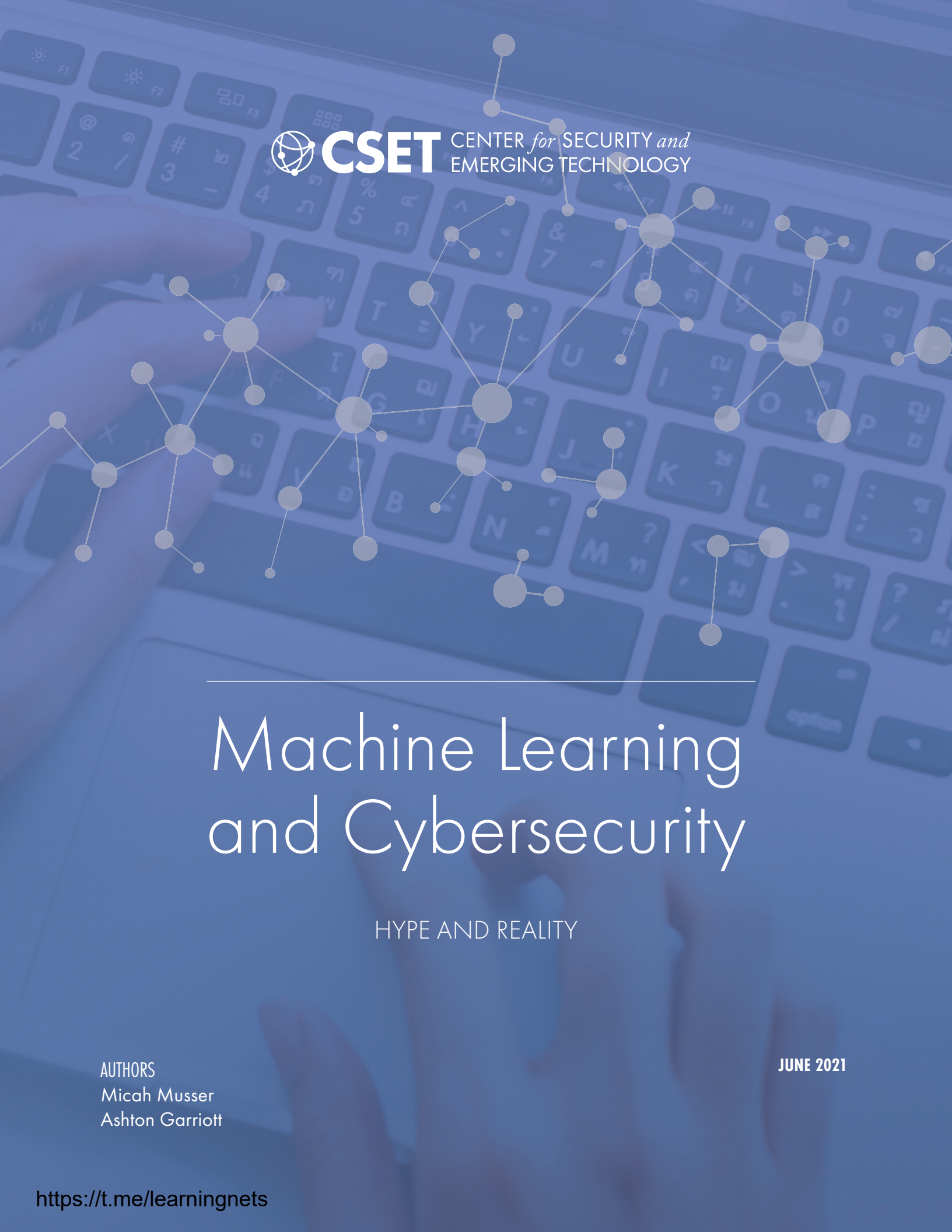


CSET CENTER *for* SECURITY *and* EMERGING TECHNOLOGY

Machine Learning and Cybersecurity

HYPE AND REALITY

AUTHORS
Micah Musser
Ashton Garriott

JUNE 2021

CENTER *for* SECURITY *and* EMERGING TECHNOLOGY

Established in January 2019, the Center for Security and Emerging Technology (CSET) at Georgetown's Walsh School of Foreign Service is a research organization focused on studying the security impacts of emerging technologies, supporting academic work in security and technology studies, and delivering nonpartisan analysis to the policy community. CSET aims to prepare a generation of policymakers, analysts, and diplomats to address the challenges and opportunities of emerging technologies. CSET focuses on the effects of progress in artificial intelligence, advanced computing, and biotechnology.

CSET.GEORGETOWN.EDU | CSET@GEORGETOWN.EDU

JUNE 2021

Machine Learning and Cybersecurity

HYPE AND REALITY

AUTHORS
Micah Musser
Ashton Garriott

ACKNOWLEDGMENTS

For their comments, discussion, and review, we would like to thank John Bansemer, Drew Lohn, Ben Buchanan, Cindy Martinez, Perri Adams, Jim Richberg, Chris Rohlf, Jamie Butler, and the reviewers from the MITRE Corporation: Mike Hadjimichael, George Roelke, Scott Musman, Roshan Thomas, Sam Visner, Malachi Jones, Andy Applebaum, Andres Molina-Markham, Kristin Esbeck, Lashon Booker, Anne Townsend, and Frank Posluszny. For their assistance editing and formatting this report, we would like to thank Alex Friedland and Melissa Deng. Any errors that remain are the fault of the authors alone.

AUTHORS

Micah Musser is a Research Analyst for the CyberAI team at CSET, where Ashton Garriott is a former Semester Research Analyst.

DOCUMENT IDENTIFIER

doi: 10.51593/2020CA004

Cover photo: https://www.flickr.com/photos/byrawpixel/39726353110.

Contents

Executive Summary

The size and scale of cyber attacks has increased in recent years as a result of a number of factors, including the increasing normalcy of cyber operations within international politics, the growing reliance of industry on digital infrastructure, and the difficulties of maintaining an adequate cybersecurity workforce. Many commentators across government, media, academia, and industry have wondered how cybersecurity professionals might be able to adapt machine learning for defensive purposes. Could machine learning allow defenders to detect and intercept attacks at much higher rates than is currently possible? Could machine learning–powered agents automatically hunt for vulnerabilities or engage an adversary during an unfolding attack? Should policymakers view machine learning as a transformational force for cyber defense or as mere hype?

This report examines the academic literature on a wide range of applications combining cybersecurity and artificial intelligence (AI) to provide a grounded assessment of their potential. It breaks cybersecurity practice into a four-stage model and examines the impact that recent machine learning innovations could have at each stage, contrasting these applications with the status quo. The report offers four conclusions:

- Machine learning can help defenders more accurately detect and triage potential attacks. However, in many cases these technologies are elaborations on long-standing methods—not fundamentally new approaches—that bring new attack surfaces of their own.

- A wide range of specific tasks could be fully or partially automated with the use of machine learning, including some forms of vulnerability discovery, deception, and attack disruption. But many of the most transformative of these possibilities still require significant machine learning breakthroughs.
- Overall, we anticipate that machine learning will provide incremental advances to cyber defenders, but it is unlikely to fundamentally transform the industry barring additional breakthroughs. Some of the most transformative impacts may come from making previously un- or under-utilized defensive strategies available to more organizations.
- Although machine learning will be neither predominantly offense-biased nor defense-biased, it may subtly alter the threat landscape by making certain types of strategies more appealing to attackers or defenders.

This paper proceeds in four parts. First, it introduces the scope of the research and lays out a simplified, four-stage schema of cybersecurity practice to frame the different ways that future machine learning tools could be deployed. Second, it contextualizes recent machine learning breakthroughs and their implications for cybersecurity by examining the decades-long history of machine learning as applied to a number of core detection tasks. The third and most substantial section examines more recent machine learning developments and how they might benefit cyber defenders at each stage of our cybersecurity schema, and considers whether these newer machine learning approaches are superior to the status quo. Finally, a concluding section elaborates on the four conclusions mentioned above and discusses why the benefits of machine learning may not be as transformative in the immediate future as some hope, yet are still too important to ignore.

Introduction

As a typical internet user goes about her day, she will be quietly protected by a bewildering number of security features on her devices—some obvious, others less so. If she uses Gmail, Google will automatically scan every email that arrives in her inbox to determine if it is spam, and if the email contains attachments, those will also be scanned to determine if they contain malware.[1] Whether she uses Chrome, Firefox, or Safari to browse the web, her browser will analyze every website she visits and attempt to alert her if a site is malicious.[2] If she uses an antivirus product—among the most common of which are Microsoft Windows Defender, Symantec, and ESET—then the files on her device will be regularly scanned to check for potentially malicious files.[3]

All of these services utilize machine learning to protect users from cyber attacks. Over the past decade, the role of machine learning in cybersecurity has been gradually growing as the threats to organizations become more serious and as the technology becomes more capable. The increasing commonality of cyber operations as a geopolitical tool means that many organizations risk being targeted by well-resourced threat actors and advanced persistent threats.[4] At the same time, the supply of trained cybersecurity professionals struggles to meet the growing need for expertise.[5] And to add to the sense of risk, a nearly endless stream of publications—including a recent report released by CSET—speculates about the ways AI will be used to further intensify cyber attacks in the near future.[6]

In this climate, researchers, policymakers, and practitioners alike have found themselves wondering if machine learning might provide the means to turn the tide in the war against cyber attacks. Popular media outlets frequently write about the possibility that machine learning could vastly

If machine learning has slowly been making its way into the cybersecurity industry for decades, yet the rate and scale of cyber attacks hasn't meaningfully gone down, why should anyone believe that a transformation is just around the corner?

improve attack detection, while many in the defense community talk about future AI systems that could hunt for and dispatch intruders autonomously.[7] "AI" has become a mainstay in cybersecurity marketing materials, with advertisements comparing AI products to everything from the immune system to swarms of organisms.[8] Taking these claims at face value, a policymaker paying loose attention to the state of AI research might conclude that the total transformation of the cybersecurity industry at the hands of machine learning algorithms is at most a few years away.

So it might surprise that same policymaker to learn that machine learning systems have been commonly used for a number of key cybersecurity tasks for nearly 20 years. While the breakthroughs of the last five years have rightfully drawn attention to the field of AI and machine learning (ML) research, it is easy to forget that the algorithms behind these advances have in many cases been around for decades.[9] Learning this, that same policymaker might look at the spate of recent high-profile hacks and feel tempted to dismiss the potential of machine learning altogether. After all, if machine learning has slowly been making its way into the cybersecurity industry for decades, yet the rate and scale of cyber attacks hasn't meaningfully gone down, why should anyone believe that a transformation is just around the corner?

Part of the difficulty in evaluating machine learning's potential for cybersecurity comes from the fact that different parties have different standards for success. For our purposes, it is useful to make a distinction between two types of standards that we might use to evaluate the impact of a new technology: counterfactual standards and historical standards.

Using counterfactual standards means that we ask the question: Where would we be without machine learning, all else being equal? Approaching the question from this angle should give us a great deal of appreciation for machine learning: given that the number of events that signal a possible cyber attack has grown from several hundred to several million per day in some industries, it's surprising that most

That machine learning can offer significant benefits to cybersecurity practitioners is broadly—though not universally—agreed upon. Whether these benefits will amount to a "transformation" in cybersecurity is more contested.

companies aren't entirely consumed by the task of keeping their IT infrastructure secure.[10] In no small part, this success is thanks to ML systems that can automatically screen out potential attacks, generate alerts on suspicious behavior, or perform some rudimentary analysis of anomalous activity.

Counterfactual standards are what matter to the cybersecurity practitioner—the person who knows the threat landscape and has to respond to it one way or another. But policymakers also need insight into a set of more general questions: Is the world getting more secure or less? How worried should we be over the long-term about the shortage of cybersecurity talent? Can machine learning make us less vulnerable to our adversaries in absolute terms? To answer these questions, we need to use historical standards. Rather than comparing the world with machine learning to a hypothetical world without it, we should compare the past to the present to the (most likely) future.

That machine learning can offer significant benefits to cybersecurity practitioners is broadly—though not universally—agreed upon.[11] Whether these benefits will amount to a "transformation" in cybersecurity is more contested. For the purposes of this paper, "transformative impact" refers to an impact that makes a difference by historical—and not merely counterfactual—standards. In the context of cybersecurity, this means that a technology should do more than help defenders simply *keep up* in the face of growing threats: to be transformative, the technology should bring the total number of threats requiring a human response down (and keep them down), or it should meaningfully alter cybersecurity practice.

Ultimately, it is more important for policymakers to understand *how* machine learning will transform cybersecurity than it is to quibble about whether it will bring changes. For that reason, this paper explores the potential impact that a wide range of ML advances could have on cybersecurity. We try to simultaneously present the reader with a healthy dose of skepticism regarding some of the most-hyped ML

applications in cybersecurity, while also drawing attention to some potential applications that have garnered less popular attention. And, in keeping with our emphasis on historical standards for success, we try to contextualize the recent growth of interest in machine learning for cybersecurity by examining how it has already been deployed in the past.

This is a large task, and unfortunately not everything relevant to machine learning's growing role in cybersecurity can be covered in this report. Our focus is squarely on the technical—what machine learning *could* do. We sidestep many important but non-technical issues, such as privacy or legal concerns surrounding cybersecurity data collection, as well as the economics of implementing newer ML methods. We also adopt a general model of cybersecurity and avoid organizations that may have unique cybersecurity needs or goals, such as companies that rely on industrial control systems.

This report is structured as follows: We begin in the first section by introducing a four-stage model of cybersecurity practice that we will return to in later sections as a way to schematize different ML applications. The second section, "Traditional Machine Learning and Cybersecurity," discusses how decades-old ML methods have been widely studied for three core cybersecurity tasks: spam filtering, intrusion detection, and malware detection. Providing this historical context can help the reader understand which ML applications are genuinely new innovations, and which ones are merely extensions of long-standing trends in cybersecurity. In the third section, "Cybersecurity and the Cutting Edge," we turn our attention to newer advances in ML research and examine how these methods might generate new types of applications at each part of our four-stage model of cybersecurity. This section emphasizes lines of academic research that seem promising and may translate into fully functional products within the near- to medium-term. The final section presents conclusions that appear to follow from this research, along with some additional analysis.

1 A Four-Stage Model of Cybersecurity

The most well-known schema for conceptualizing cybersecurity is the Cybersecurity Framework designed by the National Institute of Standards and Technology (NIST).[12] This framework broadly breaks cybersecurity into five functions to help defenders assess their level of risk: identify, protect, detect, respond, and recover. This paper uses an adapted version of the NIST model that emphasizes four slightly different categories: prevention, detection, response and recovery, and active defense. There are two primary reasons to deviate from the NIST framework. First, while there are many instances of machine learning being used to detect cyber attacks, the use of ML in the other four categories of the NIST framework is still rather immature, which justifies grouping multiple categories together for analysis. Second, "active defense"—which we elaborate on below—is a growing area of interest that is conceptually distinct from the other categories for both technical and operational reasons; it therefore merits its own discussion.

TABLE 1

A Comparison of Our Model and the NIST Cybersecurity Framework

NIST FUNCTIONS	OUR MODEL
Identify	Prevention
Protect	
Detect	Detection
Respond	Response and Recovery
Recover	
N/A	Active Defense

In this model, *prevention* refers to all the actions that security experts take to minimize the vulnerabilities of their organization. At a minimum, prevention requires that defenders make secure decisions regarding network configurations and user privileges, and that they maintain an active knowledge of their networks and software dependencies, patching known vulnerabilities in a timely fashion. For software companies, prevention includes the work of evaluating one's own products for vulnerabilities, preferably before a product is brought to market. For mid- or large-sized organizations, prevention often requires frequent audits of one's overall security posture as well as periodic penetration testing, in which benign attacks are launched against a network to discover and expose weaknesses. This category roughly corresponds to the NIST categories of identification, which emphasizes asset management and security strategy, and protection, which focuses on the technological and policy controls that are used to secure assets.

No amount of anticipatory work can make an organization immune from cyber attacks. When attacks do occur, *detection* systems are needed to quickly alert defenders so that they can respond. A comprehensive set of detection systems must monitor everything from network traffic to email attachments and employee behavior in order to identify anomalies and potential threats. Because attackers often evade discovery for months after breaching a network, effective detection systems can make an enormous difference to an organization's ability to limit the impact of cyber attacks.[13]

Once an attack is detected, security professionals must determine how to *respond and recover*. For some types of threats, the response is straightforward: in the case of spam detection, for instance, an email service simply redirects mail that is likely illegitimate into a spam folder. For many other types of attacks, however, the appropriate response is far from clear. In the face of an ongoing attack, cybersecurity personnel must often decide whether to shut down machines, sequester parts of their network, or otherwise take steps that could significantly disrupt an organization's operations. An effective response must identify the scale and scope of an attack, thwart the attacker's access, and fully eliminate any foothold the attacker might have. After doing so, it is important to restore the system to its original state prior to the attack.

The triad of prevention, detection, and response and recovery forms the core of cybersecurity. For most organizations, performing these tasks well is the height of good cybersecurity practice. However, for organizations facing attacks from well-resourced threat actors, compliance with pre-existing frameworks may not be sufficient. These organizations must also ensure that they can flexibly adapt to changes in the threat landscape.

To account for actions that allow organizations to respond more flexibly to sophisticated threats, this report includes the additional stage of *active defense*. This term is used analogously to the way the SANS Institute has used it: as a spectrum of activity that includes annoyance, attribution, or outright counter-attack.[14] Active defense can be thought of as an "other" category that includes any attempt to deliberately engage or study external actors rather than simply responding to problems as they arise. This category can be broken down into a few more clearly defined subcategories, of which this report emphasizes three: (1) deception, or attempts to mislead and slow down adversaries; (2) threat intelligence, or attempts to actively study potential adversaries to better anticipate their actions; and (3) attribution, or attempts to connect multiple events to a single entity that can then be studied in more detail.* Active defense, done well, can allow defenders to stay ahead of their adversaries and can potentially create disincentives against attacking in the first place.

Section 3 will examine each of these four components of cybersecurity and ask how newer ML architectures may play a role in augmenting current practices. But before turning towards the future of machine learning in cybersecurity, it is important

*This breakdown does not include any activities that would fall into the category of outright counter-attack. Because this is a much more difficult subject with an often unclear status of legality, we do not explore the possibility of using machine learning for counter-attack in this paper.

to get some sense of the past, because ML–based cyber tools are not as new as many assume. In the next section, this report examines three ML applications that have been studied for several decades: spam detection, intrusion detection, and malware detection.

2 Traditional Machine Learning and Cybersecurity

Although the last decade has seen major advances in AI and ML research, the use of ML methods for cybersecurity has a long history. Until recently, these applications fell almost entirely into the detection stage of cyber defense, with the most significant attention paid to spam detection, intrusion detection, and malware detection. This section discusses how simpler, longstanding ML approaches have historically been adapted for these three applications. Figure 1 presents a rough timeline of major developments in the cyber threat landscape and how machine learning has evolved to combat them.

"Traditional machine learning" here refers to the broad set of decades-old algorithms that were dominant in ML research until the major, predominantly deep learning–driven advances of the past 5–10 years.* These methods are typically divided into one of two categories: supervised learning and unsupervised learning. In supervised learning, labeled datasets are used to train a model to classify new inputs, while in unsupervised learning, unlabeled datasets are examined to identify underlying patterns within the data. Either approach can be useful for cyber defenders looking

* Some examples of traditional machine learning algorithms include naive Bayes classifiers, random forests, k-means clustering, logistic regression models, and support vector machines. It is worth noting that despite the name we use, "traditional machine learning" is not itself a stagnant field; XGBoost, for instance, is a popular and newer gradient boosting algorithm, but one that simply represents a better optimization of older, more traditional approaches.

FIGURE 1

A Timeline of Machine Learning Developments for Three Major Cybersecurity Tasks*

	Pre-1990s	1990s	2000s	2010s
SPAM DETECTION	**1978:** First spam email	Spam continues to worsen due to growth in email **1996:** First spam blockers	**2002:** Machine learning methods first proposed for spam detection **2003:** First attempts to regulate spam in the United States	Machine learning spam detection widely embedded in email services Emergence of deep learning-based classifiers
INTRUSION DETECTION	**1980:** First intrusion detection systems **1986:** Anomaly detection systems combine expert rules and statistical analysis	**Early 1990s:** Neural networks for anomaly detection first proposed **1999:** DARPA creates datasets to study intrusion detection systems	Machine learning further studied as a possible tool for misuse-based and anomaly-based intrusion detection	**Late 2010s:** Emergence of large-scale, cloud-based intrusion detection systems Deep learning studied for intrusion detection
MALWARE DETECTION	**Early 1980s:** First viruses found "in the wild" **Late 1980s:** First antivirus companies founded	**Early 1990s:** First polymorphic viruses **1996:** IBM begins studying machine learning for malware detection	**Early 2000s:** First metamorphic viruses Wide number of traditional machine learning methods studied to detect malware	Rise of "next-gen" antivirus detection Emergence of ML-focused antivirus companies

to detect potentially malicious traffic. If well-labeled data on previous attacks exists, supervised methods can be used to detect future attacks by matching malicious traffic to a known profile, while unsupervised methods can be used to identify attacks merely because they are anomalous and out of place.†

To show the history of how these traditional machine learning approaches have been used in cybersecurity, we turn first to the example of spam detection.

*See the note on page 10 below for a definition of polymorphic and metamorphic viruses.

†This distinction is not as clear-cut as it seems; in some cases, unsupervised learning can be used to match attacks to known profiles and supervised learning can be used to detect anomalies. For the most part, however, the way we have made the distinction corresponds to the way that supervised and unsupervised algorithms have been studied for detection purposes.

SPAM DETECTION

Machine learning has been a major part of spam detection since the very early 2000s, and many of these early ML methods are still used today. Before the introduction of ML techniques, spam detection relied on blocklists that screened out mail from (known) malicious IP addresses or on keyword detection that blocked emails containing hand-curated lists of spammy terms like "free" or "sexy." Unfortunately, because these methods were applied indiscriminately, they could often block legitimate emails. To address this problem, computer scientists began to propose machine learning–based solutions around the turn of the century.[15]

These early methods were relatively straightforward: First collect a large body of emails, label them as either spam or legitimate, and split them into individual words. For each word, calculate the probability that an email was spam if it contained that word. When a new email arrives, the probabilities associated with each of its words could be used to calculate the risk that the email was spam, and emails with risk scores above a preset threshold could be automatically blocked.

The core elements of spam detection have not changed much since the early 2000s, though researchers have made improvements. Better spam classifiers can be built by extracting more technical details from mail headers, such as IP addresses and server information, or by treating words that appear in a subject line differently from words that appear in the body of an email.[16] Better algorithms can be used that recognize phrases or synonyms rather than treating all words as independent of one another.[17] Some companies have developed extremely complex spam detectors that can, among other things, track a user's past email interactions to flag anomalous contacts or use deep learning models to determine whether or not branded emails are sent from authentic companies.[18]

Nonetheless, even the most advanced spam classifiers used by companies like Google have mostly developed out of a slow process of elaboration and evolution from these earlier methods. Of course, even moderate improvements in accuracy can matter a great deal to massive companies responsible for protecting billions of emails. But it would be an error to portray recent innovations in ML spam detection systems as a fundamental transformation of past practice: in reality, machine learning has been central to the task for nearly two decades.

INTRUSION DETECTION

Intrusion detection systems attempt to discover the presence of unauthorized activities on computer networks, typically by focusing on behavior profiles and searching for signs of malicious activity. Intrusion detection systems are typically classified as either misuse-based or anomaly-based. In misuse-based detection,

attacks are identified based on their resemblance to previously seen attacks, whereas in anomaly-based detection, a baseline of "normal" behavior is constructed and anything that does not match that baseline is flagged as a potential attack. Both methods can make use of different ML methods.

The simplest forms of misuse-based detection rely on known indicators of compromise to detect previously encountered threats. For instance, if an organization has seen malware that attempts to contact a specific website, cyber defenders could write a simple detection system which provides an alert any time a machine on the network attempts to contact that website. Misuse-based detection—especially when based on simple methods like these—typically has high processing speeds and low false positive rates, which allows it to quickly and accurately identify malicious events. However, this form of detection can only monitor for known threats, providing little meaningful protection against novel attacks.

Machine learning can be used to automate some forms of misuse-based detection by allowing a system to "learn" what different types of attacks look like. If many (labeled) examples of past attacks are available, a supervised learning classifier can be trained to identify the tell-tale signs of different types of attacks, without the need for humans to generate specific lists of rules that would trigger an alert.* Since at least 1999, researchers have attempted to generate network traffic profiles of different types of attacks so that ML classifiers can learn how to identify previously seen attacks.[19] This research was initially pushed by DARPA, and the results suggested that machine learning could prove competent at misuse-based detection.[20]

Although it is tempting to think that newer ML methods—like the rise of deep learning—have enabled dramatically more powerful detection tools, one review of several dozen experimental results in 2018 suggested that deep learning is not reliably more accurate at misuse-based detection than decades-old ML approaches.[21] Because no one type of model is consistently best for misuse-based detection, researchers often find that the most successful machine learning systems are ensemble models, or models that classify new inputs by utilizing multiple classifiers that "vote" on a classification.[22] The relatively common use of this method avoids overreliance on any specific model—which may have its own blind spots—but it also indicates that no single architecture is clearly superior to the rest.

*Some researchers tend to define misuse-based detection more restrictively as being strictly based on curated lists of indicators of compromise rather than probabilistic models. Under this definition, ML cannot be used to perform misuse-based detection, since ML is by its very nature probabilistic. However, many ML classifiers resemble misuse-based detection systems in that they use labeled examples of previous attacks to detect future attacks, struggle to generalize to zero-day attacks, and do not seek to identify a baseline model of "normal" behavior. We therefore think it makes the most sense to categorize ML classifiers with these characteristics under the label of "misuse-based detection."

In contrast to misuse-based detection, anomaly-based detection flags suspicious behavior without making specific comparisons to past attacks. This type of detection system is more likely to use unsupervised learning methods to cluster "normal" traffic within a network and alert as suspicious any activity which deviates from that pattern. In theory, anomaly-based detection can identify novel attacks—one of the most difficult aspects of cybersecurity. To enable this type of capability, research in this area has focused on finding ways to appropriately baseline "normal" traffic for a given network, since even normal traffic can be highly variable, and a poorly tuned intrusion detection system will generate many false positives that are expensive to investigate.[23]

Unfortunately, although anomaly-based detection can be highly effective when tracking an individual machine or user, it often struggles to effectively identify suspicious behavior across a network. In addition, a long-standing complaint regarding anomaly detection systems has been their tendency to generate many false positives, which speaks to the difficulty of defining "normal" traffic strictly enough to detect any anomalies but loosely enough that no legitimate behavior is flagged as anomalous.[24] Moreover, changes in an organization's standard procedures can dramatically undermine the usefulness of anomaly detection, at least until new baselines are learned—a lesson that many businesses discovered last year when millions of employees suddenly began working from home in response to COVID-19.[25] Because of the difficulties associated with anomaly detection, many organizations use it only as a complement to more standard misuse-based detection systems.

This discussion illustrates that machine learning already has a long and multifaceted history in the field of intrusion detection. Different ML methods have been adapted to multiple types of intrusion detection, in a research process dating back over two decades. Moreover, empirical studies and the continued importance of ensemble models speak to the fact that newer innovations have not fully displaced these older models. As with spam detection, it would be a mistake to think that the developments of the past decade of ML research, specifically the rise of deep learning, have entirely transformed intrusion detection—though in Section 3, we will return to this topic to discuss some of the ways in which intrusion detection *has* meaningfully been changed by newer ML innovations.

MALWARE DETECTION

While intrusion detection systems monitor a system or network's *behavior* to identify signs that a network is under attack, malware detection systems examine specific *files* to determine if they are malicious. The simplest forms of malware detection found in early antivirus products would monitor machines for specific indicators of compromise, such as exact filenames or signatures (which are specific sequences

of bytes or strings taken from the contents of a file). By maintaining long lists of malware signatures and conducting regular scans, these antivirus products tried to determine if any files on a machine were associated with these known definitions.

Unfortunately, these detection methods can be easily evaded by polymorphic or metamorphic viruses—types of malware that change their own code each time they propagate—thereby ensuring that different versions will have different signatures.* By some estimates, in 2018 up to 94 percent of malicious executables exhibited polymorphic traits.[26] While traditional detection techniques can still be adapted for the detection of polymorphic or metamorphic malware—for instance, by looking at the sequence of actions the malware will take when executed rather than by matching based on raw code—these traditional methods become increasingly complex and computationally intensive as attackers improve.[27]

Machine learning, however, excels at identifying shared features between samples that can't be classified using simple rules. As early as 1996, researchers at IBM began to explore the use of neural networks to classify boot sector viruses, a specific type of virus that targets a machine's instructions for booting up.[28] Additional research throughout the early 2000s explored the use of statistical models and standard ML classifiers for detecting malware.[29] Recent years have seen an explosion of interest in malware detection methods that utilize newer, deep learning–based approaches, methods which come with many advantages, such as the ability to extract relevant features from raw data with less human guidance.[30] Yet, as with intrusion detection, some experimental results indicate that—at least on some datasets—decades-old ML classifiers remain on par with more advanced deep learning methods when trained on the same data.[31]

Notably, many techniques that do not use machine learning have remained effective at detecting and analyzing malware, despite the rise of polymorphic and metamorphic viruses. For instance, cyber defenders may execute unrecognized files in sandboxes—isolated virtual environments where a file can be allowed to execute without any ability to interact with real systems and data.[32] The sandbox collects information about the file, such as what kinds of functions it tries to execute, to determine if it is malicious. This method allows an antivirus product to detect polymorphic or metamorphic code *without* relying on machine learning and underscores the fact that machine learning is by no means the only way for cyber defenders to respond

*Polymorphic code can refer to any type of code which appears in multiple forms, for instance, because attackers have created multiple variants or because the code can encrypt itself using a different key each time it propagates. By contrast, metamorphic code actually rewrites its underlying code while preserving the same functionality.

to the intelligent evolution of cyber attacks. Even sandboxes, however, can be augmented with machine learning to identify files that *resemble* past malware but that do not necessarily try to execute the exact same behavior.[33]

Throughout this section, we have emphasized two major points. First, there is a multi-decade-long history of researchers applying traditional ML techniques to major cybersecurity tasks, though with a very heavy emphasis on detection tasks. And second, though more powerful methods exist today, they typically represent natural evolutions from more traditional approaches. These facts are important to keep in mind when determining just how "transformative" recent ML advances will be for cybersecurity.

3 Cybersecurity and the Cutting Edge of AI Research

Although many decades-old ML approaches in cybersecurity remain competitive with more recent algorithms, the recent spike in machine learning interest is driven by some genuinely impressive breakthroughs. In recent years, AI innovations have led to self-driving cars, accurate language translation, and better-than-human video game playing. Although the possibility of transferring these successes into the cyber domain is uncertain, there is strong reason to think that—could this transfer be achieved—machine learning might become a major aid for cyber defenders.

This section explores how newer ML architectures might be applied to cybersecurity. It is organized around the four-stage model of cybersecurity introduced earlier and discusses potential ML applications for each stage in turn. As it proceeds, it pays particular attention to four types of ML methods that have been responsible for many of the AI breakthroughs of the past five years: 1) deep learning; 2) reinforcement learning; 3) generative adversarial networks (GANs); and 4) massive natural language models.* This section assumes a basic familiarity with these concepts; readers without a background in ML should refer to the appendix for a brief description of each type of model. Of these breakthroughs, the development of deep learning systems has in many ways been the most fundamental. It is the *combination* of deep learning architectures with other types of approaches—such as GANs, reinforcement learning systems, or natural

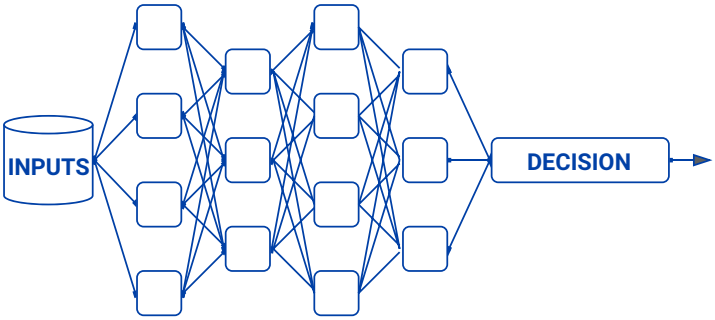
* Cutting-edge language models are impressive in their own right, but they have also drawn attention to the applications of simpler forms of natural language processing. Many of the natural language tools we discuss in this section are not particularly complicated, but they do represent a growing interest in the question of how linguistic data might be better leveraged for cyber defense.

language models—that has enabled most of the progress of the past half-decade, as visualized in Figure 2.

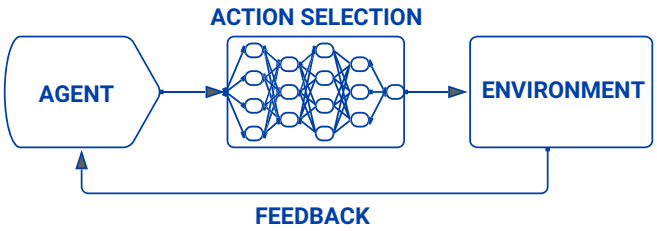
FIGURE 2

The Relationship Between Cutting-Edge AI Architectures

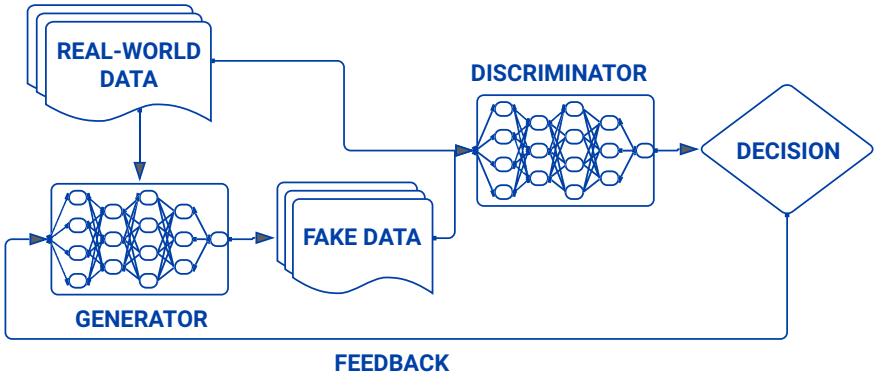
Deep learning is a general AI architecture modelled off of neural networks. It can be adapted for many tasks ...



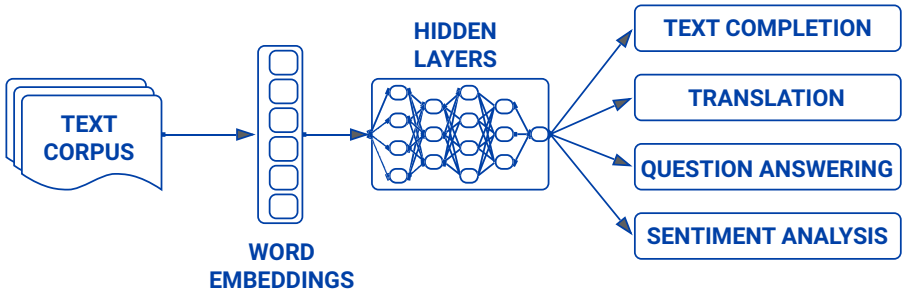
... from reinforcement learning systems where AI agents learn how to interact with their environments ...



... to GANs, where a generator learns how to produce outputs that can fool a discriminator ...



... to massive natural language models that can perform a wide range of language-related tasks.

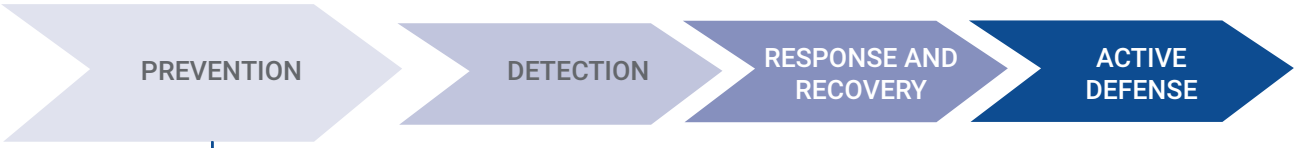


At the beginning of our discussion of each stage, we provide a graphic listing a few potential ML tools that could be leveraged by defenders at that stage. These graphics also list the type of ML architecture that implementations of each tool might rely on and a rough assessment of how significantly ML can transform the task itself. All of these assessments are meant to be very general and draw from the discussions that follow.

PREVENTION

FIGURE 3

AI Applications for Prevention*



CYBERSECURITY TASK	UNDERLYING TECHNOLOGY OF NEW AI APPLICATIONS	SIGNIFICANCE OF TASK	TRANSFORMATIVE POTENTIAL OF NEW AI
Fuzzing	Deep Learning	High	Medium-High
Pentesting	Reinforcement Learning	High	Medium-High
Bug Triage and Classification	NLP, Traditional ML Methods	Medium	Medium
Vulnerability Severity Assessment	NLP, Traditional ML Methods	Medium	Medium-Low

*NLP refers to natural language processing. This figure lists only the **most common** and **most specific** underlying technology for each application; a deep learning-based GAN, for instance, would be simply listed as "GAN."

The first stage of our cybersecurity model is prevention—the work that defenders do to find and patch vulnerabilities in order to eliminate potential weaknesses. There has long been interest in building tools that can autonomously find and patch new vulnerabilities, but machine learning has only recently emerged as a plausible way of doing so. As recently as the 2016 DARPA-sponsored Cyber Grand Challenge, the most promising routes for automated vulnerability discovery still relied on carefully coded systems without the use of ML methods. Though several teams at that competition attempted to use machine learning to identify software vulnerabilities, Mayhem—the winning program—ultimately made no use of ML in its vulnerability discovery modules.

Instead, Mayhem used two more common vulnerability discovery methods to identify potential weaknesses. First, a symbolic execution engine analyzed programs by building up representations of their execution logic, thereby learning how potentially crash-inducing behavior could be triggered. Second, it made use of fuzzers, which feed many modified and semi-random inputs to a program to determine if they cause any unexpected states that can be exploited.[34] In Mayhem, both of these elements were coded using traditional, rules-based systems, which were combined with modules that could autonomously write and deploy patches for discovered vulnerabilities.

The Cyber Grand Challenge resulted in several significant innovations in the field of automated vulnerability discovery. That no team at the time made heavy use of machine learning demonstrates that machine learning is far from the only way to build autonomous cyber tools.* At the same time, ML research has made significant strides since 2016, and experts disagree about whether participants in a new Cyber Grand Challenge would find more use for machine learning today.[35]

Consider fuzzers. In recent years, researchers studying fuzzers have increasingly explored deep learning as a means to more efficiently learn from successful inputs and find a larger number of vulnerabilities over time.[36] Deep learning–based fuzzers are often more efficient than older models; as one example, the deep learning–based program NeuFuzz was able to find three times as many crash-causing inputs as a modified version of a leading open source fuzzer across a variety of file types.[37] Outside of academia, Microsoft has also studied the use of deep learning to augment its own fuzzers.[38]

Fuzzers look for vulnerabilities in code, but cyber defenders can also use penetration testing (or pentesting for short) to look for publicly known vulnerabilities and

*Mayhem did use machine learning in other areas. For instance, the team found that machine learning could be used to generate realistic-seeming attack traffic, which could mislead other teams and cause them to waste resources. While intriguing, this application has less relevance outside of the 2016 competition, with its unique scoring system and capture-the-flag setup.

insecure configurations in networks. In a pentest, experienced hackers systematically probe a network for vulnerabilities to identify potential weaknesses. For large organizations, pentests can be expensive, potentially costing tens of thousands of dollars and consuming weeks of employee time. Automated tools such as the open-source program Metasploit can partially offset these costs, but these tools simply run through a list of pre-selected, known exploits to determine if any machines on a network are vulnerable to them; they are not capable of strategizing how they use resources.[39]

Recently, researchers have studied how reinforcement learning can allow cyber defenders to build AI agents that conduct pentests more strategically.[40] Some researchers have demonstrated that reinforcement learning–based agents can devise plausible strategies for a variety of capture-the-flag style simulations, and can do so reasonably quickly if they first watch a few examples of human-led pentesting.[41] These types of reinforcement learning–based approaches allow automated tools to search a network for vulnerabilities much faster than rules-based pentesting tools like Metasploit.[42]

These successes require caveats, especially due to the high computational cost of reinforcement learning in complex environments. Studies exploring the use of reinforcement learning for pentesting typically rely on small environments—often simulated networks of around ten machines—with a limited number of exploits provided to the program. As either the complexity of the environment or the number of actions available to the program increases, reinforcement learning can quickly become computationally prohibitive.[43] This problem is difficult but not fully intractable: in other contexts, researchers have developed models that can efficiently narrow down the number of options that a program must consider.[44] If researchers could develop computationally feasible methods of simulating complex networks and efficiently choosing among many options, reinforcement learning programs could become important aids for pentesters, just as Metasploit was adopted as a major tool in the pentester's toolkit in previous decades. But it is also worth emphasizing that of all the technologies we discuss in this paper, this is the one that is perhaps most obviously of interest to attackers as well—indeed, attackers have often been observed using pentesting tools developed for legitimate use to instead compromise their targets' networks.[45]

Beyond finding vulnerabilities with autonomous fuzzers and pentesters, machine learning may soon be able to provide tools that can help defenders allocate their time and attention to the most pressing vulnerabilities. Although there are many other ways that machine learning could be adapted for preventative purposes, we mention two other applications here: analysis of new bug reports and severity assessment for identified vulnerabilities.

Since at least 2005, researchers have explored ways in which machine learning can be used to automatically analyze bug reports. ML systems might, for example, direct bugs to the software engineer most able to address them, label malware characteristics based on human-generated reports, or predict the likelihood that a new bug could present a security vulnerability.[46] Though much of the research in this area has yielded results that are insufficiently accurate for operational use, Microsoft recently demonstrated that machine learning can successfully use bug report titles to classify bugs as security-relevant or non-security-relevant. The Microsoft model was trained on more than a million actual Microsoft bug reports, suggesting that some automated forms of bug report analysis may be feasible for software companies with access to large amounts of training data.[47]

Once vulnerabilities have been identified, defenders have limited time and resources to patch them. One prominent—though somewhat controversial—metric that helps defenders determine the severity of a vulnerability is the Common Vulnerability Scoring System (CVSS), which relies upon expert analysis to assign severity scores to new vulnerabilities. Some researchers think that machine learning and text mining could be used to automate the process of assigning CVSS scores by intelligently interpreting vulnerability descriptions.[48] Other researchers have used data about attacks observed in the wild to build machine learning systems that predict the likelihood of some vulnerabilities actually being exploited.[49] There is some evidence that when these machine learning–based risk assessments are used in conjunction with the CVSS, organizations can achieve similar levels of risk remediation for significantly less effort by prioritizing their remediation efforts on the vulnerabilities that are most likely to be exploited.[50] Taken together, the use of machine learning for improved fuzzing, pentesting, bug report analysis, and severity assessment could allow organizations to improve their ability to identify and prioritize vulnerabilities.

DETECTION

FIGURE 4
AI Applications for Detection

CYBERSECURITY TASK	UNDERLYING TECHNOLOGY OF NEW AI APPLICATIONS	SIGNIFICANCE OF TASK	TRANSFORMATIVE POTENTIAL OF NEW AI
Accurate Detection	Deep Learning	High	Low
Alert Prioritization	Deep Learning	Medium-High	Medium
Adversarial Hardening of Detection Systems	GANs	Medium-High	Medium-Low

Detection remains, at least among many popular-facing venues, the key place where deep learning and newer ML methods are thought to be potentially transformative forces.[51] Unfortunately, as of yet, deep learning has not provided the revolutionary breakthroughs that many have hoped for. While sufficiently large models do tend to perform incrementally better than simpler models—especially at large enough scales—these improvements are at times outweighed by the growing number of threats facing most organizations. The bottom line: despite the fundamental role that deep learning has played in the ML advances of the last half-decade, many cybersecurity companies continue to make significant use of simpler models today.

One problem with ML-based detection, sometimes overlooked in popular coverage, is that ML systems are vulnerable to several classes of attack that do not apply to other types of detection systems. The process by which many ML systems reach decisions can often be poorly understood and highly sensitive to small changes that a human analyst would view as trivial, which often makes it possible

for attackers to find "adversarial examples"—slightly altered inputs that dramatically change a model's response despite being undetectable to a human.[52] The use of ML models also opens up new avenues of attack: the model itself must be kept secure, but defenders must also make sure that their data is not poisoned and that the (typically open source) algorithms and statistical packages they use have not been tampered with.[53]

In addition, while machine learning is sometimes presented as an objective process of "learning patterns from data," in reality the design of ML systems is often the result of many judgment calls. In one somewhat infamous example, the security firm Cylance deployed an ML-based malware detection product, only for a group of white hat hackers to discover that if a short 5MB string were attached to any malicious file, the file would bypass the detection system nearly 90 percent of the time.[54] It seems that Cylance had built an ML malware detection system that worked relatively well, only to discover that it also blocked a number of legitimate games from being downloaded. In response, the firm added on a *second* ML system that calculated how similar a file was to any file on a whitelist of approved games, and which let high-scoring files through the system, even if the first ML model had identified them as potentially malicious. It was that second model that the white hat hackers were able to target: by adding on a string of code from the popular game *Rocket League*, they were able to create malware that Cylance's detection system failed to flag.

This anecdote is not necessarily a story about ML's flaws so much as it is a story about the difficulties of implementing ML systems in real-world environments. Cylance's core malware detection algorithm seems not to have been vulnerable to this "universal bypass"; rather, it was the designers' decision to add on a second component in order to avoid blocking popular video games that introduced the vulnerability. Ultimately, the use of machine learning can bring several types of vulnerabilities, whether because those weaknesses are inherent to the ML approach itself (as is the case with data poisoning threats and adversarial examples), or because wrangling the ML model into a deployable product leads developers to inadvertently add in further vulnerabilities.

None of this means that newer ML innovations are unimportant at the detection stage. For one, while older models can typically only provide rudimentary assessments of the severity of different types of alerts, deep learning can allow more sophisticated forms of analysis, which can help defenders prioritize more serious threats. This would provide an enormous benefit to cyber defenders, because many detection systems today generate so many false alerts that it is nearly impossible for human analysts to investigate them all. Some organizations are already making

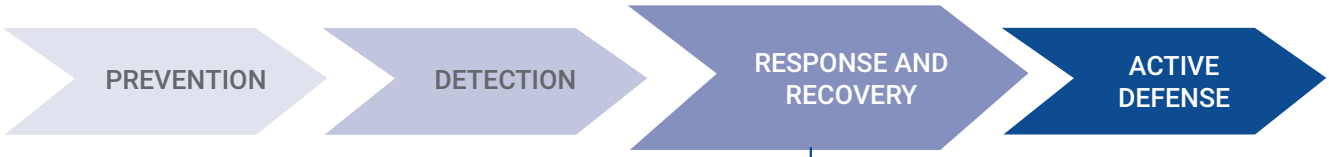
use of tiered systems, in which deep learning models analyze alerts from first-level detection systems to determine which potential threats require the highest priority.[55] These types of systems are most useful for large organizations, since they require not only a great deal of data about "normal" traffic, but also many examples of unique attacks.

In addition, while adversarial examples are specifically crafted to fool ML models, the rise of other types of ML systems—most notably GANs—that can generate synthetic and realistic-looking data poses a threat to both ML and rules-based forms of attack detection. Researchers have already demonstrated that GANs can develop attack strategies that slip by existing intrusion detection systems, whether or not they are machine learning–based.[56] But GANs can also help to harden ML systems against these types of more elaborate threats: when intrusion detection systems are trained using a GAN to anticipate potential adversarial attacks, their accuracy against adversarial attacks is greatly improved.[57] GANs can also augment existing datasets by generating new data that can be reincorporated into a classifier's training data. GAN-augmented datasets of spam and malware have been used by some research teams to create classifiers that perform better than those trained only on real-world data.[58]

Even though we have been somewhat pessimistic regarding the potential of deep learning to transform detection capabilities, the rise of adversarial attacks and the ever-increasing scale of threats means that defenders cannot afford to ignore advances in machine learning. Because GANs allow attackers to subvert both ML- and rules-based detection systems, doing nothing is not an adequate response, and the best defense will likely rely at least in part upon the use of ML methods. And yet, even here it is important to emphasize that some researchers do not believe that the ML field has made meaningful progress in making ML systems robust to adversarial examples.[59] While adversarial training remains critical for defensive systems, it may not be enough to effectively counter the rising threat of adversarial attacks—another reason to be skeptical that machine learning can provide a silver bullet for cyber defenders.

RESPONSE AND RECOVERY

FIGURE 5
AI Applications for Response and Recovery



CYBERSECURITY TASK	UNDERLYING TECHNOLOGY OF NEW AI APPLICATIONS	SIGNIFICANCE OF TASK	TRANSFORMATIVE POTENTIAL OF NEW AI
Adversary Engagement	Reinforcement Learning	High	Medium-Low
Moving Target Defense	Reinforcement Learning	High	Medium-High

While there is a great deal of research on ML-driven detection systems, more ambitious proposals posit AI systems that might one day move through networks autonomously, patching vulnerabilities and fighting dynamically against attackers. After all, in recent years researchers have developed ML systems that can master competitive games as diverse as Go or StarCraft.[60] But as complex as these games are, they are comparatively more structured and easier to model than the nebulous world of cybersecurity. Furthermore, in most of the other stages of cybersecurity, ML progress has come from automating discrete and self-contained (yet increasingly complex) tasks, rather than from attempting to build ML systems that can operate fully autonomously. Response and recovery, by contrast, is a dynamic and continuous process that is not as easily broken into discrete components, which makes it much more difficult to build ML tools that can adequately automate human decision-making.

In recent years, some authors have begun to identify more targeted roles that AI/ML systems could play in the response and recovery process. A 2020 report from the National Science & Technology Council, for instance, identifies at least two

concrete ways in which AI could aid the response and recovery process: by accurately categorizing ongoing attacks and selecting an appropriate initial response strategy, or by automating the decision to isolate machines from a network or impose user restrictions to contain infections.[61] The first type of tool could allow cyber defenders to automate their initial responses to a wide variety of common types of attacks, while the second type would be generally useful in containing security breaches from spreading to other parts of a network.*

Despite being identified as a potential area of opportunity, we are not aware of any significant research demonstrating the viability of a ML system that could automate a wide range of responses to multiple types of attacks.† In part, this absence is likely due to the fact that such a goal remains too broad to be effectively automated by current ML methods. At the same time, there have been some attempts to further break this goal down, with the intention of building ML systems that can intelligently respond to specific *types* of attacks. As perhaps the most high-profile example, DARPA is currently sponsoring a project, ongoing since 2017, to create autonomous systems that can respond to botnets. If successful, this program would result in autonomous systems that can identify botnet-infected machines, select publicly-known vulnerabilities to deploy against them in order to remove malware, and move laterally to neutralize adjacent compromised machines.[62]

There has been more progress in building ML models that can learn to isolate potentially compromised machines from a network in order to contain a security breach. This type of application is an example of moving target defense, a defensive strategy in which defenders try to impede attackers by dynamically restructuring parts of their IT infrastructure in response to an attack. Reinforcement learning in particular has been useful in exploring this topic. Several researchers have attempted to model network systems under attack in order to test the ability of reinforcement learning agents to respond. In one example, researchers allowed a defending agent with imperfect knowledge of an attacker's actions to reimage—or restore to factory settings—machines which it suspected may have become infected. By imposing a cost upon the defender for each reimage it made, the researchers demonstrated that a reinforcement learning–based defender could learn the appro-

*In Figure 5, we refer to the first application as "adversary engagement," which would be the ultimate goal of an ML system that can understand ongoing attacks and respond accordingly. The second goal is a simple example of moving target defense.

† Here it is worth again reiterating that the focus of this paper is specifically on machine learning and not on other types of technological advances that might be called "AI." There have been more efforts to address this sort of problem using non-ML methods, but we do not comment on the success of these efforts here.

priate cost-benefit tradeoff and make useful decisions regarding when to reimage.[63] Similar approaches have allowed reinforcement learning agents to learn when to isolate potentially infected nodes within a constrained network or to develop game theoretic strategies for adaptively responding to adversaries.[64]

These results are promising, because they suggest that machine learning could be useful for automating tactics like moving target defense or for providing responses to some types of threats, such as botnets. But it remains far from certain that such tools will become broadly useful. The examples mentioned above were the result of extremely simplified simulations of real networks, and as with pentesting, it is not clear that the complexity of the simulation can be reasonably scaled up without becoming too computationally intensive to allow for real-time decision-making. Moreover, machine learning continues to be most successful in contexts with relatively clearly-defined problems and outcomes. Although it may become increasingly useful as a way to implement specific defensive strategies in response to various threats, we do not foresee it becoming capable in the short- to medium-term of fully automating the work that human analysts do when investigating the sources of a compromise and formulating strategic responses.

ACTIVE DEFENSE

FIGURE 6

AI Applications for Active Defense

CYBERSECURITY TASK	UNDERLYING TECHNOLOGY OF NEW AI APPLICATIONS	SIGNIFICANCE OF TASK	TRANSFORMATIVE POTENTIAL OF NEW AI
Deceptive Document Generation	NLP, GANs	Medium-Low	High
Dynamic Honeypotting	Reinforcement Learning	Medium	Medium-High
Automated Phishing Response	NLP, Reinforcement Learning	Low	Medium-High
Dark Web Threat Intelligence	NLP	Medium	Medium
Attack Clustering for Attribution	NLP, Traditional ML Methods	Medium-Low	Medium
Code De-Anonymization	NLP	Low	Medium-Low

Organizations with significant cybersecurity needs must often take proactive steps to shape their cybersecurity strategies in response to new threats. This report groups these actions under the broad framework of "active defense." Though this term has a specific national security definition, it is here used analogously to the way it has been used by researchers at the SANS Institute: as a spectrum of activity ranging from annoyance to attribution to outright counterattacks.[65] This section sets aside the issue of outright counter-attack and instead focuses on three less legally fraught tactics: deception, threat intelligence, and attribution. The wide range of potential AI applications across these three tactics suggests that active defense may be one of the stages of our cybersecurity model with comparatively much to gain from recent ML breakthroughs.

Deception

One of the most basic active defense measures available to cyber defenders is deception: intentionally faking something to mislead and slow down attackers. Though this strategy seems simplistic, it can have major operational benefits. In the lead-up to the 2017 French presidential election, for instance, the Macron campaign—aware that it was being targeted by Russian hackers—chose to forge internal emails full of misleading or outlandish content. Russian hackers could not easily distinguish real information from forgeries, and although they eventually chose to leak everything, this strategy effectively allowed the Macron campaign to minimize the public perception of the issue.[66] Deployed well, deceptive behavior can also cause attackers to second-guess themselves or otherwise manipulate their attention towards unproductive activities.[67]

Generating realistic-looking documents or activity profiles is an area where machine learning excels. The rise of massive natural language models like GPT-3 will likely make the process of generating deceptive textual documents—like the Macron campaign's fake emails—relatively easy to automate. While this technology has an enormous potential for misuse, defenders could plausibly use it for deceptive purposes, generating waves of deceptive internal documents to mislead attackers. The use of other types of ML models—like GANs—could be used for non-textual deception. For instance, researchers have shown that it is possible to use a GAN to generate realistic network data that might be used to confuse attackers.[68] (An analogous type of forgery was used by the winning team of the 2016 Cyber Grand Challenge, which used machine learning to create traffic that looked like an attack and fooled opposing teams into wasting time trying to stop it.[69])

Beyond forgeries, the honeypot—a security mechanism designed to lure cyber attackers to attractive-seeming but fake targets—is a long-standing deceptive tool for network defenders. Honeypots can offer "bait" in the form of fake files and data, run scripted interactions to thwart adversaries, or mislead attackers into thinking that they have been more successful than they have been. However, honeypots often require manual configuration, deployment, and maintenance to look like realistic production networks long enough to keep the attackers engaged, which can be costly and makes them difficult for smaller organizations to maintain.

Since at least the early 2000s, researchers have explored how machine learning could be used to create more realistic and dynamic honeypots, including through the use of relatively simple clustering methods.[70] In more recent years, researchers have experimented with reinforcement learning–based honeypots that can learn to engage attackers for as long as possible, while also ideally tricking them into downloading custom software onto their own machines.[71] It would be premature to say that the use of reinforcement learning is a game-changer in this

area, but it may at least make these types of honeypots more accessible to a larger number of organizations by automating much of the complexity of maintaining a dynamic honeypot.

Threat Intelligence

Gathering threat intelligence about potential adversaries can theoretically allow defenders to anticipate attacks and build stronger defenses, though for most organizations, the labor-intensive nature of collecting, processing, and analyzing threat intelligence makes it cost-prohibitive. But because finding patterns in data is a core strength of machine learning, some researchers have explored how ML and text mining can be used to improve threat intelligence analysis. For instance, ML methods can be used to cluster dark web users, or text mining methods could be leveraged to automatically collect, classify, and analyze posts on dark web forums and marketplaces, allowing researchers to identify zero-day exploits before they are deployed.[72] It is not difficult to imagine a fully automated ML system that could be given a company's name or a list of its products and told to search for mentions of potential vulnerabilities on the dark web, analyze them, and generate a report describing anticipated vectors of attack.

Other tools introduced as deceptive tactics could be repurposed to collect threat intelligence about potential adversaries. Data collected from honeypot systems, for instance, can be analyzed and collated to determine if an organization is facing a persistent threat actor, which can then inform security strategies.[73] Even something as simple as a phishing email could become an opportunity to collect threat intelligence about an adversary. Some researchers have proposed systems that can produce misleading responses to phishing emails to elicit information from attackers, for instance by appearing willing to give money to a scam but asking for the attackers' routing numbers, which can then be traced and identified.[74] In fact, this is a major line of research within the Active Social Engineering Defense initiative at DARPA, which is currently attempting to build systems that can detect social engineering attacks and respond autonomously to trick adversaries into revealing identifying information.[75] These tactics cross the line between threat intelligence collection and outright attribution, the third active defense tactic discussed in this section.

Attribution

When facing an attack from an advanced persistent threat, federal agencies and cybersecurity companies will typically make an effort to attribute the attack to a specific adversary. Doing so enables federal actors to determine what type of response is warranted, and it helps cybersecurity companies determine the likely motivations of hostile actors in order to best defend their customers. Other companies are less likely to engage in attribution work. While there is some strategic

value to being able to share precise information about adversaries with other organizations, the fact that most corporations cannot legally respond in kind to an adversary severely reduces the value of a successful attribution, particularly considering how difficult the process can be.

When trying to attribute multiple attacks, the simplest approaches rely on manual analysis of attack indicators obtained through network logs, firewalls, or honeypots. However, because most persistent cyber attackers can avoid leaving obvious patterns, attribution also relies on human analysis and an understanding of potential adversaries and their overarching goals; attribution therefore relies not only on technical but also strategic indicators and, potentially, knowledge about relevant geopolitical situations. This is a level of analysis that machine learning is not well-equipped to carry out, so it remains unlikely that machine learning will be able to successfully automate the full process of attribution.

Nonetheless, machine learning may still be able to assist in the attribution process. For example, if research analysts first extract high-level descriptions of an attack—including the tactics used, the country of attack, and so forth—ML methods may be able to cluster this information to identify similar attacks.[76] Natural language processing may help attribution systems automatically extract attribution-relevant information from blogs, research papers, and incident response reports, reducing the amount of work that humans need to do to make a successful attribution.[77]

Applying ML methods to an attacker's source code, when available, may offer a different path to attribution. ML methods have already shown the ability to distinguish and de-anonymize authors based upon their distinctive writing styles.[78] In recent years, some researchers have demonstrated that human generated code works in a similar way: different programmers make unique choices as they code, which allows machine learning to create digital fingerprints that can be used to identify coders.[79] By comparing to databases of malicious code, machine learning may soon be able to cluster bits of code written by the same coder, providing one more clue in the attribution process. Attackers might always further muddy the waters by repurposing code from other sources, and the fact that advanced persistent threats typically contain numerous independent coders makes digital fingerprinting difficult. But the possibility is nonetheless intriguing.

Some of these applications are speculative. There are notable limitations with relying on ML techniques for attribution; for example, threat data can often be inconsistent if it is collected from multiple security companies. Still, integrating high-level cyber threat behavior with machine learning is an understudied area, and further research may allow federal agencies and cybersecurity companies to augment their attribution toolkit, while also making some simpler forms of attribution available to other types of organizations.

Conclusions

This report has summarized a range of potential ways AI could be used to aid cyber defenders as well as some of the limitations of current techniques. We offer four major conclusions for what this means for the actual practice of cybersecurity:

1. At the detection stage, ML will allow for continued improvements over existing detection tools, especially for companies with extensive big data capabilities. However, these developments should be viewed as incremental advances that introduce new attack surfaces of their own.

2. At the prevention, response and recovery, and active defense stages, the use of ML is less commonplace but is making progress. Some of the most exciting advances will still require major ML breakthroughs in order to be fully deployable.

3. Overall, we anticipate that gains from ML will continue to be incremental rather than transformative for the cybersecurity industry. Barring major ML breakthroughs, the most transformative impacts may come from ML's ability to make previously un- or under-utilized defensive strategies more accessible to many organizations.

4. At the present time, machine learning is unlikely to fundamentally shift the strategic balance of cybersecurity towards attackers or defenders. However, for specific applications, the introduction of machine learning techniques may be more useful to one side or the other.

In focusing on the underlying technical capabilities of machine learning, this report has set aside many other issues, such as whether or not defenders will successfully implement ML systems and fully leverage their potential. These conclusions are preliminary, not predictive: whether they hold true will depend on how attackers and defenders choose to invest in and deploy this technology.

ML CAN IMPROVE DETECTION, BUT IT BRINGS RISKS OF ITS OWN

Machine learning has played an increasingly large role within attack detection for several decades, and the coming years will likely make it even more important. Among other things, this is due to the development of better algorithms and the use of greater computational power.

But classification—whether of specific attack patterns or "anomalies" more generally—is a task where quality data makes or breaks the model. Because of the critical importance of data for training classification algorithms, the benefits of improved detection systems will be most easily leveraged by companies with the ability to collect the most cyber data. This is one area where defenders have an asymmetric advantage relative to attackers: defenders can collect and store far more data about their own networks than attackers can, which makes it possible for defenders to continually improve their defenses.

Nonetheless, these benefits are likely to be incremental for three reasons. First, because the size and scale of cyber attacks continues to rise, it is unclear how important the gains from better ML models will be. Slow and steady improvement in algorithms and data collection abilities may only be enough to keep pace with the threat landscape, without allowing defenders to get ahead.

Second, attackers have an asymmetric advantage of their own: an attack only has to work once, while a defensive strategy has to work all the time. Considering that attackers can use GANs to generate malicious traffic that resembles benign traffic enough to evade many types of ML-powered and rules-based defenses, the defender's data advantage may not be enough to protect them from the offensive use of ML methods.[80] Further, because most network data tends to be highly variable even under normal conditions, building algorithms that can reliably identify malicious anomalies without flagging legitimate behavior is a difficult task. This increases the chances that motivated attackers—especially those using heuristic or ML tactics to camouflage themselves within a network's background traffic—will continue to find ways to slip past even the best detection systems.

Finally, the benefits of newer ML-based detection systems may be less long-lasting than many hope, since ML models bring new attack avenues of their own. These

attacks can be internal to the ML algorithms themselves, which rely on datasets that attackers could poison and which attackers can often circumvent using subtly altered adversarial examples.[81] Moreover, the decisions of ML models can also be opaque, and when they behave in strange or unproductive ways, defenders may be tempted to make adjustments that introduce additional vulnerabilities, as appears to have been the case in the Cylance model discussed above.

None of this is meant to trivialize the important role that machine learning can play in improving detection systems. Even if machine learning cannot vastly improve baseline detection abilities, deep learning models in particular may be useful for performing rudimentary triage of alerts in order to identify the behaviors that are most likely indicative of malicious activity. But most of these developments represent incremental improvements that require the context of the above caveats. While the developments are important, we do not view them as likely to significantly transform the nature of threat detection.

PROGRESS EXISTS AT OTHER STAGES BUT MAY STILL AWAIT MAJOR BREAKTHROUGHS

While machine learning has already been widely deployed for various types of detection, it remains rarer in the other stages of the cybersecurity model. Nonetheless, researchers have slowly been exploring the use of machine learning for a diverse range of applications. Table 2 summarizes each of the major ML applications for each stage that this report has covered.

At the prevention stage, ML systems could one day allow for improved fuzzing systems that software developers can use to detect vulnerabilities before they are exploited, or for pentesting aids that can more effectively search a network for vulnerabilities than current tools can. ML systems may also allow defenders to more accurately identify potential threats by triaging bug reports and attempting to automatically predict the potential severity of a newly discovered vulnerability.

At the response and recovery stage, there has been some progress in building reinforcement learning systems that are capable of learning when to sequester potentially compromised machines from a network in order to limit an attacker's lateral movement. There has also been growing interest in the use of machine learning to automate defensive strategies, as demonstrated by DARPA's ongoing project to build autonomous systems that can neutralize compromised devices within an attacking botnet.

TABLE 2

Summary of AI Applications

PREVENTION	DETECTION	RESPONSE AND RECOVERY	ACTIVE DEFENSE
Fuzzing	Accurate Detection	Adversary Engagement	Deceptive Document Generation
Pentesting	Alert Prioritization	Moving Target Defense	Dynamic Honeypotting
Bug Triage and Classification	Adversarial Hardening of Defense Systems		Automated Phishing Response
Vulnerability Severity Assessment			Dark Web Threat Intelligence Collection
			Attack Clustering for Attribution
			Code De-Anonymization

At the active defense stage, there is a broad and increasing number of applications for which machine learning could be adapted to help defenders deceive, analyze, and attribute attackers. Dynamic honeypots, deceptive document generation, and phishing response may soon leverage reinforcement learning and/or advanced natural language models. In addition, advances in natural language processing may improve threat intelligence collection from dark web sources. Finally, clustering methods and advanced code analysis may allow for more sophisticated attribution methods, including some that may become available to organizations that have not traditionally invested in attribution work.

Major advances within these applications could meaningfully transform cybersecurity by automating many tasks that typically consume a cyber defender's time or by adding new streams of information and analysis to the defender's situational awareness. Nonetheless, these advances may still require significant breakthroughs of uncertain likelihood. For instance, the computational intensity of reinforcement learning in highly complex environments continues to make it difficult to build ML systems that can handle the complexity of cybersecurity networks. Building deployable reinforcement learning–based cyber responders that could fully mimic human expertise would require major breakthroughs in the ability of researchers to effi-

> Major advances within these applications could meaningfully transform cybersecurity by automating many tasks that typically consume a cyber defender's time or by adding new streams of information and analysis to the defender's situational awareness. Nonetheless, these advances may still require significant breakthroughs of uncertain likelihood.

ciently represent all relevant network data. Machine learning will likely become an increasingly important part of each of these stages in the cybersecurity model, but transforming the work of the cyber defender may still be out of reach for the near- to medium-term.

EXPECT ML GAINS TO BE INCREMENTAL RATHER THAN TRANSFORMATIONAL

Machine learning can offer a great deal to cyber defenders across all four stages of the cybersecurity model used in this paper. Nonetheless, whether or not it is a truly "transformational" technology depends very heavily on what standards are used to assess its impact. There is no doubt that machine learning can make significant improvements on a variety of cybersecurity technologies. If it were not for machine learning, defenders today would likely be consumed with low-level analysis that is more often delegated to automated systems which frequently make heavy use of ML systems. This is especially true of a task like spam detection—which would likely be all but impossible at scale without machine learning—but it is also true of intrusion and malware detection.

By counterfactual standards, then, machine learning is indeed a transformative technology: without it, cybersecurity today would have a very different—and very worse—record of success. But this does not mean that machine learning is a truly transformational technology by historical standards. While machine learning successfully automates many low-level tasks today, the rapidly growing number of threats and sophisticated attacks facing most organizations means that ML-based

Policymakers should not anticipate that developing better ML tools is a replacement for addressing other, more fundamental cybersecurity problems, such as encouraging cybersecurity workforce development or enforcing secure practices across federal agencies.

detection systems have mostly only allowed cyber defenders to keep pace with the threat landscape. So far, machine learning has not transformed the core type of work that cyber defenders are required to perform, nor has it made the current cybersecurity skills shortage significantly less worrisome.[82]

By historical standards, we conclude that machine learning is not presently a transformational technology. This does *not* mean that ML is unimportant for cyber defenders. Continued investment in both researching and implementing promising ML-cyber tools should be an important area of focus for cyber defenders, and should be promoted by the federal government wherever possible. But policymakers should not anticipate that developing better ML tools is a replacement for addressing other, more fundamental cybersecurity problems, such as encouraging cybersecurity workforce development or enforcing secure practices across federal agencies.

While we do not think that machine learning is poised to transform the cybersecurity industry, this requires two caveats: First, fundamental breakthroughs in ML research may significantly change the threshold of possibility. For instance, if researchers are able to develop computationally simpler methods of simulating complicated networks, reinforcement learning–based tools may be able to take on significantly more complex tasks than what is currently possible. While we do not currently anticipate machine learning being able to automate many high-level decisions during the incident response process, a sufficiently large breakthrough might change that assessment and enable ML tools that could be given substantially more autonomy.

Second, not all the defensive strategies discussed in this report are equally common, which means that machine learning may be able to have a larger impact on some areas than others—especially where it can make previously un- or under-utilized strategies more accessible for more organizations. For instance, some survey results suggest that while upwards of 90 percent of organizations use antivirus products, the number of organizations actively using honeypots or related

technologies may be much lower—potentially as low as 25 percent.[83] Improvements in the use of machine learning for malware detection are useful and important—but they are unlikely to change many organizations' underlying defensive strategies. By contrast, if ML tools make it possible to easily deploy honeypots that can realistically integrate into an existing network, many more organizations might use honeypots.

This would in some ways represent a larger shift in the threat landscape than improvements in malware detection tools, because the widespread deployment of a new strategy could have more significant impacts on attackers' decision-making processes than an incremental improvement on an existing strategy. As another example, some researchers have explored how the presence of deceptive tactics such as decoys can disorient attackers and shape their strategies.[84] If ML makes it easy to automatically create deceptive internal records or engage in moving target defense, for instance, then widespread adoption of the technologies may meaningfully shape attacker strategies, even if the ML models themselves are not particularly impressive. The mere knowledge that such deceptive tactics are more commonly used may be enough to influence the attacker's decision-making.

ML APPLICATIONS PRESENT NO CLEAR-CUT ADVANTAGE TO OFFENSE OR DEFENSE ON NET

Most of the ML applications this report has discussed are dual-use: they could be used either by attackers or defenders. In some cases, the opportunity for attackers is obvious, as in the cases of ML-powered fuzzing and pentesting. In other cases, attackers might find creative ways to repurpose or circumvent defensive technologies. For instance, if machine learning is used to calculate the odds of an exploit being used, attackers might simply incorporate this information into their decisions regarding which exploits to develop, making the model increasingly useless as attackers adapt and the training data becomes unrepresentative of actual realities.

Because of these dual-use considerations, ML will not give a clear-cut advantage either to attackers or to defenders in the coming years. However, while ML will not clearly benefit either attackers or defenders *overall*, certain specific applications of ML may be either offense-biased or defense-biased.

For example, the rise of sophisticated natural language processing models is likely to improve spearphishing abilities.[85] With the ability to generate detailed, fluent, and personalized text or voice recordings, attackers will be able to engage in more effective social engineering campaigns. Although the same natural language processing models can be adapted for defensive purposes—say, to build chatbots that can mislead social engineers—these are reactive measures and should not distract from the important takeaway: barring major developments, machine learning will allow attackers to conduct more effective social engineering campaigns in

the near future. Therefore, the use of machine learning for text generation is likely an offense-biased technology.

At the same time, there is arguably a larger number of ML applications that may be primarily useful for defenders. For instance, attack detection is a major focus of ML research for cybersecurity, but attackers may have relatively little use for attack detection technologies, since there is little need to detect an attack that you yourself are responsible for.* Similarly, the use of dynamic honeypots are more obviously relevant for organizations seeking to protect their networks than for attackers hoping to penetrate them, which makes ML-based honeypots another generally defense-biased technology. These are technologies where defenders may generally gain more than attackers from increases in ML adoption.

SUMMARY

Policymakers and practitioners alike need to think about how machine learning can alter *specific* tasks within cybersecurity, rather than talking in general terms about how machine learning can alter cybersecurity as a whole. We have attempted to provide greater clarity by surveying a range of potential machine learning applications and the factors that continue to constrain their development. In addition, by focusing on the continuity in the use of machine learning for cybersecurity tasks dating back to the 1990s, we have aimed to demonstrate that newer breakthroughs will bring *evolutionary* cybersecurity gains rather than *revolutionary* changes to the industry.

It is possible that one day, machine learning tools will automate entire swathes of our cybersecurity model at once, seamlessly performing the work of multiple stages instead of functioning as tools that merely aid in already-existing, well-defined tasks. But in the near- to medium-term future, this possibility is remote. It is far more pragmatic for policymakers and practitioners to work towards a more nuanced understanding of the types of tasks that could benefit from machine learning and the types of tasks that can't. Effective policy will need to take these nuances into account to promote useful types of research, enable organizations to take full advantage of new machine learning tools, and help defenders prepare against their constantly improving adversaries.

*Attackers who study newer detection models may be able to find ways to craft adversarial attacks that can bypass them, so progress in this area is only *strictly* defense-biased if it results in detection systems that are less vulnerable to adversarial attacks. Since it is not clear that this is the case with newer ML detection models, it is unclear whether current improvements in ML detection systems are capable of shifting the overall strategic balance towards defenders.

Appendix

	BASIC STRUCTURE	ADVANTAGES	DISADVANTAGES
DEEP LEARNING	Deep learning is based on the use of artificial neural networks and may include hundreds or thousands of nodes whose structure can be adapted for specific tasks.	1) Performance can often improve with more data, well after performance in simpler models flattens 2) Can be used as an underlying approach for most AI applications 3) Models can often automatically learn to extract relevant features from data	1) Requires significant processing power to run and significant memory to store due to model size 2) Widely regarded as one of the most opaque types of machine learning 3) Benefits of more data may be difficult to fully leverage in the cyber domain, which is often very secretive
REINFORCEMENT LEARNING	Reinforcement learning simulates an agent with several available actions, an environment, and a reward function and allows the agent to experiment with the use of different actions to learn strategies that maximize rewards.	1) Allows for AI agents that can learn strategic thinking 2) Can be combined with deep learning systems to improve performance 3) Does not require historical data if realistic environments can be simulated	1) Computational cost increases extremely quickly when the environment complexity or number of actions are increased 2) Difficult to simulate realistic yet computationally tractable cyber networks
GENERATIVE ADVERSARIAL NETWORKS (GANs)	A GAN includes two neural networks: a generator that uses random input to create new inputs and a discriminator that attempts to distinguish real from generated inputs.	1) Can be combined with deep learning systems to improve performance 2) Can be used to augment existing data sets or evade existing classifiers	1) Does not have general-purpose applications like deep learning 2) Typical caveats about the need for robust underlying data apply
MASSIVE NATURAL LANGUAGE MODELS	These models are not unified by a single underlying structure but are instead characterized by their use of massive models that can include billions of parameters in order to learn patterns within human language.	1) Increasingly capable of automating text generation, summarization, and translation, among other tasks 2) Could be used to automate security tasks that involve written text (e.g. generating bug reports)	1) Cannot be built by individual companies without investing millions of dollars; access will depend on how corporate owners choose to make them available

Endnotes

1. For spam detection, see James Vincent, "Gmail Is Now Blocking 100 Million Extra Spam Messages Every Day with AI," *The Verge*, February 6, 2019, https://www.theverge.com/2019/2/6/18213453/gmail-tensorflow-machine-learning-spam-100-million. For malware detection, see Lily Hay Newman, "Gmail Is Catching More Malicious Attachments With Deep Learning," *WIRED*, February 25, 2020, https://www.wired.com/story/gmail-catching-more-malicious-attachments-deep-learning/.
2. Stephan Somogyi and Allison Miller, "Safe Browsing: Protecting more than 3 billion devices worldwide, automatically," *Google Security Blog*, September 11, 2017, https://security.googleblog.com/2017/09/safe-browsing-protecting-more-than-3_11.html.
3. Catalin Cimpanu, "Symantec, ESET, McAfee rank first in Windows anti-malware market share," *ZDNet*, November 18, 2019, https://www.zdnet.com/article/symantec-eset-mcafee-rank-first-in-windows-anti-malware-market-share; Microsoft Defender ATP Research Team, "Windows Defender ATP Machine Learning: Detecting New and Unusual Breach Activity," Microsoft, August 3, 2017, https://www.microsoft.com/security/blog/2017/08/03/windows-defender-atp-machine-learning-detecting-new-and-unusual-breach-activity/; Beth Stackpole, "Machine Learning: A Modern-Day Cyber Security Hero?" Broadcom, June 4, 2019, https://symantec-enterprise-blogs.security.com/blogs/feature-stories/machine-learning-modern-day-cyber-security-hero; Ondrej Kubovič et al., "ESET Advanced Machine Learning" (ESET, November 2019), https://www.welivesecurity.com/wp-content/uploads/2019/11/ESET_Advanced_Machine_Learning.pdf.
4. See Ben Buchanan, *The Hacker and the State: Cyber Attacks and the New Normal of Geopolitics* (Cambridge, MA: Harvard University Press, 2020).
5. Rob Sobers, "110 Must-Know134 Cybersecurity Statistics and Trends for 20210," Inside Out Security Blog on Varonis, accessed June May 2517, 20210, https://www.varonis.com/blog/cybersecurity-statistics/.
6. Ben Buchanan, John Bansemer, Dakota Cary, Jack Lucas, and Micah Musser, "Automating Cyber Attacks" (Center for Security and Emerging Technology, November 2020), https://doi.org/10.51593/2020CA002.
7. For commentary in popular media organizations, see Danny Palmer, "AI Is Changing Everything about Cybersecurity, for Better and for Worse. Here's What You Need to Know," *ZDNet*, March 2, 2020, https://www.zdnet.com/article/ai-is-changing-everything-about-cybersecurity-for-better-and-for-worse-heres-what-you-need-to-know/; Lily Hay Newman, "AI Can Help Cybersecurity—If It Can Fight Through the Hype," *WIRED*, April 29, 2018, https://www.wired.com/story/ai-machine-learning-cybersecurity/. For perspectives on autonomous cyber defenders in the defense community, see Carlo Munoz, "DARPA Sees 'Rich Space' for Advanced AI in Cyber Operations," Janes, August 3, 2020, https://www.janes.com/defence-news/news-detail/darpa-sees-rich-space-for-advanced-ai-in-cyber-operations; Paul Théron and Alexander Kott, "When Autonomous Intelligent Goodware Will Fight Autonomous Intelligent Malware: A Possible Future of Cyber Defense," in *Proceedings of the Military Communications Conference*, (MILCOM-2019, November 12-14, 2019), https://arxiv.org/ftp/arxiv/papers/1912/1912.01959.pdf.
8. "Darktrace Antigena: Product Overview," Darktrace, accessed April 30, 2021, https://www.darktrace.com/en/resources/ds-antigena.pdf. "Project Blackfin," F-Secure, accessed May 6, 2021, https://www.f-secure.com/en/about-us/research/project-blackfin.
9. Although there have been some improvements in the algorithms used to build ML systems, many researchers agree that the advances of the last decade have had far more to do with increased data availability and computing power, rather than improvements in algorithms. For a discussion of the various parts of the "AI triad," see Ben Buchanan, "The AI Triad and What It Means for National Security Strategy" (Center for Security and Emerging Technology, August 2020), https://doi.org/10.51593/20200021.

10. Tami Casey, "Survey: 27 Percent of IT professionals receive more than 1 million security alerts daily," *Imperva*, May 28, 2018, https://www.imperva.com/blog/27-percent-of-it-professionals-receive-more-than-1-million-security-alerts-daily/.
11. For a competing view, see David Brumley, "Why I'm not sold on machine learning in autonomous security," CSO Online, August 27, 2019, https://www.csoonline.com/article/3434081/why-im-not-sold-on-machine-learning-in-autonomous-security.html.
12. "Framework for Improving Critical Infrastructure Cybersecurity: Version 1.1," National Institute of Standards and Technology, April 16, 2018, https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04162018.pdf.
13. "M-Trends 2020" (FireEye, 2020), 11–15, https://content.fireeye.com/m-trends/rpt-m-trends-2020.
14. Christopher Jarko, "Finding the Fine Line — Taking an Active Defense Posture in Cyberspace without Breaking the Law or Ruining an Enterprise's Reputation" (SANS Institute, February 6, 2016), https://www.sans.org/reading-room/whitepapers/legal/finding-fine-line-active-defense-posture-cyberspace-breaking-law-ruining-enterprise%E2%80%99s-reputation-36807.
15. Paul Graham, "A Plan for Spam," August 2002, http://www.paulgraham.com/spam.html.
16. For an example of an automated tool that can perform feature extraction, see Wadi' Hijawi et al., "Improving Email Spam Detection Using Content Based Feature Engineering Approach," in *2017 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)* (ieeexplore.ieee.org, 2017), 1–6.; for a discussion of how machine learning can be used to aid feature extraction, see Melvin Diale, Turgay Celik, and Christiaan Van Der Walt, "Unsupervised Feature Learning for Spam Email Filtering," *Computers & Electrical Engineering* 74 (March 1, 2019): 89–104.
17. For a review of common machine learning methods used for spam detection, see Emmanuel Gbenga Dada et al., "Machine Learning for Email Spam Filtering: Review, Approaches and Open Research Problems," *Heliyon* 5, no. 6 (June 2019): e01802.
18. Anne Johnson and Emily Grumbling, "Implications of Artificial Intelligence for Cybersecurity: Proceedings of a Workshop (2019)" (The National Academies Press, 2019), 25–26, https://doi.org/10.17226/25488.
19. Much academic research on intrusion detection systems continues to rely on these datasets generated between 1998 and 2000. A 2018 review estimated that roughly 75% of the previous decade's top-cited papers on ML for intrusion detection relied on one of these datasets or their later variants. See H. Hindy et al., "A Taxonomy of Network Threats and the Effect of Current Datasets on Intrusion Detection Systems," *IEEE Access* 8 (2020): 104650–75. Unfortunately, there are few up-to-date public datasets available for researchers to use in their research instead, which means that reliance on these datasets continues to be a problem despite well-known flaws with this data. (For a discussion of some of the flaws in this data, see R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," in *2010 IEEE Symposium on Security and Privacy*, 2010, 305–16.) This is a major area where policymakers may be able to advance the research community's ability to develop better methods by funding the creation of newer datasets for academic use, though we do not develop this suggestion in the present work.
20. Richard Lippmann et al., "The 1999 DARPA Off-Line Intrusion Detection Evaluation," *Computer Networks* 34, no. 4 (October 1, 2000): 579–95.
21. Yang Xin et al., "Machine Learning and Deep Learning Methods for Cybersecurity," *IEEE Access* 6 (2018): 35365–81.
22. Accurate descriptions of model architectures in corporate products are difficult to find. However, as one example for the claim that deep learning is not fully ubiquitous, consider Google Play Protect, Google's machine learning–based approach for detecting malicious apps in the Android store, which it calls "the most widely deployed mobile threat protection service in the world." (See "Google Play Protect: 2.5 Billion Active Devices," Android, accessed October 22, 2020, https://www.android.com/play-protect/.) As recently as November 2018, Google's research blog claimed that Google Play Protect used both deep learning and other, much simpler approaches such as logistic regression—a simple mathematical tool as old as the 19th century. Mo Yu, Damien Octeau, and Chuangang Ren, "Combating Potentially Harmful Applications with Machine Learning at Google: Datasets and Models," *Google Security Blog*, November 15, 2018, https://security.googleblog.com/2018/11/combating-potentially-harmful_14.html.

23. Francesco Palmieri, Ugo Fiore, and Aniello Castiglione, "A Distributed Approach to Network Anomaly Detection Based on Independent Component Analysis," *Concurrency and Computation: Practice & Experience* 26, no. 5 (April 10, 2014): 1113–29; Yu Gu, Andrew McCallum, and Don Towsley, "Detecting Anomalies in Network Traffic Using Maximum Entropy Estimation," Internet Measurement Conference, 2005, https://www.usenix.org/legacy/event/imc05/tech/full_papers/gu/gu.pdf.
24. See, for instance, the discussion of false positives in Johnson and Grumbling, "Implications of Artificial Intelligence for Cybersecurity: Proceedings of a Workshop (2019)," 23, 26–27.
25. Stephen Verbeke, "COVID-19's Impact on Cybersecurity Incident Response," Novetta, May 26, 2020, https://www.novetta.com/2020/05/cyber-covid/.
26. Nicholas Duran et al., "2018 Webroot Threat Report," Webroot, 2018, https://www-cdn.webroot.com/9315/2354/6488/2018-Webroot-Threat-Report_US-ONLINE.pdf.
27. Mustafa Irshad et al., "Effective Methods to Detect Metamorphic Malware: A Systematic Review," *International Journal of Electronic Security and Digital Forensics* 10, no. 2 (January 2018): 138–54.
28. G. J. Tesauro, J. O. Kephart, and G. B. Sorkin, "Neural Networks for Computer Virus Recognition," *IEEE Expert* 11, no. 4 (August 1996): 5–6.
29. For research exploring the use of Hidden Markov models, see: Wing Wong and Mark Stamp, "Hunting for Metamorphic Engines," *Journal in Computer Virology* 2, no. 3 (December 1, 2006): 211–29.; for a general summary of much of this research, see Muazzam Ahmed Siddiqui, Morgan Wang, and Joohan Lee, "A Survey of Data Mining Techniques for Malware Detection Using File Features," in *ACM-SE 46: Proceedings of the 46th Annual Southeast Regional Conference on XX* (Auburn, Alabama: Association for Computing Machinery, March 2008), https://doi.org/10.1145/1593105.1593239.
30. Daniel Gibert, Carles Mateu, and Jordi Planes, "The rise of machine learning for detection and classification of malware: Research developments, trends and challenges," *Journal of Network and Computer Applications* 153 (March 2020), https://doi.org/10.1016/j.jnca.2019.102526.
31. Hemant Rathore et al., "Malware Detection Using Machine Learning and Deep Learning," *arXiv [cs.CR]* (April 4, 2019), arXiv, http://arxiv.org/abs/1904.02441.
32. Sophisticated malware has grown increasingly better at detecting that it is being run in sandboxes and changing or stopping its own execution through a number of measures, such as delaying execution, hardware detection, or requiring input from the user. See Thomas Roccia and Chintan Shah, "Evolution of Malware Sandbox Evasion Tactics – A Retrospective Study," McAfee, September 9, 2019, https://www.mcafee.com/blogs/other-blogs/mcafee-labs/evolution-of-malware-sandbox-evasion-tactics-a-retrospective-study/.
33. S. L. S. Darshan, M. A. A. Kumara, and C. D. Jaidhar, "Windows Malware Detection Based on Cuckoo Sandbox Generated Report Using Machine Learning Algorithm," in *2016 11th International Conference on Industrial and Information Systems (ICIIS)*, 2016, 534–39; Chani Jindal et al., "Neurlux: Dynamic Malware Analysis Without Feature Engineering," *arXiv [cs.CR]* (October 24, 2019), arXiv, https://doi.org/10.1145/3359789.3359835.
34. Thanassis Avgerinos et al., "The Mayhem Cyber Reasoning System," *IEEE Security & Privacy* 16, no. 2 (March/April 2018): 55–56.
35. It is worth noting that the head of the 2016 winning team, David Brumley, remains skeptical about the use of AI in cybersecurity. See David Brumley, "Why I'm not sold on machine learning in autonomous security," *CSO Online*, August 27, 2019, https://www.csoonline.com/article/3434081/why-im-not-sold-on-machine-learning-in-autonomous-security.html.
36. Yan Wang et al., "A Systematic Review of Fuzzing Based on Machine Learning Techniques," *PloS One* 15, no. 8 (August 18, 2020): e0237749.
37. Yunchao Wang et al., "NeuFuzz: Efficient Fuzzing With Deep Neural Network," *IEEE Access 7* (January 2019): 36340–52.
38. Mohit Rajpal, William Blum, and Rishabh Singh, "Not All Bytes Are Equal: Neural Byte Sieve for Fuzzing," *arXiv [cs.SE]* (November 10, 2017), arXiv, http://arxiv.org/abs/1711.04596.
39. *Metasploit-Framework* (Github), accessed September 10, 2020, https://github.com/rapid7/metasploit-framework.

40. Dean Richard McKinnel et al., "A Systematic Literature Review and Meta-Analysis on Artificial Intelligence in Penetration Testing and Vulnerability Assessment," *Computers & Electrical Engineering* 75 (May 1, 2019): 175–88.
41. Fabio Massimo Zennaro and Laszlo Erdodi, "Modeling Penetration Testing with Reinforcement Learning Using Capture-the-Flag Challenges and Tabular Q-Learning," *arXiv [cs.CR]* (May 26, 2020), arXiv, http://arxiv.org/abs/2005.12632.
42. Mohamed C. Ghanem and Thomas M. Chen, "Reinforcement Learning for Efficient Network Penetration Testing," *Information–An International Interdisciplinary Journal* 11, no. 1 (December 20, 2019): 6.
43. Jonathon Schwartz and Hanna Kurniawati, "Autonomous Penetration Testing Using Reinforcement Learning," *arXiv [cs.CR]* (May 15, 2019), arXiv, http://arxiv.org/abs/1905.05965.
44. Demis Hassabis, *The Power of Self-Learning Systems*, Video (United States: Center for Brains, Minds & Machines, March 2019), https://cbmm.mit.edu/video/power-self-learning-systems.
45. Catalin Cimpanu, "Cobalt Strike and Metasploit accounted for a quarter of all malware C&C servers in 2020," *ZDNet*, January 7, 2021, https://www.zdnet.com/article/cobalt-strike-and-metasploit-accounted-for-a-quarter-of-all-malware-c-c-servers-in-2020/.
46. For an initial attempt to assign bug reports to developers, see John Anvik, Lyndon Hiew, and Gail C. Murphy, "Who Should Fix This Bug?," in *Proceedings of the 28th International Conference on Software Engineering*, ICSE '06 (New York, NY: Association for Computing Machinery, 2006), 361–70; for an attempt to classify security and non-security bugs, see Shaikh Mostafa et al., "SAIS: Self-Adaptive Identification of Security Bug Reports," *IEEE Transactions on Dependable and Secure Computing*, 2019, 1–1. For automated generation of malware characteristics, see Swee Kiat Lim et al., "MalwareTextDB: A Database for Annotated Malware Articles," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Vancouver, Canada: Association for Computational Linguistics, 2017), 1557–67.
47. Mayana Pereira and Scott Christiansen, "Identifying Security Bug Reports Based Solely on Report Titles and Noisy Data," Microsoft, March 6, 2020, https://docs.microsoft.com/en-us/security/engineering/identifying-security-bug-reports.
48. Georgios Spanos, Lefteris Angelis, and Dimitrios Toloudis, "Assessment of Vulnerability Severity Using Text Mining," in *Proceedings of the 21st Pan-Hellenic Conference on Informatics*, PCI 2017 49 (New York, NY: Association for Computing Machinery, 2017), 1–6.
49. Jay Jacobs et al., "Exploit Prediction Scoring System (EPSS)," *arXiv [cs.CR]* (August 13, 2019), arXiv, http://arxiv.org/abs/1908.04856.
50. Jay Jacobs et al., "Improving Vulnerability Remediation through Better Exploit Prediction," *Journal of Cybersecurity* 6, no. 1 (September 14, 2020), https://doi.org/10.1093/cybsec/tyaa015.
51. See note 7, above.
52. For a primer on this topic, see Tim G. J. Rudner and Helen Toner, "Key Concepts in AI Safety: Robustness and Adversarial Examples" (Center for Security and Emerging Technology, March 2021), https://doi.org/10.51593/20190041.
53. For a primer on threats to AI systems, see Andrew Lohn, "Hacking AI" (Center for Security and Emerging Technology, December 2020), https://doi.org/10.51593/2020CA006.
54. "Cylance, I Kill You!" Skylight, July 18, 2019, https://skylightcyber.com/2019/07/18/cylance-i-kill-you/.
55. Ram Shankar Siva Kumar, "Azure Sentinel Uncovers the Real Threats Hidden in Billions of Low Fidelity Signals," Microsoft, February 20, 2020, https://www.microsoft.com/security/blog/2020/02/20/azure-sentinel-uncovers-real-threats-hidden-billions-low-fidelity-signals/. This is also an area of active interest in the defense community; see Jason Miller, "Defense Innovation Unit Out to Prove AI, Automation Can Keep Up With the Speed of Cyber," *Federal News Network*, October 30, 2020, https://federalnewsnetwork.com/ask-the-cio/2020/10/defense-innovation-unit-out-to-prove-ai-automation-can-keep-up-with-the-speed-of-cyber/ and Lauren C. Williams, "AI Meets Cyber as Army Tests Tactical Network Protection," Federal Computer Week, November 4, 2020, https://fcw.com/articles/2020/11/04/netmodx-army-tactical-test.aspx.

56. Zilong Lin, Yong Shi, and Zhi Xue, "IDSGAN: Generative Adversarial Networks for Attack Generation Against Intrusion Detection," *arXiv [cs.CR]* (September 6, 2018), arXiv, http://arxiv.org/abs/1809.02077; Aritran Piplai, Sai Sree Laya Chukkapalli, and Anupam Joshi, "NAttack! Adversarial Attacks to Bypass a GAN Based Classifier Trained to Detect Network Intrusion," *arXiv [cs.LG]* (February 20, 2020), arXiv, http://arxiv.org/abs/2002.08527; Weiwei Hu and Ying Tan, "Generating Adversarial Malware Examples for Black-Box Attacks Based on GAN," *arXiv [cs.LG]* (February 20, 2017), arXiv, http://arxiv.org/abs/1702.05983. GANs are not the only type of AI architecture that can be used to generate adversarial attacks to evade malware detection or intrusion detection; see, e.g., Hyrum S. Anderson et al., "Learning to Evade Static PE Machine Learning Malware Models via Reinforcement Learning," *arXiv [cs.CR]* (January 30, 2018), arXiv, https://arxiv.org/abs/1801.08917.
57. Muhammad Usama et al., "Generative Adversarial Networks For Launching and Thwarting Adversarial Attacks on Network Intrusion Detection Systems," in *2019 15th International Wireless Communications Mobile Computing Conference (IWCMC)* (Tangier, Morocco: 2019), 78–83.
58. Gray Stanton and Athirai A. Irissappane, "GANs for Semi-Supervised Opinion Spam Detection," *arXiv [cs.LG]* (March 19, 2019), arXiv, http://arxiv.org/abs/1903.08289; Hojjat Aghakhani et al., "Detecting Deceptive Reviews Using Generative Adversarial Networks," *arXiv [cs.CR]* (May 25, 2018), arXiv, http://arxiv.org/abs/1805.10364; Jin-Young Kim, Seok-Jun Bu, and Sung-Bae Cho, "Zero-Day Malware Detection Using Transferred Generative Adversarial Networks Based on Deep Autoencoders," *Information Sciences* 460–461 (May 1, 2018), https://doi.org/10.1016/j.ins.2018.04.092.
59. See Florian Tramèr et al., "On Adaptive Attacks to Adversarial Example Defenses," *arXiv [cs.LG]* (October 23, 2020), arXiv, https://arxiv.org/abs/2002.08347.
60. See David Silver et al., "AlphaZero: Shedding new light on chess, shogi, and Go," DeepMind, December 6, 2018, https://deepmind.com/blog/article/alphazero-shedding-new-light-grand-games-chess-shogi-and-go; The AlphaStar Team, "AlphaStar: Grandmaster level in StarCraft II using multi-agent reinforcement learning," DeepMind, October 30, 2019, https://deepmind.com/blog/article/AlphaStar-Grandmaster-level-in-StarCraft-II-using-multi-agent-reinforcement-learning.
61. National Science and Technology Council, *Artificial Intelligence and Cybersecurity: Opportunities and Challenges: Technical Workshop Summary Report* (Washington, DC: Executive Office of the President, March 2020), 6, https://www.nitrd.gov/pubs/AI-CS-Tech-Summary-2020.pdf.
62. Dustin Fraze, "Harnessing Autonomy for Countering Cyberadversary Systems (HACCS)," DARPA, accessed October 22, 2020, https://www.darpa.mil/program/harnessing-autonomy-for-countering-cyberadversary-systems. Note that this program has not yet concluded, making it difficult to ascertain its success and even, to a certain extent, how its ultimate goals have developed over time.
63. Taha Eghtesad, Yevgeniy Vorobeychik, and Aron Laszka, "Adversarial Deep Reinforcement Learning Based Adaptive Moving Target Defense," *arXiv [cs.CR]* (November 27, 2019), arXiv, http://arxiv.org/abs/1911.11972.
64. Ahmad Ridley, "Machine Learning for Autonomous Cyber Defense," *The Next Wave*, 2018, 7–14; Thanh Thi Nguyen and Vijay Janapa Reddi, "Deep Reinforcement Learning for Cyber Security," *arXiv [cs.CR]* (July 21, 2020), arXiv, https://arxiv.org/abs/1906.05799.
65. Jarko, "Finding the Fine Line — Taking an Active Defense Posture in Cyberspace without Breaking the Law or Ruining an Enterprise's Reputation."
66. Heather A. Conley and Jean-Baptiste Jeangène Vilmer, "Successfully Countering Russian Electoral Interference" (Center for Strategic and International Studies, June 21, 2018), https://www.csis.org/analysis/successfully-countering-russian-electoral-interference.
67. See, for instance, Robert Gutzwiller et al., "'Oh, Look, a Butterfly!' A Framework for Distracting Attackers to Improve Cyber Defense," in *Proceedings of the Human Factors and Ergonomics Society 2018 Annual Meeting*, 272-276.
68. Markus Ring et al., "Flow-based network traffic generation using Generative Adversarial Networks," *Computers & Security* 82 (May 2019): 156-172.
69. National Academies of Sciences, Engineering, and Medicine, "Implications of Artificial Intelligence for Cybersecurity: Proceedings of a Workshop" (Washington, DC: The National Academies Press, 2019), 2-5.

70. W. Z. Ansiry Zakaria and M. L. M. Kiah, "A Review on Artificial Intelligence Techniques for Developing Intelligent Honeypot," in *2012 8th International Conference on Computing Technology and Information Management (NCM and ICNIT)*, vol. 2 (ieeexplore.ieee.org, 2012), 696–701; D. Fraunholz, M. Zimmermann, and H. D. Schotten, "An Adaptive Honeypot Configuration, Deployment and Maintenance Strategy," in *2017 19th International Conference on Advanced Communication Technology (ICACT)* (ieeexplore.ieee.org, 2017), 53–57.
71. Adrian Pauna and Ion Bica, "RASSH - Reinforced Adaptive SSH Honeypot," in 2014 *10th International Conference on Communications (COMM)*, 2014, 1–6.
72. Kylie Foy, "Artificial Intelligence Shines Light on the Dark Web," MIT News, May 13, 2019, https://news.mit.edu/2019/lincoln-laboratory-artificial-intelligence-helping-investigators-fight-dark-web-crime-0513; Eric Nunes et al., "Darknet and Deepnet Mining for Proactive Cybersecurity Threat Intelligence," *arXiv [cs.CR]* (July 28, 2016), arXiv, http://arxiv.org/abs/1607.08583.
73. Daniel Fraunholz et al., "YAAS-On the Attribution of Honeypot Data," IJCSA 2, no. 1 (2017): 31–48.
74. Adam Dalton et al., "The Panacea Threat Intelligence and Active Defense Platform," *arXiv [cs.CL]* (April 20, 2020), arXiv, http://arxiv.org/abs/2004.09662.
75. Walter Weiss, "Active Social Engineering Defense (ASED)," DARPA, accessed October 22, 2020, https://www.darpa.mil/program/active-social-engineering-defense.
76. Umara Noor et al., "A Machine Learning-Based FinTech Cyber Threat Attribution Framework Using High-Level Indicators of Compromise," *Future Generations Computer Systems: FGCS* 96 (July 1, 2019): 227–42; Steve Miller and Tom Davenport, "Machine Learning Support for Cyber Threat Attribution at FireEye," *FireEye*, June 15, 2020, https://www.fireeye.com/blog/products-and-services/2020/06/machine-learning-support-for-cyber-threat-attribution-at-fireeye.html.
77. Microsoft Defender Security Research Team, "From Unstructured Data to Actionable Intelligence: Using Machine Learning for Threat Intelligence," Microsoft, August 8, 2019, https://www.microsoft.com/security/blog/2019/08/08/from-unstructured-data-to-actionable-intelligence-using-machine-learning-for-threat-intelligence/.
78. Chen Qian, Tianchang He, and Rao Zhang, "Deep Learning Based Authorship Identification," Department of Electrical Engineering, Stanford University, 2017, https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1174/reports/2760185.pdf.
79. Louise Matsakis, "Even Anonymous Coders Leave Fingerprints," *WIRED*, August 10, 2018, https://www.wired.com/story/machine-learning-identify-anonymous-code/.
80. See note 54, above.
81. See Lohn, "Hacking AI."
82. The fact that machine learning has not, to date, meaningfully transformed the day-to-day of a typical cybersecurity defender can be somewhat substantiated by examining hiring trends in the cybersecurity industry. If ML were not merely a useful tool, but an innovation that was actively reshaping the life of cyber defenders, we would expect to see a marked increase in demand for cyber defenders with technical knowledge of ML systems. But, as other research from CSET has demonstrated, the increase in demand for employees with even very limited knowledge of both cybersecurity and ML has grown only slowly over the previous decade. See Cindy Martinez and Micah Musser, "U.S. Demand for Talent at the Intersection of AI and Cybersecurity" (Center for Security and Emerging Technology, November 2020), https://doi.org/10.51593/2020CA009.
83. Andrea Dominguez, "The State of Honeypots: Understanding the Use of Honey Technologies Today" (SANS Institute, November 2017), https://www.sans.org/reading-room/whitepapers/detection/state-honeypots-understanding-honey-technologies-today-38165.
84. See Gutzwiller, "'Oh, Look, a Butterfly!'"
85. Buchanan et al., "Automating Cyber Attacks."

CSET.GEORGETOWN.EDU | CSET@GEORGETOWN.EDU