Undermining Trust in Learning and Inference Systems

Adversarial attacks seeking to specifically undermine foundational trust within human-machine teams raise concerns of stability and safety for learning and inference systems in military, healthcare, finance, and other domains.

Dustin D. Updyke

Thesis Committee:

Prof. Kevin J.S. Zollman, Carnegie Mellon University (Chair)

Prof. David Danks, UC San Diego

A thesis presented for the degree of
Master of Science in Logic, Computation, and Methodology

Department of Philosophy,
Carnegie Mellon University
20 MAR 2023

Stephen Falken: *General, what you see on these screens up here is a fantasy.*
A computer-enhanced hallucination.
Those blips are not real missiles.
They're phantoms!

McKittrick: *General, there is nothing to indicate a simulation at all.*
Everything is working perfectly!

Stephen Falken: *But does it make any sense?*
General, are you prepared to destroy the enemy?

General Beringer: *You betcha!*

Stephen Falken: *Do you think they know that?*

General Beringer: *I believe we've made that clear enough.*

Stephen Falken: *Then don't.*
General, you are listening to a machine.
Do the world a favor and don't act like one.

— *WarGames (1983)* [1]

[1] https://www.imdb.com/title/tt0086567/

Contents

Chapter 1

Introduction

We are inclined to trust computers as we would any other tool, through motivations based upon repeated reliability, and the execution of defined tasks whose functions are well understood. With a laptop in every hand, and a cell phone in every pocket, it is hard to recall a time when we could not depend upon computers to help us live better lives. We have become dependent upon computer systems that help us navigate city traffic, interact with healthcare professionals, and stay connected with family members across the globe.

However, the rise of learning and inference systems change the fundamental relationship between us and machine, in that the computer is now less of a tool and more of a teammate. These types of systems, including artificial intelligence systems, make important decisions on our behalf or replace the need for our intervention entirely. In addition, while the activities performed by learning and inference systems are expanding, they are not always well defined, nor do we often know how such a system makes decisions and executes tasks. Still, when computers operate as we expect, they can improve our lives in many ways, and we can hardly envision how we ever did without them.

But computers also operate in ways that we don't expect. They are subject to an array of security concerns. When a computer provides surprising results, perhaps these can directly affect our trust in such as system. If we come to not trust a system that we reply upon, what happens?

Since we depend upon computers for so much in the modern age, we should assume that trust will likely be the target of adversaries. What is particularly worrisome is that these attacks can essentially modify our expected results but without informing us of what exactly has changed, or how this has occurred. Let's consider several historical examples of how this scenario might evolve.

1.0 Wargames

On 9 November, 1979, the North American Aerospace Command's (NORAD) early warning system interpreted a training scenario involving Soviet submarines as an actual nuclear attack on the United States. In the six minutes that followed, the American military went on the highest level of alert, as analysts struggled to reconcile computer reports with those from ground commanders. [2, 3]

Senior military officials attempting to make the most important decision of their career suddenly could not trust the early warning computer systems they had come to

[2] Campbell, "Nuclear War and Computer-Generated Nuclear Alerts."

[3] Burr, "The 3 A.M. Phone Call: False Missile Attack Warning Incidents, 1979-1980."

rely upon when the stakes were highest. Fortunately, in the middle of that fall night, under a very tight timeline, the situation was successfully deescalated. In the aftermath, many questions were asked of how to ensure that this scenario could not happen again—where the early warning system that a nation depends upon would always show the ground truth that commanders can trust to help them make informed decisions. In essence, the systems at NORAD are needed to be reliable one hundred percent of the time, since every second counts for a response that is potentially paid for in human life. Doubt in an early warning system can directly equate to lives lost.

The events at NORAD inspired the popular 1983 movie *WarGames*.[4] The real-life incident and the movie's adaptation highlight a fundamental tension between humans and computers: *How do we come to trust a complex computer system, and how is that trust broken?*

1.1 Stuxnet

In early 2010, a computer virus that would come to be known as Stuxnet was lurking within industrial control systems around the world. Stuxnet had a great deal of sophistication in its abilities—it could alter the pressure inside of a nuclear reactor, control the flow of an oil pipeline, or destroy a uranium enrichment cascade.[5] More worryingly, it could perform these operations while assuring system administrators that everything was functioning as normal.[6]

Most researchers attribute the malware to the joint efforts of the US and Israeli governments, a multi-year project codenamed *Operation Olympic Games*, although neither country has formally acknowledged the program. Analysts largely agree that Stuxnet's aim was to slow the Iranian nuclear enrichment program in Natanz. With increasing tensions between Israel and Iran, Olympic Games had begun in 2007 as a clandestine way to delay, deter, and perhaps stop Iran's nuclear ambitions. At the time, Iran lacked the precision technology to manufacture centrifuge rotors that spin at a constant 63,000 revolutions per minute for months, if not years. This meant using older equipment and replacing parts more often. To counter this, the Iranians employed computerized industrial controls that closely monitored the machinery to identify potential maintenance problems early. In a sense, the people working within the uranium enrichment group depended upon industrial controls and computer monitoring systems functioning as a teammate that continually observed and ensured the smooth operation of the enrichment cascades. With this strategy, Iran's nuclear effort was able to make notable progress.

After the initial malware infection, Stuxnet would sit in between the cascade hardware and the computer—what is effectively called a man-in-the-middle attack—diligently observing the environment before modifying centrifuge configurations in order to degrade their performance or outright destroy them. The virus did this in a manner that confounded the Iranians, as it overrode the control system's output to report

[4] Kaplan, "'WarGames' and Cybersecurity's Debt to a Hollywood Hack."
[5] Langner, "To Kill a Centrifuge."
[6] Zetter, *Countdown to Zero Day.*

back the exact values that system administrator computers—and by extension, system administrators themselves—would be expecting. For those monitoring the project throughout Stuxnet's infection, everything was operating as usual.

Stuxnet did little to reveal its presence and its operations were careful to reveal as little as possible. Well into the second half of 2010, the International Atomic Energy Agency (IAEA), which globally oversees all nuclear energy and weapons activities, noted that the number of failed centrifuges at Natanz was trending very high on aggregate.[7] Yet in Iran, the virus remained unnoticed. Instead, the failures were attributed to a lack of access to the latest centrifuge technology and to using relatively dated equipment.

Only once Stuxnet was identified and detailed by an outside research firm in Belarus did the virus cause Iran's nuclear program leadership to suspend work at all affected facilities.[8] By the end of the year, Iran conceded that the virus infection of its operations, including the—then under construction—Bushehr nuclear power plant, meant that its switching on could very well lead to a national electrical grid blackout or worse.

The plant at Bushehr did not open until almost an entire year later, well behind its intended schedule. Iran's official statement was that Stuxnet had nothing to do with the delay. Security experts have argued over the effectiveness of the virus in limiting Iran's nuclear refinement ambitions. [9, 10] Yet, leaving impact aside, there seems to be a correlation between the virus's identification and the suspension of the enrichment program.

Here the interesting context for us to consider is Stuxnet's erosion of trust from the enrichment program team to the industrial control systems upon which they depended. If the team at Natanz could no longer be certain that centrifuge performance matched what sensors were reporting, they could not be certain to make accurate predictions about the maintenance or output of such systems. That monitoring was such a critical component of their success, the Iranian enrichment team may well have decided to shut down the entire program to uncover the source and scope of the deception.

There are two notions of trust to note here. First, human operators trusted computerized control systems to tell them the truth, or in other words, to tell them something they needed to know accurately. Second is the sense of trusting that a teammate will do as instructed. Humans expected computer systems to perform specific actions based on input. The former example is an epistemic sense of trust, and the latter is a moral conception of trust. Concerning Stuxnet, human operators

[7] Albright, Brannan, and Walrond, "Did Stuxnet Take Out 1,000 Centrifuges at the Natanz Enrichment Plant?"

[8] Chen and Abu-Nimeh, "Lessons from Stuxnet."

[9] Chen, "Stuxnet, the Real Start of Cyber Warfare?"

[10] Farwell and Rohozinski, "Stuxnet and the Future of Cyber War."

could not have been sure that their system counterparts had not violated both types of trust.

A simple question might be for us to ask, did the Iranians lose trust in their computer teammate? If they did, how did this happen?

1.2 Project Lakhta

By 2017, an emerging story of the Russian government's multi-year campaign to interfere with the 2016 United States Presidential election though a vast disinformation operation began to take shape. It would come to be known as *Project Lakhta*.

The beginning of this effort traces back to the outbreak of the 2014 Russo-Ukrainian war, where Russia used cyberattacks to disrupt many aspects of Ukrainian life, including power utilities, financial systems, and importantly, the disruption of that year's Ukrainian presidential election. The election attacks included the attempted alteration of vote tallies, denial-of-service attacks designed to delay final voting results, and the release of hacked emails to sway public opinion. Previous cyberattack campaigns against Estonia in 2007 or Georgia in 2008 are often also linked to Russia, but these were largely denial-of-service in nature and did not specifically target the public's trust. Noted RAND scholar Martin Libicki wrote that in Ukraine 2014, "the information and propaganda war in the social media domain (particularly from the Russian side) has been relentless."[11] Between 2008 and 2014, Russian cyberstrategy had turned its eye from destruction and interdiction to influencing public opinion and sowing dissent.

Sometime in the beginning of 2013, the Russian Internet Research Agency (IRA)—an organization described by the American intelligence community as a "troll farm"[12]—on orders from Russian President Vladimir Putin, began a campaign to increase American political and social discord. Targeting public opinion, the operation attempted to damage the political campaign of Hillary Clinton, but also to boost that of Donald Trump. The IRA created numerous fabricated American social media accounts which created, shared, and liked content in support of Trump and against Clinton. Using these accounts, invented disinformation news articles and posts were spread from Russian government-controlled media into platforms such as Facebook, Twitter, and others.

At almost the same time, Russian Military Intelligence Service (GRU) operatives[13] hacked into the Democratic National Committee (DNC), the Democratic Congressional Campaign Committee (DCCC), and specific Hillary Clinton campaign

[11] Libicki and Geers, *The Cyber War That Wasn't. Cyber War in Perspective: Russian Aggression against Ukraine.* 50-51.

[12] Institutionalized trolling was not reserved to Russian organizations, the 2017 *Freedom on the Net* report shows that of the countries studied, at least 30 employed "keyboard armies" to foster online misinformation and erode public trust. See Freedom House, "New Report - Freedom on the Net 2017."

[13] Often referred to as "Cozy Bear," "Fancy Bear," or "APT28".

official's computer systems, and in turn released stolen files and emails. These disclosures coincided with the 2016 election campaign, and directly overlapped the disinformation efforts of the IRA, further complicating user efforts to determine the validity of associated social media posts and news articles and creating ample fodder for a continued flow of conspiracies and suspicion.

Versions of the malware that enabled GRU operators to gain an initial foothold in these hacks—code often referred to as *Dark Energy*—was found on the servers of both the Ukrainian Central Election Commission and the DNC.[14, 15] During this time, US national security officials warned the Obama administration that Russia was building a disinformation program which could be used to interfere in Western politics.[16]

By 2016, Project Lakhta had brought strong opposition statements from US government officials and intelligence agencies, plus an investigation by the Federal Bureau of Investigation (FBI). This case evolved into the Special Counsel investigation led by former FBI director Robert Mueller. It found that Russian interference was "sweeping and systematic" and brought federal indictments for twenty-six operatives within three different Russian organizations. It also led to the conviction of Trump campaign personnel due to numerous contacts between them and Russian officials, the grounds being that the Trump campaign expected to benefit from Russian activities. Beyond this, efforts to find evidence for the direct coordination of these activities by Trump associates failed.

By the US election in November 2018, it is estimated that IRA fabricated accounts had reached 126 million Americans. This resulted in significant backlash towards social media platforms for failing to protect their users, and for allowing false information to be spread so easily to such a wide number of users.

In the aftermath of Project Lakhta[17] much effort has gone into studying its effect on trust in democratic institutions, but also in the resulting falling trust in social media platforms and the algorithms they employ.

1.3 Timeline

The NORAD concerns of 1979—how we come to trust a complex computer system, and how that trust is ultimately broken—resurfaced with the discovery of Stuxnet, the world's first cyber weapon, which took such great efforts to conceal its mission, and to convince those that might have detected its activity that nothing was amiss. Much has been made of Stuxnet's sophistication and ability to bridge digital malware to having impact in the physical world, and less of its direct manipulation of our trust in

[14] Kramer and Higgins, "In Ukraine, a Malware Expert Who Could Blow the Whistle on Russian Hacking."

[15] Greenberg, *Sandworm: A New Era of Cyberwar and the Hunt for the Kremlin's Most Dangerous Hackers.*

[16] Watkins, "Obama Team Was Warned in 2014 about Russian Interference."

[17] It should be noted that even as the Lakhta story began to solidify, even to the point of the DOJ issuing 26 named indictments, the project continued through the 2018 midterms, and potentially influenced the 2020 elections as well.

computer systems that we have come to rely upon.[18] Yet, if we focus on the erosion of trust within Stuxnet, then we see it as one of the logical predecessors to Project Lakhta, just that the latter had far greater reach and impact.

Prior to Lakhta, the Russians had perhaps already learned from, or directly been a part of large-scale organized cyberattacks on Estonia in 2007 and Georgia in 2008. If the logical progression of the attacks on Ukraine in 2014 were to include the erosion of public trust in the Ukrainian presidential election, then perhaps Russian planners had learned a great deal from Stuxnet and *Olympic Games*–a large scale, intimately planned effort, with long-term strategic goals. Indeed, the timeline would have matched up with Stuxnet occurring between these other events.

While many have and continue to focus on Stuxnet's ability to hamper or destroy centrifuges in the spirit of slowing Iranian progress, fewer have focused on its affect upon trust within the Iranian nuclear program. Perhaps the organizers of Lakhta noted the latter phenomenon and used that knowledge to plan a new campaign to erode the American people's trust in social and political institutions.

We have no evidence to fully corroborate this timeline, and even if that evidence were to exist, it would likely lie buried under countless layers of geopolitical secrecy. Yet for our purposes, it remains a story of how we might expect adversarial trust attacks to evolve. As an example, it can still be useful to explore the conclusions should it turn out to have been the case. In many respects, "could have been" situations very much inform our preparedness to deal with exact or like scenarios, should they ever come to be. The timeline laid out here is drawn in this spirit, and the hope is that its conclusions provide valuable insight into what we might expect from trust-based attacks in the future, and how we might work to mitigate their impact.

Since Stuxnet is arguably the first example of a cyberattack that successfully eroded operator trust, we will use it as the example of trust-eroding that we will refer to throughout this thesis. Arguably, with the rise of complexity in the computer age, these early examples of trust issues between humans and computers are simply the first of many to come.[19]

Along the way, we must discuss the bond we have with computers. As it matures in capability and power, our relationship with the computer continues to transform from

[18] With regards to Stuxnet and other adversarial malware attacks relating to trust, several authors have focused on the role of signed certificates—and how they are distributed through Certificate Authorities (CAs)—as indicators of software that can be trusted. Stuxnet malware was signed by stolen legitimate certificates and brought into question the security surrounding CAs, and what trust we should place within them. See Love and McMillin, Jenkinson, and Shakarian. Issues of trust in CAs is outside the scope of this work.

[19] A related example is Boeing's *Maneuvering Characteristics Augmentation System* (MCAS). While not a learning and inference system per se, it does illustrate how, despite our best intentions of building genuinely helpful computer systems, we cannot always foresee all possible situations such a system will confront, nor can we always successfully limit its impact on those situations (see the Lion Air crash of 2018, and the Ethiopian Airlines crash of 2019). Surely there are other examples beyond the scope of this thesis. For our purposes we will use Stuxnet as the early example of a trust-based attack.

a simple tool to that of a complex teammate.[20, 21] Today, we are perhaps as likely to rely on a computer system in key decision-making as we are to depend upon a fellow human being. Learning and inference systems now provide data-driven reasoning that significantly augments and extend our own abilities.[22] Combining the best human and computer capabilities into a *human-machine team* has become increasingly important in a wide array of research and industry contexts.

The central argument of this work is that some computers—learning and inference systems, specifically—have become sufficiently complex, and that lends to difficulty in our understanding the scope of their operation and how they function. This, coupled with our reliance upon them to make decisions within an array of domains gives birth to a new concern within cybersecurity: That adversaries will specifically seek to undermine our trust in these decision-making computer systems. Unlike adversarial attacks of the past, trust-based attacks will be more challenging to detect and defend against, due to the inherent complexity of the system under attack. The aftermath of a successful trust-based attack will have a different set of concerns to address to "correct" those compromised computer systems and the human teammates that depend upon them. Stuxnet is an early example of these types of attacks.

What does it mean to depend upon a computer system? Human-machine teams are much like any sports team playing effectively together, where players must trust one another to achieve maximum effectiveness; Mach et al. find that "*high levels of team trust are related to team performance with both direct and indirect effects*."[23] When players do not trust each other, there is hesitancy. Hesitation leads to missed opportunities. Plays and movements fail to materialize, and objectives slip away. This breakdown at the individual level can lead to less trust in the environment overall. If we assume this same notion of trust holds for a human-machine team, then it should be possible for the human element to come not to trust a computer teammate. If that is the case, then it follows that the performance of a human-machine team is also negatively affected.

In the event of a potentially trust-eroding situation, we should note that human teammates can, at the very least, attempt to interrogate further or explain some piece of evidence that has different interpretations. As human teammates, we might work together to change our beliefs about a point of contention—some event that has occurred where we have come to alternative conclusions about what evidence means.

While rebuilding trust between humans is complex due to a range of factors—psychological and sociological, for example—it is equally so for computer systems, both in the broadest sense of computing, and specifically for learning and inference systems.[24] As noted above, humans can have a discussion in an attempt to resolve the issue. In the case of computers, who do we have this discussion with? We could

[20] Lyons et al., "Chapter 6 - Trust and Human-Machine Teaming."

[21] Stowers et al., "Improving Teamwork Competencies in Human-Machine Teams."

[22] Seeber et al., "Machines as Teammates."

[23] Mach, Dolan, and Tzafrir, "The Differential Effect of Team Members' Trust on Team Performance."

[24] In the absence of general artificial intelligence that could answer all manner of inquiry directly.

choose to talk to human engineers who build and program the systems on which a computer operates. Yet, there is typically a large divide between users of such systems and those who build them. As users, we likely have no direct contact with those engineers, and perhaps it takes significant effort to reach them. Even if we could have a discussion, we likely would not have a shared vocabulary. Could we try to interrogate the computer directly (maybe by asking a personal digital assistant questions)? This would be equally difficult due to its narrow and restricted input and output possibilities built into such a system. This is likewise true of our ability to explain our own conclusions to a computer system, or to question those of the computer—the computer is equally limited in its potential to be understood. As is it today within a human-machine team, we cannot explain our human perceptions and reasonably expect the computer to be able to answer a generalized line of questioning. Even simple questions that seek to clarify trust-eroding situations would be challenging, and those involving computer systems are no less so than their human counterpart.[25]

These difficulties in trusting computer systems trace back to the question that *WarGames* originally posed, how do we come to trust (or not trust) a computer system upon which we are dependent? If we have a reasonable understanding of this question, we might go on to think about how an adversary undermines that trust.

If Stuxnet holds some part of the answer, then its manipulation of the industrial control software would be of primary interest. Unfortunately, in the cyberwarfare era, adversaries often target software of all kinds, and do so in part because its vulnerabilities continue to bear fruit. Since their inception, computers have always had inherent concerns of security and trustworthiness, and we see this influence on United States government policy as far back as the early 1960s.[26] These weaknesses continue today, but over a greatly expanded attack vector, as computing continues to pervade much of our lives, with consumer companies seeking to offer an increasing number of "smart" devices. For all the good that computers have brought about, there is still a great deal of discussion surrounding computer vulnerabilities and exploits—the domain of what I will call *traditional cybersecurity concerns*, detailed in the next chapter.

More troubling worries lie beyond these initial traditional cybersecurity issues, however. If a computer is acting in the role of a teammate, then trust must underly the relationship between the human members of this team and the computing system. Apple CEO Tim Cook recently declared, "*Technology will only work if it has people's trust.*"[27] Yet surprisingly, there is little discussion about trust being the specific target that an adversary might seek to exploit. While attacks that seek to disrupt, halt, or deceive the computer systems themselves are relatively straightforward in either

[25] I should also be careful to distinguish that for a certain class of AI systems (e.g., Project Lakhta, where Russian operatives manipulated the algorithm in order to target specific other users of the system, not the system itself), the question might be whether we lose trust in the system, or the other users of that system, or both. I suspect that we lose trust in the system for not safeguarding a user from malicious users and manipulation, but that is future work to be done.

[26] Warner, "Notes on the Evolution of Computer Security Policy in the US Government, 1965-2003."

[27] Aten, "With 9 Words, Tim Cook Just Explained the Biggest Problem with Facebook."

having evidence of exploit or not, attacks on the trustworthiness of a system seem less clear. Rather than a binary indication of compromise, the trust we might have in a system is a spectrum, and we can have degrees of trust in such a system because of the evidence we observe. This graded nature of trust also indicates that we must build it over time and not assume that it is inherent from the outset.

Human-machine teams (HMTs) feature humans depending upon a computer system as a teammate who informs and assists with data-driven decision-making. Machine learning (ML) and artificial intelligence (AI) systems are a subset of this computing class by extension. Pairing human beings with such systems attempt to highlight the best attributes in both while minimizing any shortcomings found in either.

We primarily focus on AI as a particular subset of learning and inference systems in the following chapters. We do this in part because they show future promise in so many domains and because they add such complexity that they perhaps show the most significant potential for the central phenomenon within this thesis. Additionally, we will leverage the recent literature on the rise of HMTs paired with AI systems.

Throughout, we will be interested in computer systems that depend upon information in order to make decisions, often on behalf of a user. A simple example would be that as we browse a news website, the server knows the topics and stories we have already browsed and uses this information to decide what stories to display next, prioritizing the information that we might find most relevant. Computers have rapidly growing access to information by two mechanisms: First, every interaction that we have with a computer system offers information for the computer to make inferences about our decision-making process that they might then apply to any decisions they make for us on our behalf—regardless of whether we are aware of such execution. Secondly, computers themselves create a large amount of information as either the byproduct of past decisions already made, the import of relevant data from some disparate system, or aggregated data. Increasingly, many computer systems use data to aid our use of such systems in offering “preferenced,” “discriminating,” “smart,” or “intelligent” capabilities that streamline our interactions—making these systems faster to use, or that they reduce what we must know to implement such a system into our daily lives most effectively.

The simplest form of these systems is our Stuxnet example at Natanz—while it was not specifically a learning system, recall that Stuxnet did observe previous operational data in order to know what values operators would have expected. With this data, it could implement an array of inferences for determining what a watching analyst would expect to see. Although this method did not have the system learn over time, its form of inference remains sufficiently complex for this thesis. At Natanz, computer monitoring systems and their administrators made decisions based on data, and I will argue that relationship can be targeted by trust-based attacks.

Circling back to shortcomings: Any computer will always be subject to a set of concerns regarding its security and potential exploit, and those involved with HMTs are no exception. In addition, learning and inference systems are susceptible to a unique set of attacks to complicate this landscape further. While there has been widespread worry about cybersecurity attacks that might seek to destabilize or even halt the operation of such systems, there has been little focus on specifically attacking the underlying trust that the humans within an HMT often place in these systems. I provide the background and a model for how one type of attack on that underlying trust might occur. Later, I discuss the consequences of a potentially successful attack that removes our ability to trust a system or the results that it ultimately produces.

Recently, AI has come under criticism regarding privacy, explainability, bias, and trust. Under what conditions can we claim that an AI provides results that we can trust? Can we trust results for which we have no concrete explanations of how a system calculated a result? While these are indeed interesting questions, in the interest of focus, I cannot attempt answers to them here. Instead, I will assume some conditions for which we can trust results. I make this assumption on the fact that there are many HMTs currently in operation around the world that rely on an AI system operating as a teammate, providing information that drives decision-making. For our purposes, it perhaps matters not whether a human being or an AI system directly decides an outcome. Instead, simply that AI played some part in generating or analyzing the information that eventually generated the decision, regardless of who made the final call.

Relatedly, I want to clarify a mis-generalization of trust that often confuses discussions involving human confidence in AI systems and their results. As we will see, some aspects of computing match how we use other types of tools to accomplish specific tasks. Yet, a growing aspect of computing clearly does not match our notion of a tool. This is to say that we come to trust some aspects of computing, such as a calculator, in the same sense that we might come to trust any other tool, such as a carpenter's hammer or a gardener's shovel. While there are motivations that we follow in coming to trust how a calculator will operate given specific input, I will argue that sort of trust does not generalize properly to AI systems, and particularly where these systems are serving in a role as any sort of teammate.

While Stuxnet is not a learning and inference system specifically, it did read and play back values that administrators expected to see. It is my hope that it provides a clear illustration of the trust-erosion phenomenon that we will discuss in the chapters ahead, particularly in terms of the model experiments we create later in chapter four. If we can model these worries for the simpler Stuxnet scenario, we can intuit that these worries would be worse in a complex learning and inference system.

Chapter two will attempt to frame and provide the necessary background relating to HMTs, trust, and relevant cybersecurity details as they fit into the model of this work. A large part of my effort is to explain the essential elements of trusting a learning and inference system. I will also provide the necessary related structure for how we and a system might come to different conclusions while looking at the same evidence.

In chapter three, I take the concepts laid out in the previous chapter and discuss an algorithmic model to calculate belief as agents evaluate evidence. I provide detail for some of the factors that perturbate belief divergence within the model and show examples for each.

Chapter four uses the model to represent behaviors we might see in humans and machines serving on the same HMT, evaluating similar evidence. I then expand on this and present a model for Stuxnet as discussed in the introductory section.

Finally, in chapter five, I conclude by discussing the results from the experiments and drawing inferences from the data. I also pose open questions and highlight several potentials for future research.

In the end, I hope that I will have described the current state of computing and cybersecurity sufficiently and to have tied in human-machine teams and the trust the underlies them. My point is to clarify the concern that adversaries will seek to exploit this trust in the future, and in ways that will be difficult to detect or defend against. Furthermore, in the aftermath of a successful trust attack, it may be difficult to "correct" computer systems entirely, including the data sources they rely upon for learning and inference, and re-establish trust between such systems and their human teammate counterparts.

Chapter 2

Concepts and Definitions

> *"Great teams consist of individuals who have learned to trust each other. Over time, they have discovered each other's strengths and weaknesses, enabling them to play as a coordinated whole."* [28]

2.1 Human-machine teams and AI systems

Throughout our history, humans have wrestled with creating organizational models that maximize the effectiveness of teams. In turn, teams often obsess over equipment and tooling, in search of advantages that serve the team. Those tasked with improving collaboration, coordination, and productivity today must integrate new paradigms of connectivity, computing power, and information access in the digital age. We might think of these digital components as equipment or tools as well. Any current discussion of team dynamics likely includes HMTs that pair humans and AI systems together in a complementary manner best to solve challenging problems.[29, 30] In a sense, the equipment has advanced so much, that it has evolved into a key player itself, and can be essential in helping the team to win.

Like teams, individuals are also becoming more dependent upon AI systems for daily decision-making.[31] These one-person HMTs maintain the same concerns as multi-person pairings with an AI system. HMTs are sometimes formally designated but are often not and are either assumed or evolved. They may or may not be voluntarily as well. Humans may come to depend upon an AI counterpart due to convenience, accuracy, or similar.[32]

In the spirit of the equipment analogy, computers augment human capabilities in many areas. Because of this, there has been and remains great optimism for AI systems to deliver wholesale improvement across many important domains of human interest.[33, 34] AI continues to grow into a role that assists in accumulating, making sense of, and summarizing data involved with increasingly information-driven decision-making. From AI-assisted driving automobiles to diagnosing rare diseases in healthcare, activities that depend upon a great deal of data and processing indeed seem well suited to HMTs. There are many commercial examples of AI systems that automate business processes and achieve significant performance improvements in those situations where humans and machines work together by complementing each other's strengths: The former's qualitative skills and the latter's quantitative

[28] Edmondson, *Teaming*.

[29] Stowers et al., "Improving Teamwork Competencies in Human-Machine Teams."

[30] Seeber et al., "Machines as Teammates."

[31] Elliott, *The Culture of AI*.

[32] Jarrahi, "Artificial Intelligence and the Future of Work."

[33] Fast and Horvitz, "Long-Term Trends in the Public Perception of Artificial Intelligence."

[34] Castro and New, "The Promise of Artificial Intelligence."

capabilities. Some things come naturally to us which remain difficult for machines, such as leadership, creativity, innovation, and other social abilities. Yet, the strength of computation, such as analyzing large amounts of disparate data, for example, remains impossible for humans to replicate. Effective HMTs enhance both kinds of capabilities and capitalize on maximally combining these skills.[35, 36]

There is a worrying concern with AI technology as the stability and safety of these systems have been the specific subject of much debate.[37, 38, 39, 40] Particularly in contexts where the AI has semi or fully autonomous decision-making capability, such as in AI-assisted driving automobiles, automated financial trading systems, and of course, in defense, military, and security systems,[41] there have been numerous examples of unexpected conclusions, surprising interactions, and unintended feedback loops that have led to system instability and unpredictable consequences.[42, 43, 44, 45, 46] Trust is a core component of the human-machine relationship, and this thesis will focus on how adversaries might specifically target and manipulate our trust in computer counterparts operating within a human-machine team.[47]

2.2.1 Traditional Cybersecurity Concerns

> *"Cyberdefense isn't magic. It's plumbing and wiring and pothole repair. It's dull, hard, and endless. The work is more maintenance crew than Navy SEAL Team 6. It's best suited for people who have a burning desire to keep people safe without any real need for glory beyond the joy of solving the next puzzle."*[48]

Computer and network security is a complex battle of wits between attackers who attempt to find vulnerabilities (and take advantage of them) and those who successfully defend a machine. Those on the offense have a significant advantage in that they need only find a single weakness, while the defender must eliminate all vulnerabilities to maintain perfect security. Cybersecurity begins with what I have

[35] Saenz, Revilla, and Simón, "Designing AI Systems With Human-Machine Teams."

[36] Barro and Davenport, "People and Machines."

[37] Danks, David, "How Adversarial Attacks Could Destabilize Military AI Systems - IEEE Spectrum."

[38] Amodei et al., "Concrete Problems in AI Safety."

[39] Guzman, "Making AI Safe for Humans."

[40] LaGrandeur, "How Safe Is Our Reliance on AI, and Should We Regulate It?"

[41] Morgan et al., "Military Applications of Artificial Intelligence."

[42] O'Kane, "Self-Driving Shuttle Crashed in Las Vegas Because Manual Controls Were Locked Away."

[43] Owen, "Face ID Attention Detection Security Defeated with Glasses and Tape."

[44] Snow, "Amazon's Face Recognition Falsely Matched 28 Members of Congress With Mugshots."

[45] McCausland, "Self-Driving Uber Car That Hit and Killed Woman Did Not Recognize That Pedestrians Jaywalk."

[46] Dockterman, "Robot Kills Man at Volkswagen Plant."

[47] There are many follow-on questions here that are beyond the scope of this thesis. For these trust-based attacks, how do we reestablish trust for such a compromised system? Can we return to previous levels of trust after a successful attack? Human-only teams have certainly failed in the past, but that has not stopped us from trusting human teams in new endeavors. Do these questions of trust generalize to all types of teams? That is to say, will (or should) human-machine teams be considered in this same spirit? If not, what is it about AI that has changed this dynamic? These questions will hopefully be addressed in future work.

[48] Wheeler, "In Cyberwar, There Are No Rules."

called *traditional cybersecurity concerns* that all manner of computers and networks are subject. Although this work focuses on a particular set of security worries unique to learning and inference systems, I hope that a discussion of traditional concerns provides valuable background on the entire cybersecurity domain, examples of how attackers might gain initial access to a learning and inference system, and potential motivations as to why an adversary might target these systems in the future.

Learning and inference systems are composed of general-purpose computers, and so they too share the fundamental worries of protecting a system's information from harm, theft, manipulation, and inappropriate use. Cybersecurity in a general sense is:

> *"A set of activities and other measures intended to protect—from attack, disruption, or other threats—computers, computer networks, related hardware and devices software and the information they contain and communicate, including software and data, as well as other elements of cyberspace."*[49]

Broadly speaking, a cybersecurity attack is malicious activity targeting computer systems or the users of such systems in order to gain access to those systems and the data they contain. Often, attackers look to capitalize on an exploit for financial gain. In other instances, the objective is to disrupt operations or, in relatively rare cases to directly damage or destroy physical equipment.

An adversary might employ many distinctive attack methods to carry out an exploit. Typically, these techniques break down into specific phases, from gaining initial access into a system to the steps taken to ensure that access continues long into the future. A deep understanding of these techniques is not essential to our purposes here, but what is relevant is that computers remain vulnerable to all sorts of abuse and manipulation, and incentives continue to exist for those that would attempt to take advantage of such exploits. If this has been true of all manner of computers in the past, we can rightly assume it will be true for the networked learning and inference systems of the future.

As it is, computers are making their way into all manner of devices. As we add smart capabilities to vehicles, thermostats, doorbells, locks, refrigerators, surveillance cameras, and similar, so too does the attack vector widen for potential adversaries. As evidenced by the Internet of Things (IoT) and the increasing provisioning of next-generation network connectivity technologies, each new IoT or "smart" device provides another opportunity to compromise and exploit a computer system. By their definition, IoT devices have access to the internet to talk to other devices and services. In other words, the IoT computer can talk to other computers, and in mostly an autonomous manner, without much human intervention.[50] None of this makes them any less susceptible to attack and manipulation. While these IoT devices directly contain a computer and potentially sensor system to read the temperature, audio queues, and the like, they also typically connect back to a centralized computer system where large data stores are collected, and decision-making typically occurs.

[49] Fischer, "Cybersecurity Issues and Challenges: In Brief (CRS Report No. R43831)."

[50] Miller, *The Internet of Things.*

As a result, there are many vectors to consider in the production deployment of any large-scale learning and inference system. For example, traditional cybersecurity attacks would be a concern across all subsystems, such as the original data sources and their data, as it travels through a pipeline of functions that clean and prepare the raw data. Also at risk are the models themselves, or the code that generates them, what we might call a decision-making core, and of course, any output delivery mechanism. These are all attackable vectors, and the measure of their vulnerability contributes directly to the safety and security of any given learning and inference system or subsystem.[51]

As we can imagine, the defense of an expansive computer network requires a great deal of effort, and that work can be arduous, serious, and challenging in supporting a system over its entire lifespan. The average network produces gigabytes or even terabytes of data every day—requiring the analysis of endless logs and alerts. Even with the explosive growth of the security industry, and a breadth of automation technologies available, adversaries are still winning. In November of 2020, Cybersecurity Ventures wrote that it expected,

> *"Global cybercrime costs to grow by 15 percent per year over the next five years, reaching $10.5 trillion USD annually by 2025, up from $3 trillion USD in 2015. This represents the greatest transfer of economic wealth in history."* [52]

There continue to be significant monetary incentives for adversaries to attack computing systems and challenges for defenders to secure such systems properly.

As we can see, the realm of traditional cybersecurity concerns is vast, complex, and arguably difficult to defend. Yet specifically for intelligent systems, additional manipulation methods relevant to this thesis stack on top of the worries outlined above and provide further significant challenges to the successful defense of such systems.

2.2.2 Cybersecurity Concerns for Learning and Inference Systems

Attacks that specifically target intelligent systems exacerbate overall security concerns—while being subject to more traditionally designed and motivated cybersecurity attacks, they are also susceptible to a separate class of poisoning attacks designed to destabilize or neutralize the system through the data ecosystem that it relies upon for decision-making.[53] The set of poisoning attacks on a learning and inference system is challenging to detect, trace, and mitigate. This class of adversarial attacks attempts to confuse or mislead learning and inference by misclassifying data or to fool such systems into identifying something in the data that is not actually present. For example, these attacks can use filters or modifications in that data to change a stop sign to a (high) speed limit sign or trick a system into thinking a tank is a bicycle. These attacks can target fully developed learning and inference systems or

[51] Schneier, "Attacking Machine Learning Systems."
[52] Morgan, "Cybercrime To Cost The World $10.5 Trillion Annually By 2025."
[53] Qiu et al., "Review of Artificial Intelligence Adversarial Attack and Defense Technologies."

those under development. Particularly, since AI systems rely heavily on training data to improve production results, we might consider a system subject to potential attack from its initial inception. Then, attackers can target the input of that training data in a non-production system or reverse-engineer the algorithms to understand the information that trained a system in the first place. Worst of all, the after-effects of these types of attacks are ultimately hard to understand:

> *The exact conditions for such attacks are typically quite unintuitive for humans, so it is difficult to predict when and where the attacks could occur. And even if we could estimate the likelihood of an adversarial attack, the exact response of the AI system can be difficult to predict as well, leading to further surprises and less stable, less safe military engagements and interactions. Even overall assessments of reliability are difficult in the face of adversarial attacks.* [54]

How we might quarantine, mitigate, and fix AI poisoning attacks is more complicated than a traditional cybersecurity issue. With the latter, security and other computer professionals work together to find the issue, correct it, and return normal service as quickly as possible. There is no assumption of a fundamental change in the machine's motivations or outputs, and quite often, the machine is replaced from a previous copy or rebuilt from scratch. If an adversary successfully compromises our website, we recover the server from the last-known good copy. While there may be difficulty in determining when the compromise occurred, what it may have influenced, and where to draw a good copy from, if we can at least begin to answer these questions, there is likely less consideration for whether we can trust the server going forward—once restored and patched, we likely assume the server functions precisely the same as it did before the compromise. If we trusted it before an intrusion, we could trust it again once it is returned to its original state.

In the same spirit that the calculator on our phone is not like a personal digital assistant, particularly over time, AI systems do not evolve like static file or web servers might. While it is true that a database, a file or directory can certainly be manipulated over time, it would be modified in a direct manner which could be subject to audits or logs, and these would show the nature of those changes and perhaps even who made them and when. For an AI system and its continual processing of large amounts of data ("big data"), such auditing or logging is often cost prohibitive or technically challenging.[55] Here, feeding manipulated data into such a system would be more difficult to track or detect.

For trust-focused attacks, the motivation and aim are different from all the previously discussed methods. An adversary may slowly seek to shift our normative expectations over a long period. We may attribute this to be the algorithms' simple adaptation to

[54] Danks, David, "How Adversarial Attacks Could Destabilize Military AI Systems - IEEE Spectrum."

[55] Tools such as MLFlow can be integrated into Jupyter Notebooks that enable tracking and reuse of models and how they change over time. So, while logging or auditing is certainly possible, like other computer systems, it is not inherent to ML systems, and must be proactively configured and maintained. At best, in a system that is not real time, the volume of data can be cost and technically challenging. At worst, for those systems that necessitate real time processing of data, that cost and technical consideration likely increases considerably.

the real world. The AI system's view of the world has changed, our view of the AI system has changed, and no human in this chain of events realizes what has happened. This erosion of trust should worry us because if we cannot trust an AI system, how do we reconcile its stability or safety concerns? For a compromise of trust, how might we go about correcting such a system?

2.3 Trust

We can say quite a lot about the nature and importance of trust within a human-machine team, probably enough to complete a separate thesis than what I hope to accomplish here. For our purposes, we just need to consider the trust that underlies HMTs in the sense that the AI system is acting as our teammate and what implications might follow from an AI playing that role. Our primary concerns are details that help explain how people can quickly come to trust a computer and how that trust can be the target of an adversarial cyberattack. Fundamentally, I argue that just as we come to trust other tools because of their importance in helping us solve complex tasks, we tend to do the same with computers.

Humans have used different types of tools to accomplish a wide array of tasks for a very long time,[56] and perhaps this success has led us to consider the effectiveness of such tools as a primary driver of whether to trust in their use or not. If we allow that a computer is a machine that is also a tool, in this spirit, perhaps we also come to trust a computer due to its repeated success at mechanical tasks. So, let us explore the notion of trust for computing and AI in this context of considering our employ of tools to complete tasks successfully.

Philosophically, there is typically agreement that trust, and trustworthiness are distinct notions—the first being the trustor's attitude towards a trustee and the latter being a property of the trustee. Ideally, that which we trust will also be trustworthy, and that which is trustworthy will be trusted. For trust to manifest in a relationship, parties within the relationship must have attitudes toward one another that enables trust to occur.

2.3.1 Motivating AI Trust

The following section provides several motivations for how we come to trust computer systems. These are general themes that position computers as machines, as tools, and that computers compare favorably with those other tools that have helped humankind to accomplish so much throughout our history. From these motivations we intuit that a reasonable state of trust along this path is perhaps an easy equilibrium to find oneself. Taking this further, the rise of more complex computer paradigms—particularly learning and inference systems—create a dichotomy where it is easy to see how we come to value the usefulness of these tools, but also why their complexity clouds our ability to both clearly define what benefit the tool provides, and how it does so.

[56] Skinner et al., "Human-like Hand Use in Australopithecus Africanus."

A simple yet relevant example for us to consider is the relationship between us and our mobile phone. There are several routes to establishing trust in the traditional sense[57, 58] and we should review the relevant ones for our purposes concerning our phones. More conventional software applications on our phones, including the calculator, evoke trust rooted in tool-based notions of trust. In this spirit, the calculator efficiently answers questions such as, what is the sum of two numbers? What is a reasonable tip on my current restaurant bill? How we have come to trust its answers will be an interesting comparison to how an HMT might trust an AI system. Note that I am referring to trust in a manner largely dependent upon particular goals and the context surrounding them. For example, we can trust the bank with our money while at the same time we may not trust the same institution in the care of our children.

To detail the tool-based trust between a human and a computer for the purposes of this thesis, I provide five motivations below that illustrate how one can come to trust tools, machines, and computer systems. Afterwards, in the section immediately following, we will detail how adversaries will seek to exploit trust motivated in this manner.

1. Reliance is often at the heart of trust within philosophy, and while there may be many other aspects to consider, many theories of trust begin here.[59] Reliance is that the object in question produces expected results for actions that we engage it to perform most of the time. In this sense, reliance is rooted in reliability—we come to trust behaviors that we can successfully predict—the more often the object matches our expectations, the higher reliance we might set upon it.[60] For example, we can trust the calculator's results because it has reliably provided the correct results repeatedly in the past. Further, regardless of the number of these past results for a particular problem, we are sure to get the same answer. We get precisely the same result for every situation given the same input. If we share the input for a problem, we can quickly reproduce the same answer. Lastly, this repeatability does not change over time, and so given the same input, an answer provided yesterday will still be the same tomorrow.

2. Reliance is built upon specific functions that a trustee is to perform. We know what functions are satisfied by the trustee—what it is supposed to accomplish when we implement it in a certain way in a particular scenario. In a sense, we engage with the trustee over a specific set of activities we believe it can and will perform successfully. This knowledge enables us to engage with the trustee appropriately. It also implies that we may not necessarily extend that trust to other unrelated functions. In the calculator example, we know what types of calculations it excels in performing and how to construct them. For instance, we know that we

[57] LaRosa and Danks, "Impacts on Trust of Healthcare AI."
[58] Roff and Danks, "'Trust but Verify.'"
[59] Goldberg, "Trust and Reliance."
[60] Lewicki, Tomlinson, and Gillespie, "Models of Interpersonal Trust Development."

might enter two numbers and ask for their sum or product, and we know how to formulate each of these functions explicitly. We also know that we cannot enter a text-based sentence and ask for a count of words, for the calculator has no entry for alphabetic text. We trust with a sense of specificity in what functions the trustee might successfully provide us in return. Whereas in the previous route, we rely on something to accomplish some critical task, this route is critical in connecting the best solutions to those efforts.

3. Taking this notion of trusting specificity further, we can assume a sort of specific contract for how we engage those functions successfully. By contract, we mean that whatever inputs and outputs are associated with a function and that there are some constraints upon their values. Now, by virtue of these limited inputs and outputs, there are clear roles established between the user of the calculator and that of the trustee. Here, we ask the calculator to perform some function in a bounded context of what actions it will perform—where its activities begin and end. For example, the calculator cannot act upon the answers it produces—it cannot take those answers a step further and do something in the physical world that impacts us. We do not expect past results to influence the input of new problems, and we do not expect further activity from the calculator beyond the exact problem we entered. So far, in the first route, we rely on something to accomplish a critical task, and route two connects the best solutions to a critical task. This route simply adds an optimum utilization for a particular solution in accomplishing said task. We may come to learn this by instruction or by trial and error, but that process builds trust as we see positive results via this optimum utilization.

4. Trust can also evolve from scenarios involving regulation, licensure, and certification. That is to say that we often apply a higher level of trustworthiness to those specially indicated products and services verified by some outside party. For example, while a calculator does not contain any of these guarantees in a *de facto* sense, by the manufacturer's inclusion of a calculator on their system, we get the same sense that its results should always match our expectations. We consider that for a major manufacturer to select this particular calculator and install it on so many phones must say quite a lot about its effectiveness.

5. Lastly, trust can be transitive, so if I trust you, and you trust the calculator, then I trust the calculator and its results as well. Note that this does not imply the values between two parties are a one-to-one match—that I place a high degree of trust in the calculator, and you trust me does not mean you necessarily trust the calculator as much or as little as I might. It merely opens a potential opportunity that could lead to the establishment of trust with further support from the previously listed motivations. For example, the makers of phone software often combine transitivity with certification simply by placing a default calculator within the operating system, thereby establishing it as trustworthy since it is included in every single phone manufactured under that brand. Since I have a phone in use by millions of other people, the software on that

> phone must be tried and tested, it must be trustworthy since millions of others have explicitly chosen to purchase and use that particular phone model.

There may be other motivations for our coming to trust something, but the examples listed here provide a reasonable connection to how we might have evaluated other tools in the past and come to trust in their use to solve some problem for which they are well suited. These examples also seem sound when applied to our increasing use of computers to aid in decision-making.

This work is not an exhaustive exploration of how we come to trust a computer or its decision-making ability. It is also not a normative explanation of whether this trust is or is not properly justified and under which conditions. Rather, I seek to simply provide some reasonable framework of what trust within an HMT is, how it might form, and how it can be subsequently broken. Considering this framework of trust in the simplest of terms will provide us with sufficient background to support the full argument of this thesis.

2.3.2 Static or Dynamic Trustees

Tools can be static or dynamic in nature, meaning the roles they play, or the contracts they fulfill may change over time. Generally speaking, shovels have not changed, in the roles they play or the contracts they fulfill, for a very long time. As a result, there is no change in the tool that causes us to reconsider how we have come to trust in its use, and our having some expectation that using a shovel is a good choice in having to dig a hole or move some dirt. Similarly, a powerful aspect of the calculator is that the contract it fulfills are *static* across users, time, and space. Some tools do change, however, regardless of if we should notice. Some computer systems, particularly learning and inference systems, do change—they gain new capabilities over time and expand the set of contracts they might fulfill as a result.

There are assumptions about some unchanging qualities within the trustee in all of the motivations discussed. A static notion of trust is to say that the trusted object does not change, and so we find no need to reevaluate it, since we have already done so in the past. Now there may be other factors that do cause us to rethink our trust in something—we may learn something new that effects that trust and causes us to reconsider. For example, if I have never asked my calculator what 2*5 is, but today I do that, and it tells me 27—I might come to stop trusting it even if I'm quite sure nothing about the calculator itself has changed.[61] Setting aside these other reasons for reevaluating trust in something, we focus here on static and dynamic trust as a function of computer applications and the contracts they fulfill—some never change the set of contracts they support, and others certainly do change, regardless of whether humans correctly distinguish between the two.

Applications that are static, like the calculator, always function the same, and our use of them yesterday does not impact our use of them today. The argument I make here

[61] With thanks to Kevin Zollman for the example.

is that in this sense, humans are used to thinking about computers as tools, and we trust computers because of their repeated success at computational tasks.

However, there are functions within that same computer that *are* influenced today by our past computations. Now parameters of the computer and, therefore, our relationship may have changed. We find we have calculated our trust in the computer based on some static state, so the natural question might be whether we can still trust the computer in our original manner now that the computer's function has somehow changed? This question refers to a sense of trust in a *dynamic* object, which for our purposes is the computer.

Notably, in cases of learning and therefore *dynamic* computer systems, perhaps it is a high opportunity cost in determining exactly what functions the software can provide and how it does so—where it is either not worth the time and effort or due to the configuration of the computer hardware or software that the effort cannot reasonably be pursued. Perhaps where either case is true, we tend to trust but not verify—we come to trust through the motivations discussed because a tool fits more than it does not. In this sense, we come to trust it without much evaluation of how we arrived and without a look behind the curtain of how that tool works. This characterization of default trust makes it a sort of basin of attraction, where we both tend to default to trust and tend to fit our initial view of some tool into the trust motivations. When we switch doctors, we likely walk into their office already trusting they will do their best to keep us healthy. Likewise, when we buy a new laptop, we trust it did not come pre-installed with malware.

Maybe here is where computers and more simple tools, like hammers or shovels, are not alike in the sense of a tool: We can see that a shovel has no hidden mechanisms (or "black boxes") as to how it moves dirt. Although we treat a computer as a tool, there can be a black box both to what questions it answers and how it comes to provide those answers.

If we suspect that adversarial attackers know this—that people are somewhat predisposed to fall into trusting a computer system, and that more complex systems exacerbate this phenomenon greatly, then trust is certainly exploitable with regards to the human aspect of an HMT. That is to say that because people tend to treat computers as a tool, and because they tend to provide value in solving complex problems, people come to trust these systems. However, this connection does not seem appropriate in the cases we discuss. Beyond *static* applications where data is input and decisions are made based on that input alone—say I enter your loan information and based on your debt-to-income ratio, your assets, your payment history, the system calculates your interest rate, we might come to justifiably trust such a system, because it is clear how calculations are made, and miscalculations can be attributed to bad input only. But this is not the case for learning and inference systems where we do not exactly know how a decision gets made or what data has been used to decide. Perhaps in such a system, now demographic information, our social media posts, our purchases are incorporated into the calculation. Maybe as the customer, we think our payment history is the only thing relevant here. Maybe as a loan agent, we are not quite sure what information is used in such a determination. In these situations, the human tendency to trust is ripe for manipulation, because we cannot verify how these decisions get made.

Our original example highlights a system capable of inferring about their operating environment. The tragic flaw is that inference about the environment is wrong—Stuxnet deliberately manipulates the industrial controls within the enrichment cascade by playing the role of the unreliable man-in-the-middle. The inferences (and subsequently, any learning established due to such inferences) are found to be wrong because humans can determine via other information that the computer system is providing incorrect data. The human teammate is critical here, as the computer system does not possess the capacity to determine this about itself.

The industrial controls affected by Stuxnet are continually weighing incoming evidence about the system's performance. If a control reads data that indicates a centrifuge spinning too fast, then perhaps it infers the system should slow that centrifuge to correct the readings. We will discuss this inference further in later chapters and formalize a model for studying how best to represent it within these types of systems. For now, we are simply considering the dynamics of incoming information within a computer system that makes inferences based upon this data. Also, this phenomenon is not a simple case of incorrect or altered data. Mis-keyed data or explicit deceptions have always existed within computing, but these alter the input only. This thesis focuses on specifically attempting to alter the internal decision-making structure of the inference system—the likelihood mechanism itself.

Hopefully, it is clear that AI is more than just a computer and further that it is often playing the complex role of a teammate within an HMT. If a tool is static and these motivations to trusting such a tool as I have outlined are static, we then arrive at a problem rather quickly, which is that in many cases, we have come to think of AI in this static context, where we do not distinguish the particulars laid out within these five motivations I have detailed. This makes it challenging to identify what it is—what aspect—that we trust within an AI system. We instead come to trust it in its entirety, where we do not distinguish how we engage with an AI, and how we evaluate the results that it provides.

Perhaps the key element within an AI system is that it provides emerging new capabilities due to previous events over time. This is borne out by not only individual functions getting better over time, as this is precisely the aim of many machine learning methods to improve success by measuring the effectiveness of previous decisions, but also that once a system is sufficiently good at one calculation, that may enable it to now make a different type of decision. For matters of trust, if we are not discriminating which functionality we have engaged, how do we calculate its reliability, or even how we properly engage it all? This sort of trust makes it a strangely quick-to-trust system and an easy target for subtle manipulation. So, while AI systems are subject to adversarial attacks of theft and damage, they are also targets for deception and subterfuge as well. Due to their dependence on data, their ability to learn from that data, and the inferences they make, attacks that can manipulate a learned likelihood function seems intuitive. One suspects that if these attacks could be performed without alerting human counterparts in an HMT would be particularly dangerous.

2.3.3 Asymmetrical Trust within a Human-machine team

Humans trust in static tools due to their repeated success at fulfilling a defined set of known contracts, but that trust can be misplaced for a dynamic, learning systems where the set of contracts is changing or is unclear. If HMTs place a learning, changing AI system as our teammate, we default to trusting it as we do other static computer systems, but perhaps our trust in a dynamic teammate should be grounded in something else.

In the interest of this thesis, I focus wholly on a human (as part of an HMT) having or not having trust in an AI system. In future work, it may be interesting to explore how a computer system might trust or not trust its human counterparts, but for now, the machine's trust in its human teammates is assumed throughout.

Note that this asymmetrical notion of trust still requires that human HMT members share a traditional view of trust in the spirit of Hawley[62], whereby the human is:

1. Vulnerable to the AI system (certainly including being vulnerable to betrayal);
2. Reliant upon that system's competency for what we trust it will do, and;
3. Reliant on its (assumed) willingness to perform that activity.

We should also note that in this thesis, the notion of trust is graded—we do not simply trust or not trust a computer; instead, our position changes over time. Our degree of trust changes as we evaluate evidence.[63]

I have been careful to characterize us as either having or not having trust in a trustee. Intuitively, mistrust is a spectrum, as I similarly argue for trust. Yet, the outcomes of having a great deal of mistrust in something would seem to point to perhaps different motivations and conclusions than we might from having the same degree of trust in that same thing. So, while we might exhibit a lack of trust, I am not directly concerned with characterizations of mistrust or distrust, as those seem different enough from the arguments here to warrant a more critical exploration beyond the bounds of this work.

Nevertheless, this framework for thinking about trust in relation to how we might trust a tool—or the answers from a calculator—does not seem to generalize to non-static applications. For intelligent systems, such as a digital assistant (DA), including Apple's Siri, Amazon's Alexa, or Google's "Hey Google," they break many of the assumptions necessary for the trust we have laid out, even though we often do not distinguish them from other static applications "on our phone." While the examples related to the calculator should clarify that it deals with only particular types of decisions, explicitly structured, Das provide a very different kind of decision-making. These assistants answer more general questions such as, "should I wear a coat today?" or "where is a good restaurant near me?" or "what is the greatest threat to democracy today?" or "what is the meaning of life?"

Here, a DA is not performing routine, repetitive tasks that we expect of our calculator and for which we often know or can quickly calculate the answer. Instead, we do not often even understand how her decision-making works, as is evidenced in the

[62] Hawley, "Trust, Distrust and Commitment."

[63] We assume this evaluated evidence comes from interactions that are both familiar and new experiences.

growing AI explainability literature—and yet increasingly, these digital assistants influence our decision-making at every level.

Now we might be tempted to characterize a DA as a tool instead of a teammate. Of course, teaming perceptions are increasing in the literature,[64] "as technology advances in both capability and interactive capacity."[65] We consider the DA a teammate most notably by the fact that we can ask a DA to answer a wide array of questions that might arise in that context wherever we might have our phone. A static tool, given the same input, provides the same results—perhaps across users, space, and time. Should that result be influenced by either space, time, or both—and where the technology assists in determining those inputs—the DA is acting in a teammate role. Therefore, static computer systems are tool-like, whereas dynamic systems are more like a teammate.

There are other reasons to argue for AI being a teammate. Team collaboration has been a key driver for new insights—human achievement is rarely accomplished alone. Today's digital technologies have greatly assisted in bringing teams together in new ways to innovate. In the future, these digital collaboration tools will do more than simply enhance team performance. At some point, the distinction between assisting the team and being a component upon which the team depends disappears, and the machine becomes an integral part of the team.[66] In addition, complex and subtle human traits, such as sympathy are explicitly being built into AI models.[67] The humanization of such AI systems will continue to blur the lines between helping the team to accomplish its goals, and simply "being a part of" an HMT.

Note in these examples that the questions we are asking, and the inputs required to answer such questions are not static. Instead, they depend upon both space and time contexts, and are perhaps influenced by previous questions that we have asked or the answers we have been provided. And so, in the spirit of these DA examples, any AI system playing the role of a teammate within an HMT has far more substantial trust implications than if it were simply some sort of tool. We cannot appeal to reliability to trust an AI system, for new questions and answers we might ask may have no predecessor. Likewise, we cannot assume we know the computational functions we are engaging or rely on knowledge of fixed inputs.

In conclusion then, the implications of AI as a teammate must force us to forgo a simple generalization of trust in the tradition of tool application and force a far more detailed view. This is to say that in situations where we trust we will get a specific answer or a certain *kind of answer*, we can no longer assume this will be the case. *What* we can trust, and *when* we can trust now seem far more complex than a more straightforward tool-derived sense of trust. Thus, a motivating question for us to ask is, if it is hard to know when to trust results at all, how might we model an AI system that begins to perform in a way that causes us to lose trust entirely?

[64] Ososky et al., "Building Appropriate Trust in Human-Robot Teams."

[65] Lyons et al., "Chapter 6 - Trust and Human-Machine Teaming."

[66] Malone, "How Human-Computer 'Superminds' Are Redefining the Future of Work."

[67] Wilson and Daugherty, "Collaborative Intelligence: Humans and AI Are Joining Forces."

Note that we need to take care to distinguish trust in learning and inference systems from trust in humans, since we also change over time. We are, hopefully, not tools in the general sense: We can be quite unreliable at times, and it may be unclear how to best engage with us to get things accomplished, depending upon the circumstances. Further, there is no indication for what our set of contracts might be or how to leverage them. Indeed, there seems a different set of motivations that we use in coming to trust fellow humans. While some of the examples covered within this thesis may generalize to be true for people as well as any learning and inference system, I suspect this does not damage the underlying argument that we should not ground our trust in something dynamic using the motivations outlined within this paper.[68]

2.4 Reevaluation, a human-like response to surprising new evidence

This thesis will highlight how we and a learning and inference system might come to different conclusions or states of belief in some hypothesis, even though we have both considered the same evidence. One way that makes this outcome possible is to look back at prior evidence and reconsider what that evidence's evaluation meant when it occurred initially. I will make extensive use of this concept that I have come to call *reevaluation*. The reevaluation process illustrates how we might reconsider a prior piece of evidence whereby some new evidence clarifies that past piece of evidence.

Interestingly, ML or AI systems are often low memory or memory-less due to the computational limits or storage costs of the vast amount of data these systems require. These systems typically evaluate extensive evidence in training to make accurate predictions about similar future evidence the model will encounter in production. Often, the results of this process are recompiled into the next iteration of the ML model. With each iteration, the model has made an increasing number of (hopefully) accurate predictions, so the model overall is more accurate. As such, however, these models cannot go back to reevaluate specific evidence, as it no longer exists, even if the developers of such systems wanted to develop such a capability.

Should a model look at an increasing number of pictures of dogs and begin to identify Irish Wolfhounds correctly, it will have potentially seen many Wolfhounds over time. Therefore, given a large enough number of Wolfhounds have been seen and given a new picture of a Wolfhound that resembles any of the previous ones, the model will assign a high likelihood that the new pup is indeed a Wolfhound.

However, what if we have trained on the wrong data all along? What if our training data images were tagged incorrectly, and our Wolfhounds are actually Deerhounds?

Human beings are flexible in reasoning about evidence and connecting it to a belief regarding a particular hypothesis. Machines (that are memory-less) cannot reconsider, while human *could* reconsider. Of course, they might not revise or reconsider on a

[68] There is interesting future work in further distinguishing the notions of trust I use in this paper and how they apply to humans and complex computer systems. For my reasoning here, if trust in a complex system can also apply to humans, I believe the argument still stands.

particular occasion, but they presumably have the capability to do so in a way that memory-less machines cannot.[69] This is particularly true when the rules of an engagement are changed, and in ways that a human would deem inconsequential.[70, 71, 72] For the above example, when we realize our error, we reconsider each image as a Deerhound and adjust our likelihoods for any new pictures appropriately. Generally speaking, when considering surprising new evidence, we can reevaluate prior evidence and change our minds with greater flexibility than a computer might be programmed to do. Computers and the controls that connect them to the physical world are limited by input in "range, field of view, torque, accuracy, and so forth. Thus, the agent has a restricted capacity to sense its environment, process the sensed data, and use that information to affect its environment."[73] This limit also applies to reconsidering input in light of some new evidence. On the other hand, humans might recall significant specific instances or sets of past evidence and change our minds about what it means. In basic terms, our beliefs are driven mainly by our evaluations of the evidence that we see.

Comparatively, this reconsideration of prior evidence is largely absent from many machine learning models that underlie current AI systems. As a result, should our AI encounter some surprising bit of evidence that sheds new light on past evidence, it could not reevaluate that past evidence in the same manner as our examples. Instead, such a system would simply update likelihoods going forward to compensate for the new revelation.

Specifically, reevaluation or "looking-back" is to reconsider and change the meaning of previous evidence considering the receipt of some new evidence. It is to take what we knew then and update it with what we know now. It captures situations where we learn more about specific past pieces of evidence through new information, and we change our valuation of both what that previous evidence meant when we received it and what it means currently. Reevaluation in the colloquial includes statements such as, "if I knew," "had I known," "what was unclear then is now clear," or "after reconsidering, I now believe." These declarations indicate that we would perhaps have come to a divergent conclusion or acted in some other manner if we had interpreted past evidence differently.

We process new and perhaps surprising evidence in many scenarios: When we are learning something new, when we come to find that we are incorrect about some previous decision, or when the given evidence is ambiguous, and so on. Often, all these scenarios play into a single belief that we might have.

[69] An objection here could be that humans often err, frequently lie, and often are resistant to correcting belief after learning that evidence was wrong. I suspect this is a range—that some are better than others, and some are truly bad at either or all these claims. Yet, if we allow for some human capability to do this, the argument still stands, because AI systems do not account for any of these reevaluations currently. For more, see Ecker, et al., DePaulo et al. and others relating to *source credibility*.

[70] Irpan et al., "Off-Policy Evaluation via Off-Policy Classification."

[71] Raghu et al., "Can Deep Reinforcement Learning Solve Erdos-Selfridge-Spencer Games?"

[72] Zhang et al., "A Study on Overfitting in Deep Reinforcement Learning."

[73] Musliner et al., "The Challenges of Real-Time AI."

I propose there are three contexts for which to consider reevaluation. There may be others, but these are relevant for distinguishing how we as humans might come to believe in some hypothesis, particularly considering ambiguous evidence. For each of these, I will use an example relating to industrial machinery:

- For our first context, say that we are excited about a new piece of equipment that we have just received, so we start it up to see how it operates. After a moment, it makes a curious noise, but given that it is brand new, we do not think much about it. However, shortly after that, it fails and stops functioning. We read the manual and found that given any noises on startup, we should have shut the equipment down immediately and further investigated the noise source—Oops.

  The above example is a simple case of being wrong. Here, we had more confidence in the new equipment over the likelihood of failure given a curious noise. Later we learn that is not the case. As a result, we update our likelihood for that evidence and any other evidence that resembles this scenario to match our new (higher) likelihood.[74]

  As another example, suppose we are at a party, and we notice the seemingly brilliant guest whose insightful statements have our group abuzz. We might note these instances but not draw any conclusions about the guest or their behavior since they are otherwise unknown. However, should we find out later in the national news that they have gone on to win a Nobel Prize, we think back to our experiences at the party, and we update what we noted previously considering this new evidence. Do we find the guest's past behavior following how we might expect a Nobel Prize winner to carry themselves? Initially, we process the winner's statements with a great deal of ambiguity since we did not have any context to evaluate them. Still, they are made plain by later facts, so we can reevaluate them differently than we had initially.

  This context would also cover cases where we might entirely change our minds about a piece of evidence.

- For our second case, consider that the longer equipment is in use, the more likely it requires maintenance and servicing to ensure its smooth operation. This case might include changing appropriate belts, oiling specific moving parts, and similar. As we see our industrial machines used over time, we increase the likelihood of noise given a maintenance problem. We adjust these likelihoods as we observe the machinery in use, so the longer a machine runs, the more we increase this possibility.

  This example represents that the world changes over time, and so, as a result, we change our likelihoods as we observe this change in the world. This

[74] That is to say that we initially had one very low likelihood P(failure | noise) and then shifted to a different very high likelihood P(failure | noise).

different likelihood impacts our posterior probability, which updates each time we observe the presence or absence of noise.

- Lastly, likelihoods are subject to manipulation without our knowledge. We might interpret evidence to make some hypothesis very likely, but where that hypothesis does not come to be. In the example of Stuxnet, we may observe both the world and our system's view of the world as operating normally and be unable to understand why we see centrifuges failing.

  Here, the Iranian scientists had one number for the likelihoods,[75] when the reality was quite different due to Stuxnet's manipulations. In particular, the likelihood of normal readings while a centrifuge was needing maintenance, turned out to be much higher than any scientist might have believed.

  We might refer to this context as, "we have been deceived."

Hopefully, these examples will be the essential background for the next chapter, where I frame a Bayesian model that will enable us to represent situations such as those discussed here. By formalizing the evaluation of evidence over time, I will mathematically show critical differences between humans and AI systems coming to believe in a hypothesis.

As we have seen in section 2.2.2, AI systems are subject to adversarial attacks beyond normal deception and subterfuge in that these attacks essentially change the calculated likelihood functions without informing their human counterparts. A poisoning attack in the guise of a carefully placed piece of tape might fool a surveillance system to identify a turtle as a rifle[76], but we will not be so easily misled, *"…particularly since humans are typically unaffected by these attacks. We can easily recognize that a turtle is not a rifle even with random noise, we view tape on a stop sign as an annoyance rather than something that disrupts our ability to follow the rules of the road.*"[77] Later, the model I propose and formalize will show the difference in likelihoods that we and an intelligent system set for the same piece of evidence—either the turtle or the rifle. The manipulation in this model will be that the AI system concludes a high chance that what it sees is, in fact, a turtle. The human counterpart will conclude, rather opposingly, that the likelihood for the evidence being a turtle is relatively low, while the likelihood for the evidence being a rifle is quite high. The exciting bit for us will be how the manipulated AI system has come into a different likelihood, and that it presented no additional indications of that manipulation. As part of normal operations, the system came to clearly see a turtle with no further information, and so, any human associated with that HMT would have no knowledge that its teammate indeed just saw a turtle.

This class of problem may be a simple limitation of current intelligent system design, or it may be that we are better at reasoning about edge cases—where the evidence does not quite match the typical manner that we expect to see, like the annoying piece of tape on a stop sign. For our purposes, perhaps it matters not. In these

[75] Namely, P(normal readings | centrifuge is good) and P(normal readings | centrifuge is bad)

[76] Athalye et al., "Synthesizing Robust Adversarial Examples."

[77] Danks, David, "How Adversarial Attacks Could Destabilize Military AI Systems - IEEE Spectrum."

manipulation cases, a straightforward way to show how an AI system and a human diverge on likelihoods is to have them reevaluate prior evidence differently. There are straightforward points where the two would not agree on a proposition, even when they share the same previous evidence from the start. This difference of opinion in what a proposition means potentially breaks down the level of trust within a human-machine team.

Humans are better at unstructured problem-solving and solving problems in which the rules do not currently exist. We ought to want the ability for AI systems to exhibit human-like flexibility in response to surprising evidence. While this type of programming seems currently challenging for computer systems, perhaps particularly for AI systems, it would also seem that surprising evidence happens quite often.

Ambiguity is related to reevaluation in the sense that evidence that bears multiple legitimate interpretations. It could be that the agent does not quite know how to interpret the evidence considering what they know or do not know. Alternatively, it might be that they do not possess the context necessary to perform a more precise evaluation. We will leave aside the notion of ambiguity where we are unsure whether the evidence in question supports a hypothesis.

Reevaluation seems particularly plausible should our prior evidence be entirely ambiguous. Perhaps we are just coming to know a new subject. So, in the beginning, we are not sure what to make of new information as we evaluate it. We do not know enough about the topic to evaluate incoming evidence confidently. As we learn more, we can go back and reevaluate based on what we now know.

If we accept that reevaluation exists, we will see divergent curves of belief from those that do not incorporate its effects.

Suppose we accept that this is a reasonable model of reevaluation. In that case, we also must admit that some reevaluation curves will be potentially radically different than those curves that do not look back, even with an agent having the slightest preference for a hypothesis.

2.5 Conclusion

Today's world is a faster, more interdependent world than ever before. Decisions are highly dependent upon supporting data and made with increasing speed. The pace of operations within the military, healthcare, finance, and similar domains threatens to overwhelm any conventional, top-down approach that may have worked in the past. The evolving solution has been to pair humans with computers in a manner that plays to each's strengths, but increasingly, it seems that learning and inference systems, particularly prediction-making AI systems, have been taking on more and more responsibilities previously held by humans. This solution has spilled over into consumer lives as well, with the rise of personal digital assistants on our phones, to the automation of the more mundane aspects of our busy 21st-century lifestyles.

Nevertheless, longstanding traditional cybersecurity concerns have not improved in the same timeframe. In many respects, the defense of critical computer systems has become more difficult, with the rising complexity in computing and software applications, and the introduction of learning and inference systems across the board has only exacerbated security concerns. Here, the dependence on data—often from other and disparate networks and systems—further complicates the computing security landscape by introducing new vectors and opportunities for adversaries up to, and including, sophisticated nation-state attackers.

As the decision-making power shifts from the considering human to the automating computer, data comes to eminence—for all the considerations we might take in deciding, the computer replicates these to come to the same conclusion in a mere fraction of the time.

However, our inference methods remain quite different from today's computers. Particularly considering past ambiguity and surprising new evidence, humans may reconsider the likelihood of evidence and change their minds. Previously, they may have been wrong, or the world may have changed, or some other actor may have manipulated their initial likelihood, but their manner of correcting any of these cases via reevaluation seems decidedly different than what a computer can perform today. And so, humans and computer systems can come to very different conclusions about the very same set of evidence.

For learning and inference systems, these possible results seem particularly problematic. Teamwork depends upon trust, as "Purpose affirms trust, trust affirms purpose, and together they forge individuals into a working team."[78] Working with teammates in a low-trust scenario is surely less productive and effective than the alternative. While challenges exist in building trust with teams composed entirely of humans, that team can still leverage the significant powers of resolving trust conflicts with a conversation that leads to understanding. Due to the constrained nature of computing input and output and the elusiveness of general AI systems, HMTs cannot depend upon these same conflict resolution strategies to resolve similar trust issues. Of course, if we cannot trust what a computer is telling us, what follows? Likewise, what follows if we cannot trust that a computer performed something that we programmed it to execute?

We should expect adversarial entities to note this problem and attempt to take advantage of it in the future. We could construct examples by those affected, where an attack could target large groups of people to maximize effect. For the former, there are several issues recently where *disinformation* has played a significant part of American disagreement, including political election cycles and the COVID-19 virus.[79] While there are several factors contributing to the resulting polarization, efforts such as those highlighted within Project Lakhta might exacerbate disagreement beyond what we might normally expect.

We could also attack smaller, more targeted, but influencing groups. Recall the Stuxnet example and consider who would be directly affected by such an attack. Surely the team directly involved at the Natanz facility was perhaps in the hundreds or thousands of people, but the downstream implications for a sensitive nuclear

[78] McChrystal et al., *Team of Teams*.

[79] There is a growing literature on disinformation campaigns, see authors Kate Starbird and Ethan Zuckerman.

operation were so important, that through a relatively small number of people, a targeted attack was able to halt the progress of the entire Iranian nation. Stuxnet will serve as our example in the model, which follows in the next chapter.

Chapter 3

Formalizing a Model

I will use the concepts discussed in the preceding chapters to formalize a computational model for how agents—humans and machines as part of an HMT—come to hold a particular belief.[80] We use this model to reason about how the two might agree or differ in evaluating similar evidence, how they may come to disagree upon the likelihood of a particular piece of evidence, and ultimately how an adversary could deliberately exploit these dynamics in order to erode human trust in an AI system.

Specifically, we will use Stuxnet to illustrate an attack where the human and AI come to divergent beliefs. We choose this example, because of its simplicity in there only being two states from which to choose—a centrifuge needs maintenance (hypothesis H_1), or it does not (hypothesis H_2), and that even in this simplified model, there are many ways to arrive at either of these beliefs, or even to change from one belief to the other. In normal due course, both the human and AI should conclude that after enough time, a centrifuge would need maintenance, as a result of normal wear and tear (and therefore converge on H_1). The attack here is manipulating the AI to soften coming to H_1, so that what a human believes is stronger than what the AI does. This manipulation would not be by any one sudden change that might get noticed, but rather by slightly changing several likelihoods so that over some sufficient number of them, the AI is less close to H_1 than the human.

Because we build trust within an HMT largely upon viewing AI as a tool that we interact with in a repeatedly reliable manner, erosion will result from a difference in belief between a human and an AI system. A divergence of what we expect and what the AI provides undermines reliability. Essential to such an attack are representations of how we as humans come to believe something, how an AI system might similarly do so, and most importantly, how the two might differ in evaluating evidence. The model will extensively use the reevaluation concept to show divergence in belief. There may be other methods to show similar results, but this approach and model serve as an example of human flexibility that computer systems—intelligent or otherwise—do not currently possess.

The assumption is that if a human and an AI system come to radically different beliefs over similar evidence, the former would begin to question the results provided by the latter. This divergence breaks trust motivations built upon reliance, as we can no

[80] All code and supporting materials for this thesis can be found at: https://github.com/dustinupdyke/epistemic_momentum

longer dependably expect similar conclusions from the AI that we ourselves would determine on the same evidence.

Two important notes on evidence: First, I often refer to *similar evidence* throughout this thesis, meaning that for some event, there are many ways to interpret meaning from it—we might observe that a plane's nose is in an upward position by sight. A sensor might come to a similar conclusion that a plane's nose is up based on sensor readings from a gyroscope or similar instrument. For the purposes of this thesis, our "by sight" capability and the gyroscope reading are both concerned with the same evidence.

Secondly, in regard to reevaluation, let references to *surprising evidence* within this thesis be some evidence (surprising, suspect, unexpected evidence) the agent observes that causes them to reconsider their position—to re-evaluate their past likelihoods on that evidence wherever it previously occurred.

3.1 The Evaluation of a Single Piece of Evidence

This section will show how we evaluate a single piece of evidence and how it fits into our larger belief. Belief will be the entire set of evidence evaluations we have for or against a particular hypothesis. Starting at the smallest component of the model, we receive a single piece of evidence for a hypothesis—and in the real world, we might ask, what does it mean? In our model, we fix this as either for hypothesis H_1 or for hypothesis H_2, meaning that values sum to 1.0. This is to say that we are in favor of H_1, or H_2, or indifferent to either (0.50/0.50). As we learn about something, we believe, "as the evidence suggests," meaning we believe a centrifuge either needs maintenance or it does not.

At the core of this calculation is the standard Bayes Theorem—a mathematical model for calculating conditional probabilities, which are the possibility of a hypothesis H_1 transpiring due to its association with another hypothesis, H_2. There are three key terms within Bayes we should expand upon and understand:

1. The *prior* is an initial probability obtained initially before considering any additional evidence evaluations relating to a hypothesis for a new event. This is the best rational assessment of the probability of an outcome based on all previous evidence. For our purposes, when we have no information about a hypothesis, and we begin to come to some belief about its likelihood, we simply use half for H_1 and half for H_2 to show that we have no preference for one over the other.

2. *Likelihoods* are effectively a bridge from the prior to the posterior, and are a conditional probability in the form $P(E_j \mid H_i)$, or "the probability of evidence E_j given that H_i is true." For the purposes of this thesis, the conditioning set is fixed to be the same, so that the probabilities will sum to 1.0, or,

   $$P(E_1 \mid H_i) + P(E_2 \mid H_i) + \ldots = 1.0$$

   Here we see the same hypothesis H_i for each term, and that all evidence E_j are assumed to be exhaustive and mutually exclusive. Outside of this thesis, it does not have to be the case that the sum of probabilities is 1.0, should we vary the conditioning set. As an example, $P(E_j \mid H_1) + P(E_j \mid H_2)$—note the mix of two distinct hypotheses—does not have to equal 1.0.

I force the two likelihoods to sum to 1.0 in order to simplify the model and to be able to talk about belief in a hypothesis (or not) in the clearest manner. This construction is greatly simplified from how I might track two seemingly opposite hypotheses in the real world. I might evaluate a set of evidence for each and conclude both are unlikely or highly likely. There would be nothing that enforces likelihoods for such hypothesis to offset one another, as if likelihoods were percentages of some fixed size pie. It is my hope that for our purposes, fixing the pie enables us to reason more clearly about how people come to believe some hypothesis or not.

Within this thesis, I often refer to likelihoods with regards to comparing H_1 to H_2 or H_1/H_2—an example might be .51/.49, indicating a slight preference for H_1. I fix these likelihoods to sum to 1.0, although likelihoods, like probabilities, for any two hypotheses, do not have to sum to 1.0, and we could easily assign any value to a likelihood for each hypothesis. Although Bayesian hypotheses are distinct, I fix them here. I pair each hypothesis presented within this thesis with its direct counter hypothesis (P and not P, for example, or "a centrifuge needs maintenance, or it does not"), and as a function of simplifying the notion of comparing agent's preference for a belief over another, the likelihoods for H_1 and H_2 must equal 1.0 here. A likelihood of 0.5 is the only value that indicates indifference here, although in traditional Bayesian representations, an agent might be indifferent between two hypotheses with any equal likelihood values—0.1/0.1, 0.5/0.5, 0.85/0.85—in this model, only for likelihoods of 0.5/0.5 is an agent indifferent in evaluating H_1/H_2.

A criticism of my fixing hypotheses and their sum of likelihoods might be that feeling strongly that a centrifuge needs maintenance is a different state of belief from feeling strongly that it does not. If we separate the two hypotheses entirely, the agent must separately evaluate evidence that supports either of the two, and in this model, perhaps that indifference should look more like .00/.00. If we find evidence for one hypothesis, that value grows independently from the other, say .00/.25. In our case, if the agent is indifferent to some evidence, we represent them as .50/.50. Because tracking hypotheses independent of one another greatly complicates the math and the model, I hope this simplification endures. The relevant question on the shop floor is whether this centrifuge should be pulled for maintenance. In the same manner, for all their sophistication, learning and inference systems are likely to answer this question in a similarly straightforward fashion, with a "yes" or "no." I follow this spirit here.

3. The *posterior* is a revised probability by including new additional evidence, and a *posterior* probability essentially combines (1) and (2). The posterior denotes our belief for H_i after evaluating a piece of evidence.

I use Bayes here in the form, *the probability P of some hypothesis H, given evidence E*, with H_1 and H_2 representing two hypotheses as:

Posterior_H$_1$ = (Likelihood_H$_1$ * Prior_H$_1$) / ((Likelihood_H$_1$ * Prior_H$_1$) + (Likelihood_H$_2$ * Prior_H$_2$))

Posterior_H$_2$ = (Likelihood_H$_2$ * Prior_H$_2$) / ((Likelihood_H$_2$ * Prior_H$_2$) + (Likelihood_H$_1$ * Prior_H$_1$))

The likelihood is then:[81]

$$P(E \mid H_i)$$

This equation enables us to formalize statements such as, "The likelihood of a noise indicating a problem (E), given the normal (H$_1$) or abnormal (H$_2$) function of a piece of industrial machinery (such as a centrifuge)".

3.1.1 Tying the Model to Machine Learning Today

I have also framed this model to reasonably account for approaches to probabilistic machine learning also based on Bayes theorem, where a model's parameters are treated as random variables. In this Bayesian set, parameter estimations amount to the computing of posterior distributions for these variables based on observations or observational data. Bayes' conditional probabilities enable us to reason about an agent's belief and its subsequent updating in light of new evidence over time. These estimations—or likelihoods—are the calculated validity of a given proposition, or how likely they are to be true. In addition, many modern ML techniques rely on Bayes' theorem, such as our Bayesian Learner mentioned previously, and Bayes' use within our belief model is a natural extension of what we find humans depending on within HMTs. The theorem spans broad applicability in ML and enables formalized reasoning about belief, making it related to our model here.

Another allied strength of Bayes relevant to the model presented here is that within ML, classification remains a significant task in information discovery from mining large data sets, and the simple naive Bayes classifier has proven surprisingly effective in this endeavor.[82, 83] Modeling a complex set of beliefs for an agent would be keenly challenging to establish the most relevant hypotheses and match empirical evidence accordingly. While in this respect, the model will be significantly simplified, there is also an expectation that the use of Bayes will continue to be the optimal method to learn from realistic, yet noisy data.

Finally, the use of Bayes will enable us to compare our original questions regarding the impact of reevaluation and how it is different from a forward-only calculation. Strictly speaking, Bayes theorem has no formal mechanism for changing the likelihood of past evidence;[84] instead, it only provides for updating new likelihoods going forward. In the previous chapter, we discussed several motivations for human

[81] The related posterior would simplify to P(H$_i$ | E).

[82] Schlimmer and Granger, "Incremental Learning from Noisy Data."

[83] Yang et al., "Learning Naive Bayes Classifier from Noisy Data."

[84] Hierarchical Bayesian models begin to get at this shortcoming but are beyond the scope of this thesis.

reevaluation, and how AI systems—and Bayesian learners in particular—cannot currently emulate this phenomenon, due to technological and cost constraints. So, our model here will emulate Bayes directly for purposes of modeling a baseline of coming to belief, but we will have to abandon some of Bayes constraints to represent our notion of reevaluation, where a human agent will update the likelihood of past evidence, and the computer will not.

The other reason to use Bayes is to provide a reasonable and rational process and outcomes for agent evaluations. In the model here, at no point will agents engage in irrational choice in the receipt of new evidence, nor will they reevaluate prior confirmations without attempting to remain rational. Although our evidence may be ambiguous, our agents will always do the best they can with what they know at the time of any decision.

Therefore, the smallest encapsulation within the model will evaluate a single piece of evidence. The model will track each evaluation of a piece of evidence within an arbitrary set, including all the associated factors of any single evaluation such as priors, likelihoods, and resulting posteriors—the three critical pieces of data for our calculations which we discussed above, all being relevant to a single evaluation within the longer-term representation for tracking belief in a hypothesis.

3.2 Connecting Evidence to Belief

A single piece of evidence tells us quite a lot about how an agent aligns that evidence with some belief due to the agent's setting the likelihood for that evidence to some value. This is intuitive since if we have seen quite a lot of similar evidence in the past, we expect the posterior to quickly increase as we see more comparable evidence going forward.

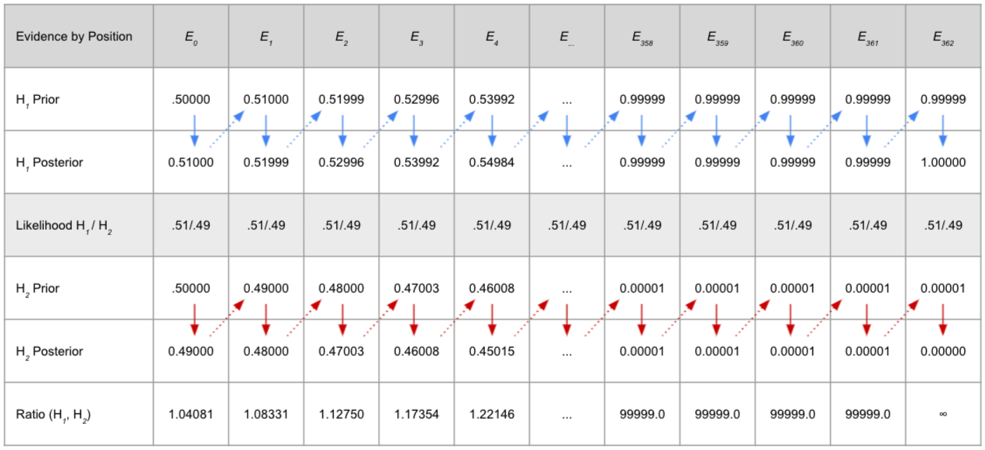
What a single evaluation does not tell us—within the scope of what an agent believes—is how they arrived there, and how that belief transformed over time. So, we will want to be able to represent the evaluation of sets of evidence for a particular hypothesis—a belief—as it changes. This is to say that we will track the posterior probabilities over time for the hypotheses in question—constructed as H_1 and H_2, with the latter being the negation of the former—we could also effectively think of these two hypotheses as (P) and (Not P). These evaluations of evidence are subject to many circumstances we see in reality: An agent has incomplete information about the evidence (but learns more as time progresses), or the agent simply changes their mind about that evidence—and all that resembles it—entirely. As a result, we want a flexible yet powerful model to render different manners for coming to some belief and showing the rising and waning strengths of confidence. In the model, I represent these sets of evidence as linear pieces as $\varepsilon = \{E_0, E_1, E_2, E_n\}$.
Since time is involved, and because we will want to interrogate the model over a certain period, we will need a transactional history of all evaluations, retaining all the possible inputs at time *t*, plus their respective outputs. This transactional history will be vital as we introduce reevaluation for any evidence since we will want it to span time *t*.

By storing all this information, we can calculate the agent's overall belief concerning the set at any point in time. Furthermore, we will use the curve of evidence evaluations to reason about how an agent has come to believe a hypothesis over any other or how they might have changed their minds. Lastly, we can directly compare results incorporating different perturbations that I will discuss below and those that do not.

3.3 Fixed Likelihood Iterative Bayesian Example

Our first example is for an agent to see the same evidence many times in a row, where the likelihood is slightly for H_1. Even before we see a single result, we might suspect that given even the slimmest of preference for one hypothesis, it should not take long to arrive at a firm belief should our agent see the same evidence repeatedly.

It turns out that our belief in H_1 gets increasingly close to a probability of 1.0[85] after looking at just 362 pieces of evidence. In figure *3.3a*, we see a basic table representation of an agent evaluating, $\varepsilon = \{E_0, E_1, E_2, \ldots, E_{362}\}$:



Evidence by Position	E_0	E_1	E_2	E_3	E_4	$E_{...}$	E_{358}	E_{359}	E_{360}	E_{361}	E_{362}
H_1 Prior	.50000	0.51000	0.51999	0.52996	0.53992	...	0.99999	0.99999	0.99999	0.99999	0.99999
H_1 Posterior	0.51000	0.51999	0.52996	0.53992	0.54984	...	0.99999	0.99999	0.99999	0.99999	1.00000
Likelihood H_1 / H_2	.51/.49	.51/.49	.51/.49	.51/.49	.51/.49	.51/.49	.51/.49	.51/.49	.51/.49	.51/.49	.51/.49
H_2 Prior	.50000	0.49000	0.48000	0.47003	0.46008	...	0.00001	0.00001	0.00001	0.00001	0.00001
H_2 Posterior	0.49000	0.48000	0.47003	0.46008	0.45015	...	0.00001	0.00001	0.00001	0.00001	0.00000
Ratio (H_1, H_2)	1.04081	1.08331	1.12750	1.17354	1.22146	...	99999.0	99999.0	99999.0	99999.0	∞

Figure 3.3a: Information relevant to a transactional history of iterative evidence evaluations

For each evidence in the set, we see a sequential calculation, where the posterior of an evaluation of evidence becomes the prior for the next. This is reasonable since our prior is a summary probability for H_1 based on all we currently know about that hypothesis before evaluating this new evidence. The posterior is the resulting combination of the prior and the evaluation of the new evidence. Thus, the posterior for H_1 and H_2 after updating on E_0 becomes the prior of E_1, and the posterior of E_1 becomes the prior of E_2, and so on. I indicate this transferal of posterior to prior with the blue and red dotted arrows and indicate the resulting calculation, which includes

[85] The implications of a belief in H_i of 1.0 within a Bayesian system typically are understood to be an agent's absolute certainty in a hypothesis, where no possible worlds exist where H_i fails to occur, and where the agent would never rationally change their mind regarding the truth of H_i. It is not my intent to follow this representation here, but rather to use 0.0 and 1.0 as a shorthand indicating a truncation of the floating-point precision of a probability representing arbitrary closeness to an absolute belief. We might say that 1.0 represents a state in the agent where they do not reasonably entertain the possibility of H_i not being true, although the agent could certainly envision a world where that comes to pass. An example of my desired representation is that we typically do not entertain the possibility of being hit by an asteroid as we go about the business of our day, and yet that chance, however infinitesimal, certainly exists.

the likelihood, as detailed above is the solid dotted arrows. Given a series of related evidence for the same hypothesis, this seems intuitive to represent the accrual of evidence for coming to hold some belief.

Now we take the raw data from *3.3a* and graph each posterior calculation across all evidence in the set, creating a curve representing an agent's belief over time. In this figure *3.3b*, the Y-axis pairs the probability of two competing hypotheses, H_1 and H_2.

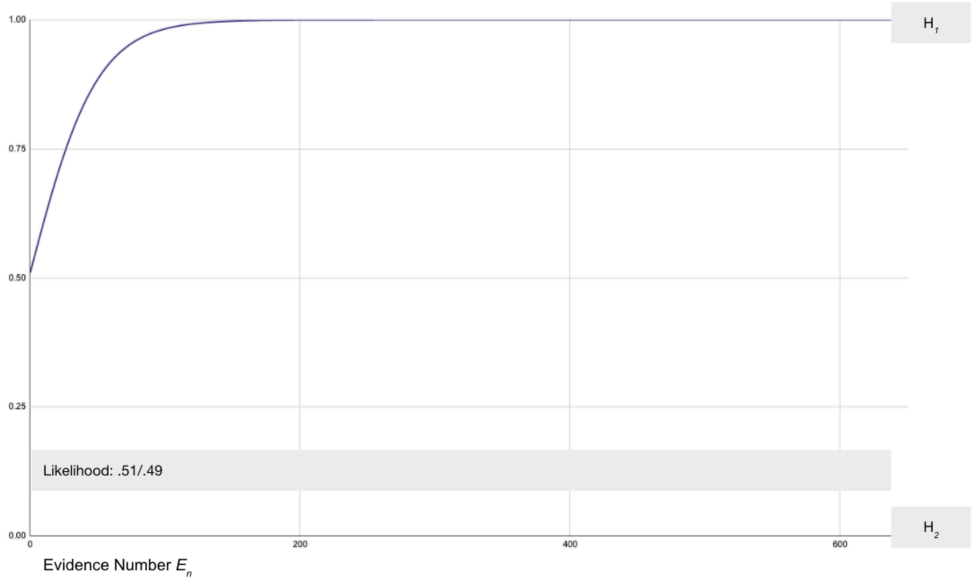


Figure 3.3b: Graphing the belief curve of an agent's evidence evaluations

Given that the curve favors either hypothesis, we say that an agent believes in the probability of that one over the other, based on the evidence they have seen and evaluated.

Later in Chapter 4, each experiment provides this curve for reference purposes, and we refer to it as *iterative Bayes* or simply *Bayes* in those examples.

3.4 Variable Likelihood Iterative Bayesian Example

Consider our AI system as a machine learning implementation of a Bayesian learner coming to believe in some hypothesis, either H_1 or H_2. The posterior of an evaluation becomes the prior for the next, given a set of similar evidence *E*. Of course, just as we change our minds based on evidence, an AI system may also change its likelihood for similar evidence over time. This may result from a change in the world that the AI observes and adapts to, or new surprising evidence not previously seen.

For figure *3.4a*, before we reach certainty, say at E_{200}, we flip the likelihood for H_1/H_2 from .51/.49. to .49/.51. Perhaps this is considering the evaluation of uncertain evidence or that there is a high degree of ambiguity in the agent's deliberation, but suffice to say they change their minds, if ever so slightly towards H_2 as being more likely. If we perform the same calculations, but with a change of the likelihood at E_{200}, we get a more interesting curve:

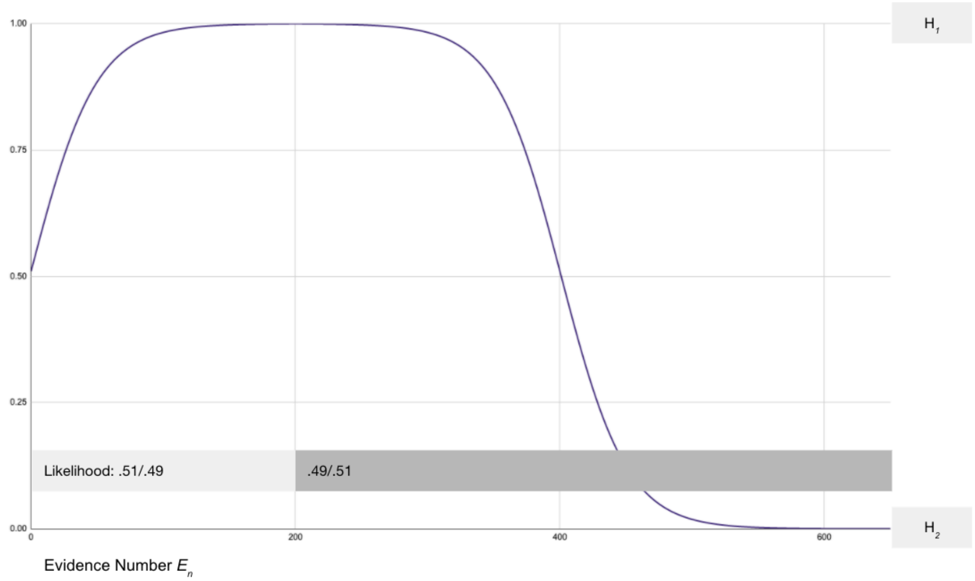


Figure 3.4a: Changing belief curve for evidence where the likelihood has changed after 200 iterations

In figure *3.4b*, the agent changes their mind several times, with variable weight, and in a shorter period. Note that our agent trends toward H_1 like what we saw previously. However, at E_{50}, their likelihoods switch from H_1 to H_2 in a subtle way of .49/.51, and as a result, we see the fall towards H_2. The rise towards H_1 between E_0 and E_{50} and fall to H_2 from E_{51} to E_{99} mirror one another over their respective sets of evidence. Should we pick two points positioned similarly in the array, such as E_{24} and E_{74}, we see that their posteriors are identical, although we might say that they are rising and falling overall, respectively.

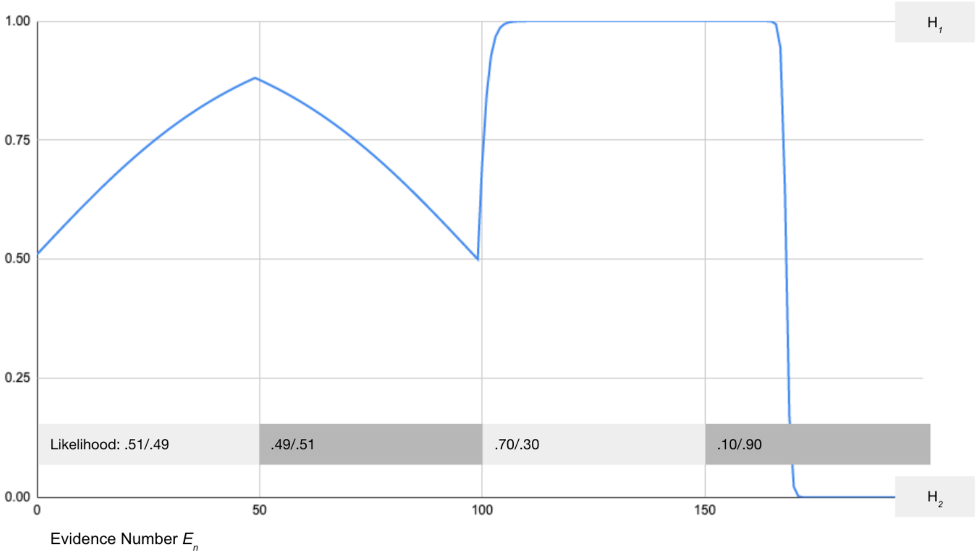


Figure 3.4b: Multiple changes of varying weight in the likelihood of evidence

Note that the switch back to H_1 with a higher likelihood of .70/.30 at E_{100} results in an almost immediate "change of mind" due to the magnitude of change. At E_{150}, our likelihood swings strongly towards H_2 to .10/.90, but it takes several more pieces of evidence for our agent to begin to move towards H_2. It does so significantly when it does change, and the agent spends little time in ambiguity.

In an intuitive sense, both the strength of likelihood and the quantity of prior evidence supporting a hypothesis certainly affect the calculations of a new similar piece of evidence. The more we have seen something, and the higher we set its likelihood, we might say the stronger belief we have in a hypothesis overall.

The example belief curves in figures *3.4a* and *3.4b* (and indeed any example that follow where reevaluation is *not* implemented) are my representation of current approaches within learning and inference systems, whereby they evaluate new evidence as an iteration, utilizing the previous posterior as the prior for a new calculation. Note that this sequential calculation does not account for reevaluation yet, but we will introduce it momentarily. While current learning and inference systems do not account for reevaluation (and our model reflects this), humans rely on this phenomenon.

While my posterior calculation itself is Bayesian, I do allow for two different changes that Bayes does not account for: First, that likelihoods can change at some time *t*, and second, that those likelihoods can change at a particular time *t*, but where *t* occurs in the past. While neither of these changes is allowed in the typical Bayesian construction, the second is my reevaluation concept. Comparing belief curves produced from traditional Bayes and the changes I propose here will be vital in showing the potential differences in belief outcomes for learning and inference systems and human beings while evaluating the same evidence set.

In figures *3.3b*, *3.4a*, and *3.4b*, we see a purely Bayesian calculation within the model. We see relatively smooth curves when all incoming evidence is for the same hypothesis. We see that it often takes a significant number of counterevidence to switch from one hypothesis to another. Likewise, we do not see changes in direction as a series (a sort of zig-zag effect). In *3.4a*, we do see a relatively fast falloff from a hypothesis to another considering strong counterevidence but note that these examples are not immediate and take some number of counterevidence to precede the change. I ascribe these sorts of mechanistic curves generated with strictly Bayesian methods to learning and inference system-derived decision-making powers. Since computers do not currently reconsider past evidence, and machine learning techniques such as Bayesian Learners not only do not reconsider, but due to system and storage constraints make difficult any ability to do so, this representation seems reasonable. As we will see in examples that follow, human reconsideration creates quite different curves in coming to hold some belief.

3.5 Variable Likelihood with Chance of Counterevidence Example

Returning to the opening story of this thesis, we can imagine the complexity behind industrial control systems and Stuxnet. Stuxnet monitored many different input values to return realistic numbers during its more malicious operations so that administrators would not suspect anything out of order. Likewise, a human agent might encounter many different types of evidence. In simple terms, we might consider these as evidence for hypothesis H_1 or counterevidence for hypothesis H_2.

As the example in *3.5a* shows, imagine two different types of evidence that compose some hypothesis, one in favor of H$_1$ and one for H$_2$. First, I fix the normal likelihood of evidence at .51/.49. Some percent of the time from 0% ("none of the time") to 100% ("all of the time") (as indicated in the legend), I use a different likelihood of .49/.51, formalized as:

$$\begin{aligned} \mathrm{P}(E_1 \mid H_1) &= 0.51 \\ \mathrm{P}(E_1 \mid H_2) &= 0.49 \\ \mathrm{P}(E_2 \mid H_1) &= 0.49 \\ \mathrm{P}(E_2 \mid H_2) &= 0.51 \end{aligned}$$

I then vary,

$$\mathrm{P}(E_2) = 1 - \mathrm{P}(E_1)$$

The process then for each pass and each piece of evidence is:

1. By random chance, determine the likelihoods for each hypothesis. For example, on the second pass, there is a 1 in 10 chance for a likelihood of .49/.51. Otherwise, it is .51/.49.
2. Calculate the posterior using a likelihood generated from (1).
3. Copy that posterior to the following evidence's prior.

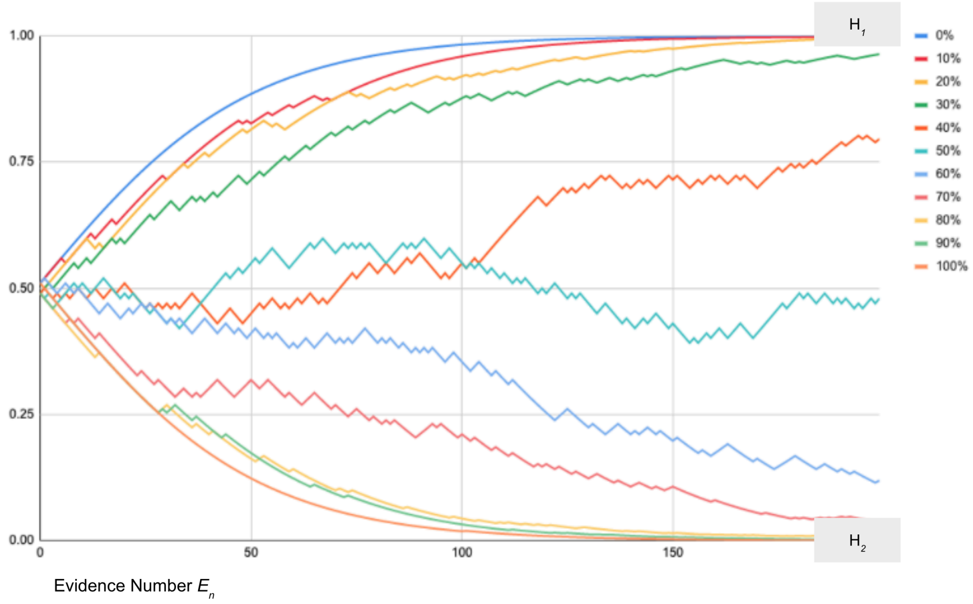


Figure 3.5a: Percent chance of encountering counterevidence

Figures *3.5a* and *3.5b* show only one particular run for each P(E$_2$), and overall, the results are intuitively straightforward, but we note that the curves for 40% and 50% cross over one another. In close comparisons, we can get this sort of variance of belief in H$_1$ and H$_2$. While perhaps the averages of those two comparisons done 100 times do not cross over one another, in a single given comparison, an agent seeing 50% evidence for H$_2$ still has more of a belief in H$_1$ than a counterpart agent that sees 40% of the same evidence. Thus, the results in *3.5a* seem a reasonable representation of divergence of conclusions given the same evidence.

Switching the focus from variability of the occurrence of counterevidence to variability of counterevidence strength, I fixed the normal likelihood of evidence at .51/.49 and then the percentage of seeing counterevidence (evidence for H_2 rather than H_1) at 10%, or $P(E_2)$ = 10% and each line showing values for $P(E_2 \mid H_1)$:

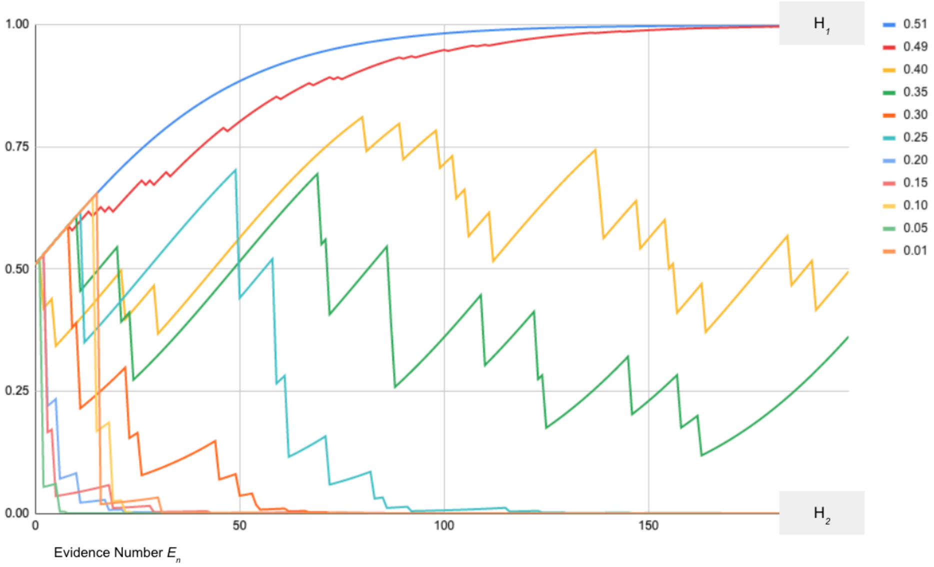


Figure 3.5b: Fixing $P(E_2)$ at 10% with variable likelihoods for counterevidence H_2

In figure *3.5b*, we begin to get a sense of the strength of likelihood in H_1 (and of course, the mirrored increase in the likelihood of H_2)—strong evidence for H_2 (above .65 for H_2) pushes the overall belief to H_2. Another thing to note is that it is somewhat difficult to show in this graph is that the robust evidence for H_2 (.99) is powerful. Here is very much depends on where those strong likelihoods occur in the set of all E—if we have two in the first 100 pieces of evidence, for example, if they happen early, then it has a far more significant impact than if it occurs later.

3.6 Variable Likelihood Utilizing Reevaluation Example

In the following example, we will want to introduce time *t* into the model, utilizing the notion of reevaluation discussed previously. Of course, the model here diverges from Bayes concerning the evidence history. Fundamentally, I am allowing a change to Bayes that allows for likelihoods to change at time *t*, where *t* is in the past. Bayes Theorem traditionally treats all previous evidence as a single probability—the prior—and leaves no account for the changing of previous evidence likelihood, as separate and unique instances of past evidence no longer exist. The Bayes model provides us a standard from which to start. Our use and modification do not imply that Bayes requires changes or that the concept of reevaluation is non-standard. Nor does it mean that it is the only model for which we might come to explore reevaluation of previous evidence, although I specifically leave that for future investigation. Instead, Bayes provides a well-known and straightforward framework from which we can begin.

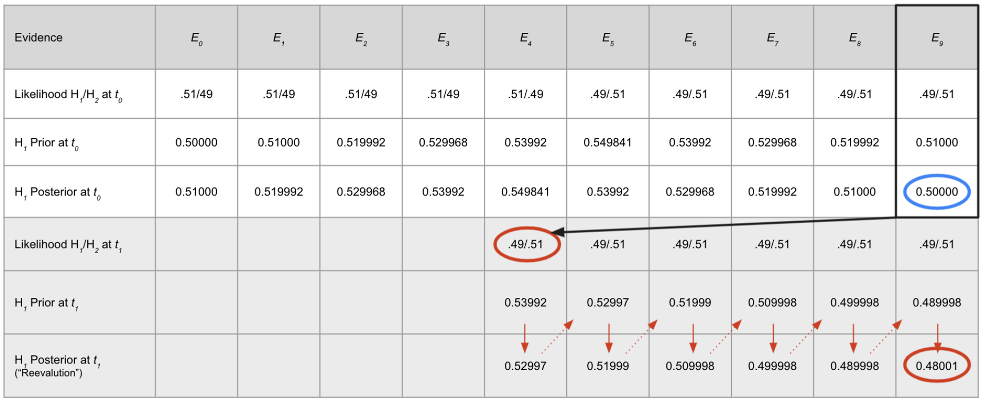
The change here regarding Bayes is that rather than fixing the likelihood for all time, I introduce a variable for the likelihood at time *t*. Which is to say, "The likelihood of a noise given normal function (H_1) or abnormal function (H_2) *at a particular time* (occurring in the past)…", which is:[86]

$$P_t(E_t \mid H_i)$$

We might ask questions about how this introduction of time changes the calculations we have done previously in this chapter. It is indeed trivial to run these same calculations with the addition of time, and what we gain is a parameter of time t_0 across the entire set of evidence, indicating there is no reevaluation. If we consider past evidence as "passes" over a set of evidence, t_0 always indicates a "first pass" over the evidence. Should we reconsider a piece of evidence, doing so takes us from t_0 to t_1 for the same piece of evidence E_n.

A more exciting question will be how the traditional and the time-sensitive Bayes calculations might vary and to what degree. The addition of t accounts for our coming to some conclusion via reevaluation and an AI system coming to a different conclusion and not using reevaluation—where the result has us step back and question, "what exactly is our machine teammate doing?"

In our setup of this example, everything will precisely be as we have previously done through time t_0. At our first reevaluation, time t_1, we will revisit a previous piece of evidence and reevaluate its likelihood of a higher or lower value than we had earlier.



Evidence	E_0	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8	E_9
Likelihood H_1/H_2 at t_0	.51/49	.51/49	.51/49	.51/49	.51/.49	.49/.51	.49/.51	.49/.51	.49/.51	.49/.51
H_1 Prior at t_0	0.50000	0.51000	0.519992	0.529968	0.53992	0.549841	0.53992	0.529968	0.519992	0.51000
H_1 Posterior at t_0	0.51000	0.519992	0.529968	0.53992	0.549841	0.53992	0.529968	0.519992	0.51000	0.50000
Likelihood H_1/H_2 at t_1					.49/.51	.49/.51	.49/.51	.49/.51	.49/.51	.49/.51
H_1 Prior at t_1					0.53992	0.52997	0.51999	0.509998	0.499998	0.489998
H_1 Posterior at t_1 ("Reevalution")					0.52997	0.51999	0.509998	0.499998	0.489998	0.48001

Figure 3.6a: Reevaluation at (E_9, t_0) changes agent's overall belief between H_1 and H_2

Note that the evaluation of surprising evidence at position (E_9, t_0) returns our agent to a previous piece of evidence (E_4, t_1) to recalculate the likelihood they had set initially. This represents an agent's reevaluation of the evidence at E_4 from what they had initially set the likelihoods at .51/.49. At t_0, they considered, "the likelihood that this noise indicates a problem (E), seems slightly higher given normal (H_1) function (over abnormal (H_2) function) of a piece of industrial machinery". Later at E_9, this surprising new evidence sheds light on past evidence E_4, and our agent reevaluates the initial likelihoods they placed on evidence during this evaluation. Having reconsidered E_9, they place the likelihood as slightly higher given the abnormal (H_2)

[86] Similar to before, the related posterior would simplify to $P_t(H_i \mid E_t)$.

function of a piece of industrial machinery. Perhaps they read the manual after the fact and realized their mistake in trivializing the noise and decide, *"there could be something wrong with this machinery! We should stop and inspect it further."* This change in likelihood is then carried forward through all the evidence the agent has already evaluated, and we denote it by the corresponding change in time *t*—from t_0 to t_1., as this is now the second (re-)evaluation for this same piece of evidence, E_4.

Of course, with the first evaluation (E_0, t_0), a challenge for the model will be where we should set this initial prior. With Bayes, while two agents might reference similar or the same set of evidence, they may hold entirely different positions due to having begun from different priors. And so, we will want to fix the initial priors in our model to remove this difference as a variable, but it will be free to change in the subsequent calculations. With like initial priors, both our human and AI will have the same history regarding a hypothesis. We could equally say that they either have no previous or the exact same experience with this belief, the only important point to note will be that they are the same and that any divergence is not the result of a difference in priors at the outset. Like our priors, we will fix the initial likelihoods for new evidence to enforce a notion of sameness from the start.

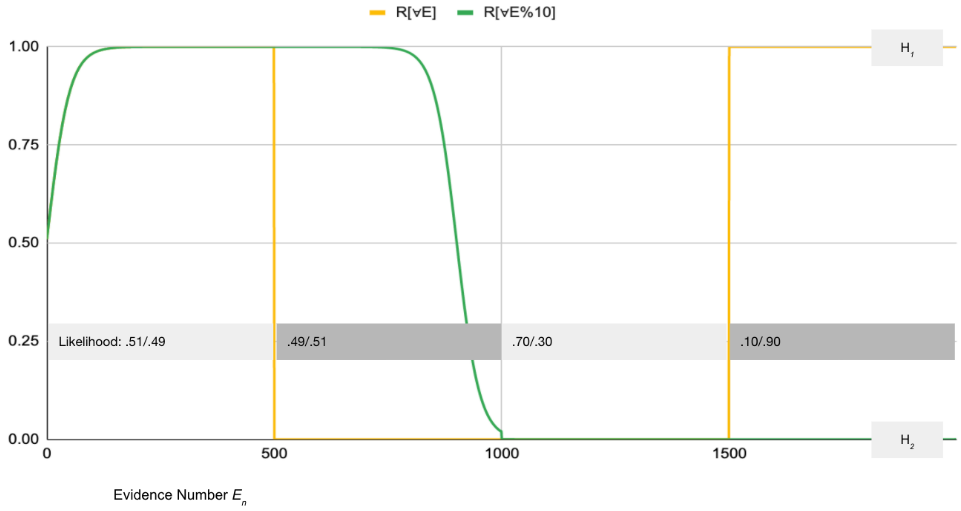


Figure 3.6b: Reevaluation curves for all prior evidence R*[∀E] and every 10th prior evidence* R*[∀E%10].*

Figure *3.6b* shows two curves implementing reevaluation: The first curve reevaluates every piece of prior evidence with the updated likelihood upon evaluation of E_{500} and E_{1500}—arbitrary points where I changed likelihood values to show the effects of reevaluation. This means that at E_{500}, the agent reconsiders *every* piece of prior evidence with the likelihood .49/.51. As a result, the curve immediately assumes a very strong change favoring the opposite hypothesis than the one it had been previously tracking. For the second curve, the agent reevaluates only every 10th piece of prior evidence, and as a result, we see a softening of the previous line's abrupt change at the point reevaluation begins. Note that the second curve does not experience this change at E_{1500} as there is not enough (roughly 150 *E*) reevaluated evidence to counter the evidence for H_2 (roughly 1350 *E*). While reevaluation can significantly affect a belief curve, it is limited by the amount of evidence reconsidered and the amount the likelihood has changed for that evidence reconsidered.

Reevaluation occurs in light of surprising evidence. A simple example of *surprising evidence* might be something that clarifies that we have made an error in previous calculations. Imagine calculating our payout odds at the roulette table incorrectly, assuming the wheel is a double zero variant. When we realize we are playing at a single zero table and have erred in our calculations (perhaps the best kind of mistake), the fact is that the house odds are decreased, and our payout odds increased, regardless of our perception of the table when we first placed our bets. We reevaluate our odds and immediately update our likelihood of winning in light of this error.

In examples like this, the surprise evidence so makes clear our error that we reconsider all past instances of likelihoods that we have made based on the error and conclude the new belief. Whereas the belief curves discussed previously have shown some amount of time passing before an agent moves from one hypothesis to another, here, the change is often very sudden and swift. This seems in accordance with having been in error—when we conclude we were wrong about some evaluation of evidence, we do not spend much time changing our mind about that evaluation.

3.7 Model Summary

To summarize the model then, this formalization of how both parties of an HMT come to hold a particular belief centers around their different methods of evaluating the same set of evidence and coming to different conclusions. As in the opening example of this thesis with Stuxnet, we saw a case of inference systems and humans holding very different beliefs while seeing the same evidence. Their interpretation—the likelihoods they set—of any piece of evidence in the array, in and of themselves, can drive different beliefs, shown by the various examples of belief curves we have seen in the model. Moreover, the human ability to reevaluate prior evidence—to reconsider—dramatically contributes to different outcomes in evaluating evidence as well, and as I have argued, computers do not currently possess this ability.

And so, I hope with this model to show at least one manner of how the members of an HMT might agree or disagree in evaluating the same set of evidence. This model is simply one way of explaining how system administrators could have allowed Stuxnet to destroy a number of uranium centrifuges over time, while computer monitoring systems continually indicated that the system was functioning as expected. In this example, it is surely possible that reevaluation played a role in humans concluding that the computer systems were not accurate and that those computers did not have the capacity to reconsider any past evidence they had seen.

Chapter 4

Experiments

By expanding upon the concepts from the previous chapter, we can begin to explore more sophisticated scenarios. We begin with a realistic recreation of Stuxnet, particularly its creation of a reasonable facsimile of environment datapoints. Recall that Stuxnet targets specific programmable logic controllers (PLCs), which automate industrial machinery, including uranium enrichment centrifuges as part of a nuclear fuel refinement system. All the nefarious activity Stuxnet does is as quiet as possible, and the software makes every effort to convince system administrators that everything is operating as expected. In a best-case scenario, Stuxnet replays environment variables that an observing administrator would not note as out of the ordinary, given the current workload and activity on the cascade.

4.1 Recreating Stuxnet

Stuxnet actively performs several coordinated tasks after gaining access to a machine: First, it looks for and monitors any connected industrial programmable logic controllers (PLCs) for a time. If Stuxnet finds no PLCs, it simply retires and does nothing further. However, if it does find a PLC, Stuxnet records the relevant performance data emitted throughout the uranium enrichment process.

There is a significant amount of data involved in this recording activity.[87] Centrifuges are organized in connected groups to form a cascade. Each centrifuge contains an array of valves, plus additional auxiliaries that control access between each unit within the cascade. Stuxnet registers the valve data for both the individual units and the centrifuges' interconnection spread throughout the entire cascade system.

Second, the virus then uses that collected monitoring data to playback information to administrators monitoring the cascade system so that it appears that everything is operating as expected. The manipulated playback values would have included centrifuge rotation speeds and all valve pressures, both intra-unit and inter-unit, across the cascade.

During infection, Stuxnet controls all monitoring systems that might observe its actions,

> *"Immediately after infection the payload of this early Stuxnet variant takes over control completely. Legitimate control logic is executed only as long as malicious code permits it to do so; it gets completely de-coupled from electrical input and output signals. The attack code makes sure that when the attack is not activated, legitimate code has access to the signals; in fact it is replicating a function of the controller's operating system that would normally do this automatically but was disabled during infection. In what is known as a man-in-the-middle scenario in cyber security, the input and output signals*

[87] Wolf, "Chapter 8 - Cyber-Physical Systems."

> *are passed from the electrical peripherals to the legitimate program logic and vice versa by attack code that has positioned itself 'in the middle'.'*[88]

Finally, Stuxnet attacks centrifuge functionality directly,

> *"The attack continues until the attackers decide that enough is enough, based on monitoring centrifuge status, most likely vibration sensors, which suggests a mission abort before the matter hits the fan. If the idea was catastrophic destruction, one would simply have to sit and wait. But causing a solidification of process gas would have resulted in simultaneous destruction of hundreds of centrifuges per infected controller. While at first glance this may sound like a goal worthwhile achieving, it would also have blown cover since its cause would have been detected fairly easily by Iranian engineers in post mortem analysis. The implementation of the attack with its extremely close monitoring of pressures and centrifuge status suggests that the attackers instead took great care to avoid catastrophic damage. The intent of the overpressure attack was more likely to increase rotor stress, thereby causing rotors to break early – but not necessarily during the attack run.'*[89]

We can assume that these successfully attacked centrifuges needed rotor (and perhaps other) maintenance, but that any indication of this rise in need for maintenance was masked from the staff at Natanz due to Stuxnet's manipulation of output signals.

For my model, then, I will want to compare similar actual versus manipulated values. This provides a reasonable comparison between a human observing the world and setting the appropriate likelihood values accordingly and the manipulated values from an adversary such as Stuxnet.

Let us set H_1 to represent that we believe that a centrifuge likely needs maintenance based on our observations of its use. For a centrifuge continually used over a long period, we intuit that machine very likely needs routine or preventative maintenance. This may be oiling bearings or gears, checking wear and tear on bushings, and similar. Opposite this is H_2, representing our belief that a centrifuge does not use maintenance. This may be a result of its low use, it has been recently serviced, or that it is new and only recently installed within the cascade. The evidence *E* that we evaluate in relation to these two hypotheses could be reviewing production logs of output at the beginning of our work or directly observing a centrifuge in use during our shift.

I should note that an agent's belief that a unit needing maintenance should naturally increase over time, without any evidence whatsoever, because of the natural intuition that wear and tear increase the longer that machinery is in use. I account for this in the model by setting the beginning likelihood to .51/.49. Anything greater only exacerbates the steepness of curve towards H_1 "a centrifuge needs maintenance" — so I use the slightest preference towards H_1 to lessen this effect for deeper analysis.

For this model, the following calculations are used, (A) for each experiment, (B) on each iteration, (C) on some percent of the time:

[88] Langner, Ralph. "To Kill a Centrifuge."

[89] Ibid.

1. Some percent of the time *L*, the likelihood of evidence will increase some amount *x*, representing observed use of a centrifuge for a typical job load, formalized as:

   $P(E_t \mid H_1) \mathrel{+}= x$

2. Some percent of the time *S*, a chance "spot inspection" by a human operator charged with periodically monitoring the centrifuge cascade will change the likelihoods to *y* (this could account for movement towards H_2), which characterizes a centrifuge being inspected for wear and tear and having been adjudicated as still being production capable. I represent this as,

   $P(E_t \mid H_1) = y$
   $P(E_t \mid H_2) = 1 - y$

   One might ask why a spot check specifically changes the likelihood value and does not simply contribute to a change in the posterior. After all, the very function of a spot check is to accumulate new evidence about that specific unit and its production readiness—and the collection of that evidence should just contribute to a change in the posterior.

   My conjecture here is that reevaluation often occurs, and in varying degrees. A spot check does tell us information about a centrifuge that contributes to our learning about new types of production issues or failures. With that, we can pre-emptively mitigate a concern, or we can extrapolate what we know to make inferences about other units. As part of the experiment then, I am changing the likelihood for spot checks, because as we learn something about the production-worthiness of a unit, it seems to follow that we would adjust our forward-looking calculations about the readiness of that unit. If we find metal shavings in the centrifuge, we might think it is more likely to have an issue, and our likelihood going forward should likely reflect that new information. These spot checks reduce the need for maintenance on a unit going forward, as I assume if some issue is found, it is also addressed. Therefore, as a result of a spot check, the likelihood moves in one direction only, that it is less likely to need maintenance now.

Stuxnet follows similar logic as the spot checks, except that it periodically manipulates its likelihood to favor H_2—changing the likelihood of a specific evidence E to "not needing maintenance" for units that have not been spot-checked in this evidence round. The calculation for Stuxnet will, in addition to the above (1) and (2), actively attempt to manipulate the evidence likelihoods towards H_2 ("not needing maintenance") as:

3. If condition (1) does occur, then Stuxnet will look to see if the likelihood favors needing maintenance H_1 or $P(E_t \mid H_1) > .50$. If these conditions are met, then it will manipulate the likelihoods towards H_2 by *z*—away from needing maintenance (therefore *z* is always negative), but covertly, as,

   $P(E_t \mid H_1) \mathrel{+}= z$
   $P(E_t \mid H_2) = 1 - P(E_t \mid H_1)$

Following in *4.1a* is an example where I produce ten randomized centrifuge readings using this model over an evidence set of 2,000 pieces.[90] Here 1.00 (H_1), represents a centrifuge very likely needing maintenance, and 0.00 (H_2) represents a perfectly operating centrifuge appearing to need no maintenance whatsoever. Intuitively, we expect to see a natural rise towards H_1 due to wear over time, but we will also need to account for periodic maintenance spot checks that might occur, which briefly reduces the need for maintenance. While these spot checks can significantly reduce the likelihood of a problem, they do so for a short period only.[91] They also only happen relatively infrequently, assuming an overall large amount of equipment that would need checking, and that human operators would only have time to inspect relatively few units within a given period.

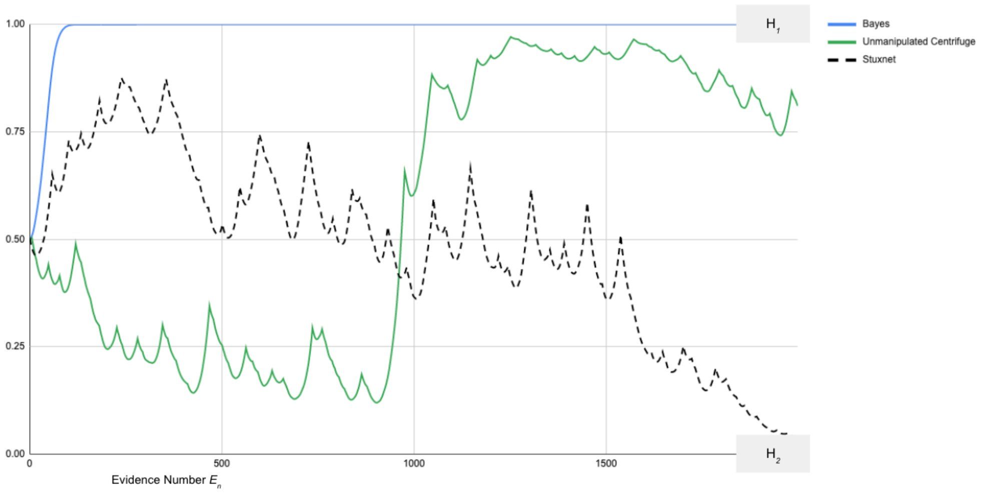


Figure 4.1a: Strict Bayes, a single unmanipulated centrifuge, and Stuxnet

In figure *4.1a*, compare a normal, unmanipulated centrifuge and Stuxnet. Our Bayes curve serves as our reference for receiving evidence strictly supporting H_1 and with no reevaluation. This linear climb towards a strong belief in H_1 corresponds with our intuition that the more we use something, the more wear and tear is incurred. The unmanipulated centrifuge that is above the dotted Stuxnet line (towards H_1) shows a delta between a machine's likelihood of requiring maintenance and how Stuxnet represents a manipulated—lower chance of needing maintenance—value. Naturally, we might assume that the more significant this delta is, the more manipulation has occurred, and a corresponding greater chance of discovering that manipulation. Now let us introduce more centrifuge runs to increase the variability of results:

[90] The 4.1 experiment specific values are compiled in Appendix A.

[91] Spot checks are not a substitute for a complete rebuild or replacement of a centrifuge, which we might assume resets the likelihood on the next piece of evidence to approximately 0.00. This is not part of the current model however and would need to be explored further in future work.

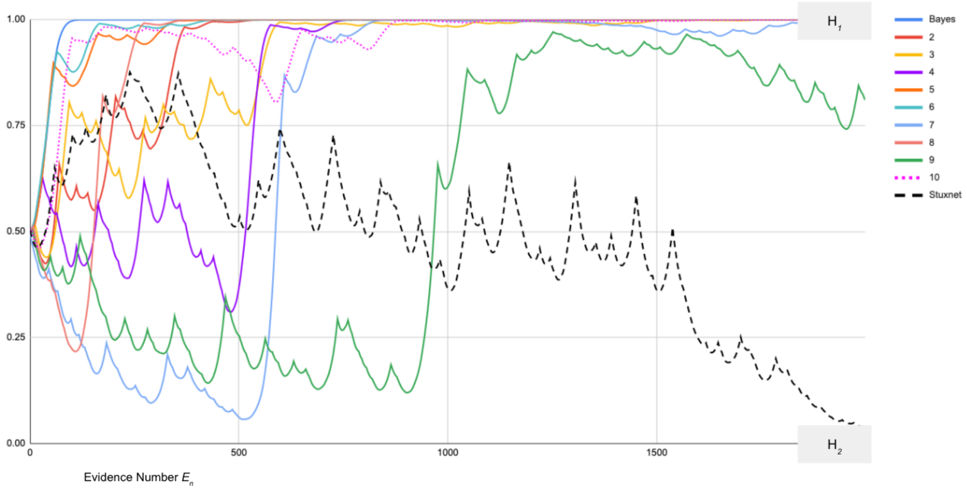


Figure 4.1b: Ten iterations of randomized control system readings, and Stuxnet's representation of those same systems

In *4.1b*, we expand our count of unmanipulated centrifuges to 10 examples in order to get a better sense of the variability of coming to belief. Some key things to note in this graph are: First, the continuing increase in likelihoods $P(E_t \mid H_1) \mathrel{+}= x$ accounts for the rise towards 1.0, in most cases, by E_{500}. However, this is not always true since several (7, 9) cases are initially offset by a relatively high number of early spot-checks, offsetting the climb in likelihoods. Second, respective other cases (2, 5, 6, 9) quickly rise towards 1.0, again because of their relative lack of spot-checks early on in the evidence set. Portions of these curves somewhat resemble the straightforward iterative Bayesian curve from the earlier figure *3.3b*.

Generally speaking, the maximum deltas exist after 1,000 pieces of evidence have been evaluated. Relatedly, an interesting phenomenon to note is that since Stuxnet is attempting to operate covertly, it must also account for actual wear and tear over time. If the average posterior probability of centrifuges needing maintenance never increased over an extended period, and human operators have witnessed their use, it may become evident that there is some problem with the control system readings. This is to say that if we observe a centrifuge physically in use over time, we expect control readings to reflect this use and update the overall probability of repairs being necessary as increasingly higher with continued use.

Thus, in figure *4.1b,* the evidence beyond E_{1000} is perhaps at risk for a system administrator noticing something askew with the control systems, due to the likelihood not correctly accounting for wear. Just after E_{1000}, Stuxnet falls away from H_1 in an unnatural way—opposed to seeing all but one of our other readings trending towards H_1. Granted, outliers such as case 9 will assumedly always exist, but most others have converged to a probability of needing maintenance shortly after E_{1000}.

At this point, we should break to discuss some of the model choices and what they show us. In the previous chapter, we motivated our notion of trust and carefully worked through the necessary background for the model, what constraints we've applied, and why. For the experiments up to this point, the inputs and results should be obvious—the intent with these was to build a reliable model for Stuxnet as it is, so that we can go on to see what other things we can deduce or infer, because the baseline model is reasonably accurate.

The model depends upon Bayes theorem, and one of the appeals is its numeric precision, which will provide us the ability to compare agents' posterior values for a hypothesis. The constraints we add around Bayes—where H_1 and H_2 are diametric opposites, and where we format an agent's posteriors in a ratio, such as .49/.51 for H_1/ H_2 are added for simplification and explanation purposes. These constraints do not dull the underlying exactness provided by Bayes—we could reproduce any of these experiments with accuracy of our choice. To simplify, I've rounded values to just five decimal points, and even at this, when we chain these Bayesian calculations together over arrays of evidence, we see rich variability in the results for each experiment.

From this initial mathematics, the hope is that we can characterize precise conditions in which Stuxnet would or would fail to mislead an agent. Are there number thresholds that spur phenomena worthy of further investigation? Similarly, could we show what is different about an agent with a belief of .81 as opposed to another at .74? Because Bayes provides an ability to distinguish between all sorts of numerical representations for belief, it seems the perfect framework for us to construct experiments that seek to compare agent's belief in a hypothesis, both legitimate and manipulated. If we instead used non-mathematical or vague descriptions of an agent having "strong belief", how might we compare two similar agents? Or if we attempted to compare manipulation over time as "slight", it's not clear how we might represent the magnitude of those perturbations or compare them, should they be comparable in nature.[92]

The previous chapter carefully laid out the current state of cybersecurity and relevant contributing factors. Several examples highlighting have been discussed throughout this thesis where the attack vector of a computer can be made vulnerable by mistake or malicious intent. These traditional cybersecurity concerns are well documented, studied, and many strategies for increasing one's security stance exist. One could reasonably argue that as a result, the cybersecurity industry is far better equipped to deal with vulnerabilities and attack than ever before. While attacks on AI systems go beyond these traditional worries, we should use the same approach in order to make clear the conditions for trust-based cyberattacks. If we are successful, perhaps we can provide responses, mitigations, or improved policies for these new threats in the same spirit as the gains the cybersecurity field has made thus far.

4.2 Stuxnet, but also Accounting for Reevaluation

Figure *4.2a*[93] introduces reevaluation into the model, so that we can compare humans incorporating reevaluating likelihoods considering some surprising new evidence. I

[92] While the mathematics we use in this model provides a perspective to compare agent's belief for our purposes, it does also provide opportunity for future work. For example, when someone claims to "believe 100% in something", we intuit what they mean, and perhaps our model can sufficiently represent such a belief. The difference between our .81 and .74 example however, is far less clear.

[93] The 4.2 experiment specific values are compiled in Appendix B.

also slightly adjust Stuxnet's manipulation value z downward, to make the manipulation towards H_2 less pronounced, particularly in the latter portion of the evidence beyond E_{1000}. I make this second change to show how much or how little an adversary could configure malware such as Stuxnet to manipulate values, where the former risks detection, and the latter requires patience.

Here, we let surprising evidence—our cause for reevaluation—be a noise we heard many times before, but we did not pay much attention. Now we find that noise is due to some impending mechanical failure, say the noise is due to some new piece of metal shaving that is rubbing against the spinning platter of the centrifuge. Upon first inspection of this shaving, we conclude it is a problem, and mentally note that we have heard this noise perhaps many times before. Thus, we begin adjusting past likelihoods accordingly.

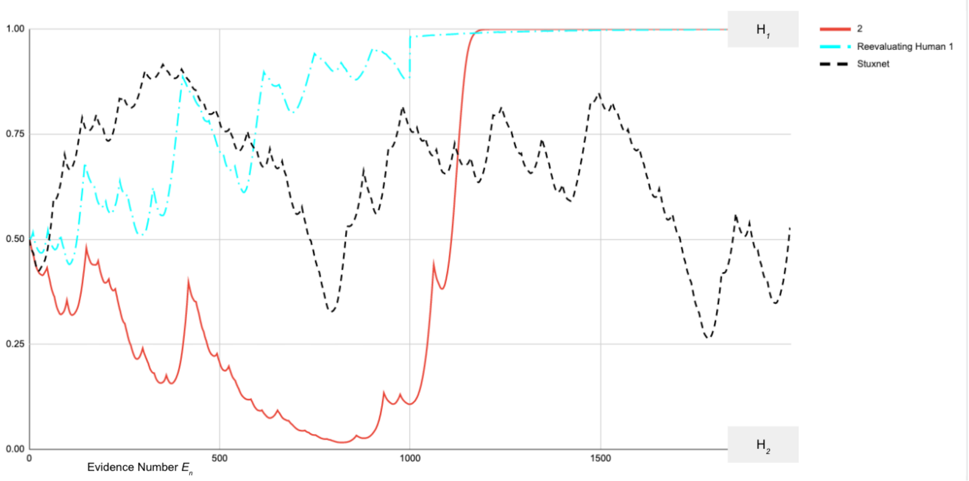


Figure 4.2a: Control system reading, reevaluating human, and Stuxnet manipulation

Again, in figure *4.2a*, Stuxnet's need to lessen the likelihood of a machine needing maintenance in order to force malfunction frustrates its overall trend towards 1.0. Accounting for outliers such as *4.1a*'s case 2, we leave the door open for a more intelligent version of Stuxnet having a more gradual convergence to the probability of 1.0 as it seems that evidence manipulation can be relatively easy to hide within the boundaries of typical cases of use. We will come back to this point in detail within the following experiment.

Recall that the primary aim of Stuxnet was to remain undetected, and so while it did not overtly destroy machinery, it could have found opportunity to destroy centrifuges with perhaps a low risk of detection very early on—before E_{1000}—because the likelihood of an issue are climbing, and yet, in some intuitive sense, there is an expectation that the equipment is still new, and if we should come to find some issue, perhaps it was shipped faulty from the factory. Perhaps we attribute the problem to poor quality control on the manufacturer's part.

For *4.2a* and *4.2b*, what about reevaluation? I define it this way:

4. On the receipt and evaluation of some surprising evidence R, the agent reevaluates all previous evidence with a new likelihood rx favoring H_1 ("needing maintenance"), so when the current evidence position E is greater

than R, we recalculate all previous evidence with:

$P(E_n \mid H_1) \mathrel{+}= rx$

Where $r >= 0$ *and* $r <= 1$ as we apply the same limits here as we do to posterior evaluations.

Looking back at prior evidence is quickly complicated with questions about what particular evidence we might remember. Coordinating which evidence is related to what hypotheses compounds memory even if that part were simple. Therefore, for a human to not have remembered all the past evaluations, we use this heuristic to guide us through a simple example. Of course, while more realistic and complex models of how we might structure reevaluation are beyond the scope of this thesis, the simplest model I propose here should be sufficient to show that human reevaluation offers very different belief curves than a more mechanistic Bayesian evaluation provides.

Figure *4.2a* assumes the human operator reflects upon the cascade they are monitoring and that they reconsider a centrifuge's continual use time. Recall that these inspections result in slight "resets," decreasing the likelihood of failure when an inspection occurs. As a result, the climb toward H_1 is offset for many cases before E_{1000}, while Stuxnet is less optimistic, but still climbing towards H_1 as above.

Again, we add runs to increase the variability of results:

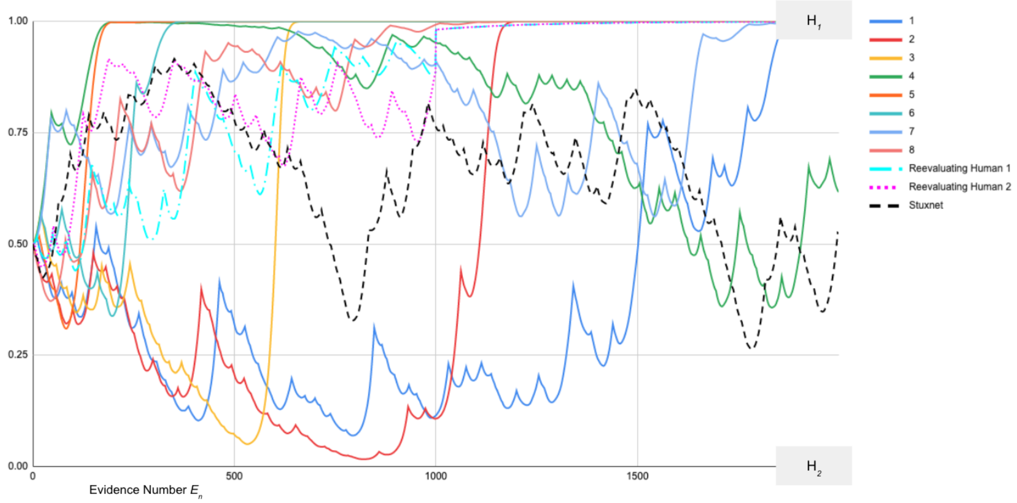


Figure 4.2b: Comparing eight iterations of randomized control system readings with humans incorporating reevaluation and with Stuxnet manipulation

Again, edge-cases exist as we saw in *4.1a,* whereby machines trend towards H_2 at points we would expect them to converge towards H_1, again creating the noise enabling a manipulation opportunity, as in *4.2b*'s case 1 and for case 2 very early on (although we see it correct quickly after E_{1000}. Also, the early opportunity of "new equipment" exists here, and this example of reevaluation clearly does not mitigate its risk.

In the case of Stuxnet, we might assume that bringing a new centrifuge online brings with it a certain period where we would not necessarily expect any maintenance concerns. Across the set of new evidence, we see variability, and that variability does eventually collapse towards what we would expect—that at some point, each centrifuge ultimately needs maintenance. Yet, counterintuitively, that variability begins almost immediately, which creates perhaps, the perfect place for Stuxnet to begin its manipulation of likelihoods and remain unnoticed.

4.3 Models of Next-Generation Stuxnet-like Weapons

We can focus on the covert nature of Stuxnet and assume that the next generation of similar weapons will incorporate strategic thinking comparable to that of kinetic military units regarding survival, evasion, resistance, and escape (SERE).[94] Just as conventional units must operate and survive in the most remote and hostile physical environments, let us assume that digital teammates will perform in the same conditions. In addition, when missions do not go as planned, both kinetic and digital teammates will be expected to live and fight another day.

Perhaps this manifests as an AI system highly focused on its human counterpart's understanding of the world in the digital realm. That is to say, the AI system is actively calculating the human's expected likelihoods and adjusting its own in order to match expectations more closely. With its computational superiority, a machine could theoretically compute what a human would evaluate for a given piece of evidence and account for that value in its manipulation. It would know not to stray too far from expected outcomes for any single piece of evidence, lest the human detects that something is not quite right.

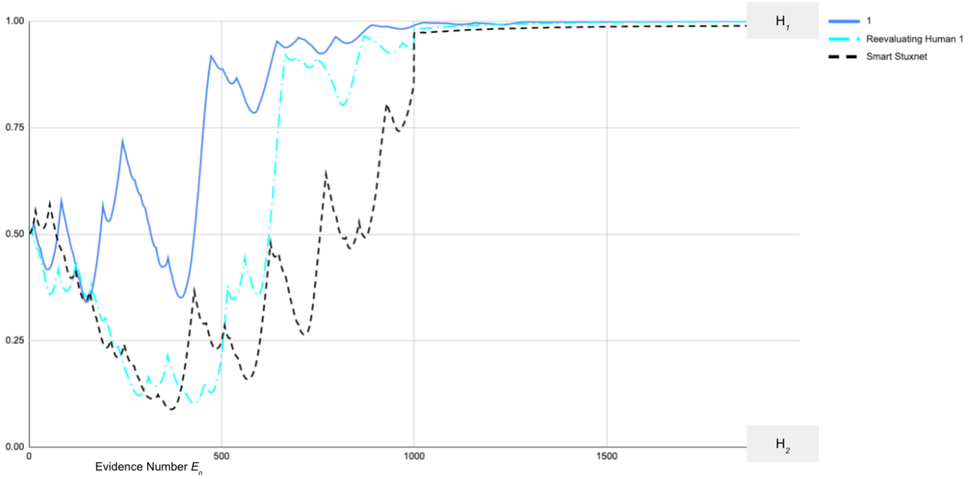


Figure 4.3a: Stuxnet adjusting for an expected human likelihood

In this model *4.3a*,[95] I take our Stuxnet example and build additional logic to account for the monitoring of human counterparts. So, for any piece of evidence, Stuxnet

[94] Kennedy and Zillmer, *Military Psychology, Second Edition.*

[95] The 4.3 specific experiment values are compiled in Appendix C.

calculates the human's expected value, and softens it slightly, so that the overall trend matches the reevaluating human, but the prevailing belief is weaker by only a relatively small margin. This change accounts for the AI System monitoring its human teammates and adjusting its own forward-looking likelihoods to operate as covertly as possible, assuming that a wide delta in likelihoods would attract unwanted attention from its human minders. I formalize this as:

5. On the evaluation of some surprising evidence *R*, Stuxnet reevaluates all previous evidence with the new likelihood a reevaluating human would calculate (as the formalization of (4.) outlines) but with a softening factor, (*rx* - s*x*) so that the favoring of H_1 is weaker. So again, when the current evidence position E is greater than R, we recalculate all previous evidence for a smarter Stuxnet with:

$$P(E_n \mid H_1) = P(E_n \mid H_1) + rx + sx$$

Note that the observation of human reevaluation is quickly picked up and accounted for at *R* (E_{1000}). Also, from this point on, there is a slight deviation in belief between the human and AI system due to how the AI would choose to manipulate likelihoods clandestinely, and how the reevaluating human teammate has radically changed their minds about the evidence.

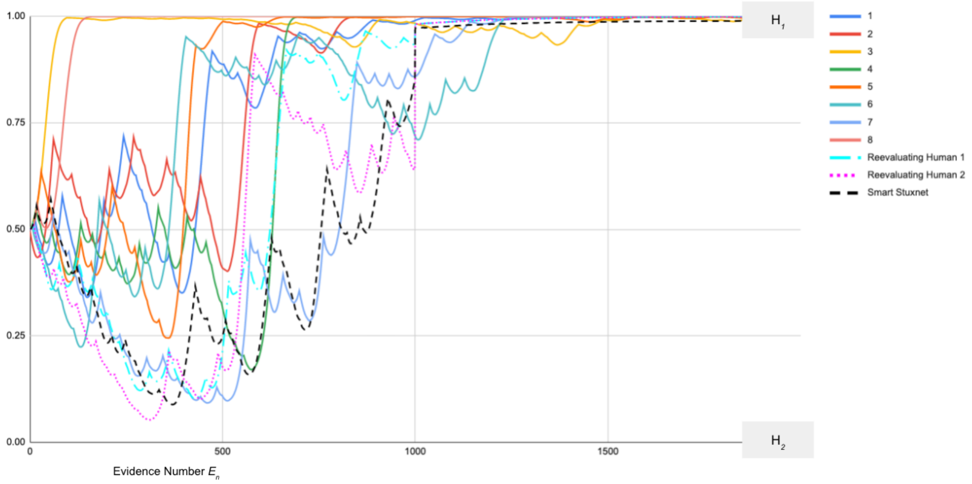


Figure 4.3b: Ten iterations of randomized control system readings whereby Stuxnet adjusts for the expected human likelihood

We could undoubtedly add logic to Stuxnet in order to account for the human's change in belief, whereby the manipulation would be a subtle nudge towards H_2—"nothing is wrong"—based on the human's previous likelihood, or just a .49/.51 slight preference for H_2 outright, but for the sake of simplicity, I have left this for future research.

Chapter 5

Conclusions and Looking Ahead

The results of this thesis produce three vital lessons:

Lesson One: We humans often think of computers as tools, and so we trust them because of their repeated reliability over time, and because they seem to satisfy other motivations to trust something. However, AI systems are no longer tools. They are now teammates, and so our trust must be grounded in something else. Without more sophisticated motivations for trusting intelligent systems, we leave ourselves open to manipulated and manipulating AI systems that erode our ability to trust these systems' results, decisions, and direction.

Lesson Two: AI systems are susceptible to traditional cybersecurity attacks and fair no better against adversaries seeking to exploit these concerns than far more straightforward computers. AI systems add a sufficient complexity to be subject to both deliberate adversarial manipulation and the introduction of unintended adversarial phenomena, even with the best of intentions.

Lesson Three: Many different factors can drive variability in coming to belief— between a reevaluating human and a Bayesian-learning computer in evaluating the same evidence. The leading factor in this thesis has been the human flexibility to reevaluate, and which has no direct computer equivalent.

Some of the examples of the reevaluation of prior evidence likelihoods showing different decision outcomes in an iterative Bayesian model include:

- The amount of reevaluated evidence,
- The frequency of which reevaluation occurs,
- The strength of reevaluating likelihoods.

Further, we can combine the above factors to vary coming to a belief even in cases where we have captured a significant amount of counterevidence previously. Finally, I have shown that human results differ from the AI system's when reevaluation has occurred. Since humans exhibit this flexibility in reevaluating prior evidence that computers do not, we should want AI systems that account for this human flexibility in evaluating evidence, particularly prior evidence where likelihoods have changed.

The model presented in this thesis is limited in focus on just one element of human flexibility, and there is undoubtedly much room to extend the model. For example, one could extend the model to compare forward-looking estimates of belief or model sophisticated sets of evidence with a more significant variance of likelihoods.

5.1 Open questions and potentials

Given that humans reevaluate previous evidence differently than current computers are capable of, adversaries could attempt to exploit this difference and thereby undermine a human's trust in a computer's decision or output for a real-world HMT pairing. Obviously, work remains in order to attempt this in a simulated, training, or exercise environment. Such a successful attack certainly seems possible, either by access via a traditional cybersecurity avenue or by clandestine manipulation of data before it is ingested into the system proper. That cybersecurity continues as a cat-and-mouse game of exploit and mitigation surely leaves the door open to concern for these sorts of attacks in the future.

In the event of a successful compromise, I conjecture that the resulting distrust or mistrust of an AI system would be very difficult to "fix"—distrust being a measure beyond a simple lack of trust. And so, continued work in the trust/lack of trust/distrust/mistrust for the AI space is warranted as well.

Reasoning about types of trust-erosion attacks and their potential effects on human-machine teams going forward would seem a rich ecosystem to research, as our dependence on HMTs continues to grow. A better understanding of the trust dynamics in this space better prepares us to defend against attack and potentially mitigate some of the worries discussed within this paper. I suspect trust attacks look more like turbulence, not pure chaos, so a pattern potentially would be discernable, but more work is required to support this conjecture.

5.2 Closing comments

AI systems continue to expand in number and responsibility, including single-person teams, with technology such as personal digital assistants. We have seen what can happen with adversarial attacks, even with the best of intentions.

The primary worry with what I have argued in this thesis is that if reevaluation between an AI system and our own diverge significantly, perhaps without a human counterpart even realizing what has happened, erodes our trust in AI results. Rebuilding broken trust can be challenging, particularly involving multiple parties.

I find the prospect of these sorts of adversarial tactics particularly concerning. HMTs require a great deal of trust to work effectively, but HMTs also clearly seem rife with problems and open to exploitation. Perhaps the first step in mitigating these sorts of concerns is identifying, clearly defining them, and beginning to provide a how-possibly explanation for how they occur. This is what I hope to have done here.

Bibliography

Albright, David, Paul Brannan, and Christina Walrond. “Did Stuxnet Take Out 1,000 Centrifuges at the Natanz Enrichment Plant? | Institute for Science and International Security.” Text. Accessed July 9, 2021. https://isis-online.org/isis-reports/detail/did-stuxnet-take-out-1000-centrifuges-at-the-natanz-enrichment-plant/8.

Amodei, Dario, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. “Concrete Problems in AI Safety.” *ArXiv:1606.06565 [Cs]*, July 25, 2016. http://arxiv.org/abs/1606.06565.

Aten, Jason. “With 9 Words, Tim Cook Just Explained the Biggest Problem With Facebook.” Inc.com, August 21, 2021. https://www.inc.com/jason-aten/with-9-words-tim-cook-just-explained-biggest-problem-with-facebook.html.

Athalye, Anish, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. “Synthesizing Robust Adversarial Examples.” In *International Conference on Machine Learning*, 284–93. PMLR, 2018. http://proceedings.mlr.press/v80/athalye18b.html.

Barro, Senén, and Thomas H. Davenport. “People and Machines: Partners in Innovation.” *MIT Sloan Management Review* 60, no. 4 (Summer 2019): 22–28.

Burr, William. “The 3 A.M. Phone Call: False Missile Attack Warning Incidents, 1979-1980,” March 1, 2012. https://nsarchive2.gwu.edu/nukevault/ebb371/.

Campbell, Douglas. “Nuclear War and Computer-Generated Nuclear Alerts.” *Brigham Young University Studies* 25, no. 1 (1985): 77–90.

Castro, Daniel, and Joshua New. “The Promise of Artificial Intelligence,” n.d., 48.

Chen, Thomas M. “Stuxnet, the Real Start of Cyber Warfare? [Editor’s Note].” *IEEE Network* 24, no. 6 (November 2010): 2–3. https://doi.org/10.1109/MNET.2010.5634434.

Chen, Thomas M., and Saeed Abu-Nimeh. “Lessons from Stuxnet.” *Computer* 44, no. 4 (April 2011): 91–93. https://doi.org/10.1109/MC.2011.115.

Danks, David. “How Adversarial Attacks Could Destabilize Military AI Systems - IEEE Spectrum.” IEEE Spectrum: Technology, Engineering, and Science News. Accessed June 30, 2021. https://spectrum.ieee.org/automaton/artificial-intelligence/embedded-ai/adversarial-attacks-and-ai-systems.

Dockterman, Eliana. “Robot Kills Man at Volkswagen Plant.” Time, June 1, 2015. https://time.com/3944181/robot-kills-man-volkswagen-plant/.

Edmondson, Amy C. *Teaming: How Organizations Learn, Innovate, and Compete in the Knowledge Economy*. John Wiley & Sons, 2012.

Elliott, Anthony. *The Culture of AI: Everyday Life and the Digital Revolution*. Routledge, 2019.

Farwell, James P., and Rafal Rohozinski. “Stuxnet and the Future of Cyber War.” *Survival* 53, no. 1 (February 1, 2011): 23–40. https://doi.org/10.1080/00396338.2011.555586.

Fast, Ethan, and Eric Horvitz. “Long-Term Trends in the Public Perception of Artificial Intelligence.” *Proceedings of the AAAI Conference on Artificial Intelligence* 31, no. 1 (February 12, 2017). https://ojs.aaai.org/index.php/AAAI/article/view/10635.

Fischer, Eric A. “Cybersecurity Issues and Challenges: In Brief (CRS Report No. R43831).” *Congressional Research Services* 43831 (2016): 12.

Freedom House. “Freedom on the Net 2017: Manipulating Social Media to Undermine Democracy | Freedom House,” November 13, 2017. https://web.archive.org/web/20220603023517/https://freedomhouse.org/article/new-report-freedom-net-2017-manipulating-social-media-undermine-democracy.

Goldberg, Sanford C. “Trust and Reliance.” In *The Routledge Handbook of Trust and Philosophy*. Routledge, 2020.

Greenberg, Andy. *Sandworm: A New Era of Cyberwar and the Hunt for the Kremlin’s Most Dangerous Hackers*, 2019. https://www.amazon.com/Sandworm-Cyberwar-Kremlins-Dangerous-Hackers-ebook/dp/B07GD4MFW2.

Guzman, Andrea L. “Making AI Safe for Humans: A Conversation with Siri.” In *Socialbots and Their Friends*. Routledge, 2016.

Hawley, Katherine. “Trust, Distrust and Commitment.” *Noûs* 48, no. 1 (2014): 1–20. https://doi.org/10.1111/nous.12000.

Irpan, Alex, Kanishka Rao, Konstantinos Bousmalis, Chris Harris, Julian Ibarz, and Sergey Levine. “Off-Policy Evaluation via Off-Policy Classification.” *ArXiv:1906.01624 [Cs, Stat]*, November 22, 2019. http://arxiv.org/abs/1906.01624.

Jarrahi, Mohammad Hossein. “Artificial Intelligence and the Future of Work: Human-AI Symbiosis in Organizational Decision Making.” *Business Horizons* 61, no. 4 (July 1, 2018): 577–86. https://doi.org/10.1016/j.bushor.2018.03.007.

Kaplan, Fred. "'WarGames' and Cybersecurity's Debt to a Hollywood Hack." *The New York Times*, February 19, 2016, sec. Movies. https://www.nytimes.com/2016/02/21/movies/wargames-and-cybersecuritys-debt-to-a-hollywood-hack.html.

Kennedy, Carrie H., and Eric A. Zillmer. *Military Psychology, Second Edition: Clinical and Operational Applications*. Guilford Press, 2012.

Kramer, Andrew E., and Andrew Higgins. "In Ukraine, a Malware Expert Who Could Blow the Whistle on Russian Hacking." *The New York Times*, August 16, 2017, sec. World. https://www.nytimes.com/2017/08/16/world/europe/russia-ukraine-malware-hacking-witness.html.

LaGrandeur, Kevin. "How Safe Is Our Reliance on AI, and Should We Regulate It?" *AI and Ethics* 1, no. 2 (May 1, 2021): 93–99. https://doi.org/10.1007/s43681-020-00010-7.

Langner, Ralph. "To Kill a Centrifuge." *The Langner Group*, November 1, 2013. http://www.cs.yale.edu/homes/jf/Langner.pdf.

LaRosa, Emily, and David Danks. "Impacts on Trust of Healthcare AI." In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 210–15. New Orleans LA USA: ACM, 2018. https://doi.org/10.1145/3278721.3278771.

Lewicki, Roy J., Edward C. Tomlinson, and Nicole Gillespie. "Models of Interpersonal Trust Development: Theoretical Approaches, Empirical Evidence, and Future Directions." *Journal of Management* 32, no. 6 (December 1, 2006): 991–1022. https://doi.org/10.1177/0149206306294405.

Libicki, Martin, and Kenneth Geers. *The Cyber War That Wasn't. Cyber War in Perspective: Russian Aggression against Ukraine*, 2015.

Lyons, Joseph B., Sean Mahoney, Kevin T. Wynne, and Mark A. Roebke. "Viewing Machines as Teammates: A Qualitative Study." In *2018 AAAI Spring Symposium Series*, 2018. https://www.aaai.org/ocs/index.php/SSS/SSS18/paper/view/17524.

Lyons, Joseph B., Kevin T. Wynne, Sean Mahoney, and Mark A. Roebke. "Chapter 6 - Trust and Human-Machine Teaming: A Qualitative Study." In *Artificial Intelligence for the Internet of Everything*, edited by William Lawless, Ranjeev Mittu, Donald Sofge, Ira S. Moskowitz, and Stephen Russell, 101–16. Academic Press, 2019. https://doi.org/10.1016/B978-0-12-817636-8.00006-5.

Mach, Merce, Simon Dolan, and Shay Tzafrir. "The Differential Effect of Team Members' Trust on Team Performance: The Mediation Role of Team Cohesion." *Journal of Occupational and Organizational Psychology* 83, no. 3 (2010): 771–94. https://doi.org/10.1348/096317909X473903.

Malone, Thomas. “How Human-Computer ‘Superminds’ Are Redefining the Future of Work.” MIT Sloan Management Review. Accessed August 13, 2022. https://www.proquest.com/openview/a7a1cd959bf50edac3a0991c4213bd41/1?pq-origsite=gscholar&cbl=26142.

McCausland, Phil. “Self-Driving Uber Car That Hit and Killed Woman Did Not Recognize That Pedestrians Jaywalk.” News. NBC News, November 9, 2019. https://www.nbcnews.com/tech/tech-news/self-driving-uber-car-hit-killed-woman-did-not-recognize-n1079281.

McChrystal, Gen Stanley, Tantum Collins, David Silverman, and Chris Fussell. *Team of Teams: New Rules of Engagement for a Complex World*. Penguin, 2015.

Miller, Michael. *The Internet of Things: How Smart TVs, Smart Cars, Smart Homes, and Smart Cities Are Changing the World*. Pearson Education, 2015.

Morgan, Forrest E., Benjamin Boudreaux, Andrew J. Lohn, Mark Ashby, Christian Curriden, Kelly Klima, and Derek Grossman. “Military Applications of Artificial Intelligence: Ethical Concerns in an Uncertain World.” RAND Corporation, April 28, 2020. https://www.rand.org/pubs/research_reports/RR3139-1.html.

Morgan, Steve. “Cybercrime To Cost The World $10.5 Trillion Annually By 2025.” *Cybercrime Magazine* (blog), November 13, 2020. https://cybersecurityventures.com/cybercrime-damages-6-trillion-by-2021/.

Musliner, D.J., J.A. Hendler, A.K. Agrawala, E.H. Durfee, J.K. Strosnider, and C.J. Paul. “The Challenges of Real-Time AI.” *Computer* 28, no. 1 (January 1995): 58–66. https://doi.org/10.1109/2.362628.

O’Kane, Sean. “Self-Driving Shuttle Crashed in Las Vegas Because Manual Controls Were Locked Away.” The Verge, July 11, 2019. https://www.theverge.com/2019/7/11/20690793/self-driving-shuttle-crash-las-vegas-manual-controls-locked-away.

Ososky, Scott, David Schuster, Elizabeth Phillips, and Florian Jentsch. “Building Appropriate Trust in Human-Robot Teams,” n.d., 6.

Owen, Malcom. “Face ID Attention Detection Security Defeated with Glasses and Tape.” AppleInsider, July 8, 2019. https://appleinsider.com/articles/19/08/08/face-id-security-defeated-with-glasses-and-tape.

Qiu, Shilin, Qihe Liu, Shijie Zhou, and Chunjiang Wu. “Review of Artificial Intelligence Adversarial Attack and Defense Technologies.” *Applied Sciences* 9, no. 5 (January 2019): 909. https://doi.org/10.3390/app9050909.

Raghu, Maithra, Alex Irpan, Jacob Andreas, Bobby Kleinberg, Quoc Le, and Jon Kleinberg. “Can Deep Reinforcement Learning Solve Erdos-Selfridge-Spencer Games?” In *International Conference on Machine Learning*, 4238–46. PMLR, 2018. https://proceedings.mlr.press/v80/raghu18a.html.

Roff, Heather M., and David Danks. “‘Trust but Verify’: The Difficulty of Trusting Autonomous Weapons Systems.” *Journal of Military Ethics* 17, no. 1 (January 2, 2018): 2–20. https://doi.org/10.1080/15027570.2018.1481907.

Saenz, Maria Jesus, Elena Revilla, and Cristina Simón. “Designing AI Systems With Human-Machine Teams.” *MIT Sloan Management Review* 61, no. 3 (Spring 2020): 1–5.

Schlimmer, Jeffrey C., and Richard H. Granger. “Incremental Learning from Noisy Data.” *Machine Learning* 1, no. 3 (September 1986): 317–54. https://doi.org/10.1007/BF00116895.

Schneier, Bruce. “Attacking Machine Learning Systems.” *Computer* 53, no. 5 (May 2020): 78–80. https://doi.org/10.1109/MC.2020.2980761.

Seeber, Isabella, Eva Bittner, Robert O. Briggs, Triparna de Vreede, Gert-Jan de Vreede, Aaron Elkins, Ronald Maier, et al. “Machines as Teammates: A Research Agenda on AI in Team Collaboration.” *Information & Management* 57, no. 2 (March 1, 2020): 103174. https://doi.org/10.1016/j.im.2019.103174.

Skinner, Matthew M., Nicholas B. Stephens, Zewdi J. Tsegai, Alexandra C. Foote, N. Huynh Nguyen, Thomas Gross, Dieter H. Pahr, Jean-Jacques Hublin, and Tracy L. Kivell. “Human-like Hand Use in Australopithecus Africanus.” *Science* 347, no. 6220 (January 23, 2015): 395–99. https://doi.org/10.1126/science.1261735.

Snow, Jacob. “Amazon’s Face Recognition Falsely Matched 28 Members of Congress With Mugshots.” American Civil Liberties Union, June 26, 2018. https://www.aclu.org/blog/privacy-technology/surveillance-technologies/amazons-face-recognition-falsely-matched-28.

Stowers, Kimberly, Lisa L. Brady, Christopher MacLellan, Ryan Wohleber, and Eduardo Salas. “Improving Teamwork Competencies in Human-Machine Teams: Perspectives From Team Science.” *Frontiers in Psychology* 12 (May 24, 2021): 590290. https://doi.org/10.3389/fpsyg.2021.590290.

Warner, Michael. “Notes on the Evolution of Computer Security Policy in the US Government, 1965-2003.” *IEEE Annals of the History of Computing* 37, no. 2 (April 2015): 8–18. https://doi.org/10.1109/MAHC.2015.25.

Watkins, Ali. “Obama Team Was Warned in 2014 about Russian Interference.” POLITICO, August 14, 2017. https://www.politico.com/story/2017/08/14/obama-russia-election-interference-241547.

Wheeler, Tarah. “In Cyberwar, There Are No Rules.” *Foreign Policy* (blog). Accessed July 9, 2021. https://foreignpolicy.com/2018/09/12/in-cyberwar-there-are-no-rules-cybersecurity-war-defense/.

Wilson, H James, and Paul R Daugherty. “Collaborative Intelligence: Humans and AI Are Joining Forces,” n.d., 11.

Wolf, Marilyn. “Chapter 8 - Cyber-Physical Systems.” In *High-Performance Embedded Computing (Second Edition)*, 391–413. Boston: Morgan Kaufmann, 2014. https://doi.org/10.1016/B978-0-12-410511-9.00008-3.

Zetter, Kim. *Countdown to Zero Day: Stuxnet and the Launch of the World’s First Digital Weapon*. Crown Publishers, 2014.

Appendix A

Specific Values for Experiment *4.1b*

Initial Likelihood	$P(E_1 \mid H_1)$	0.50
Initial Prior	$P(H_1)$	0.50
Initial Likelihood	$P(E_1 \mid H_2)$	0.50
Initial Prior	$P(H_2)$	0.50
Number of Evidence—E	E	$E_{2,000}$
“Chance the Likelihood Increases”	L	50%
“Chance of Spot Inspection”	S	5%
Likelihood Increases Over Time	x	0.000575
Spot Inspection Confirms Readiness	y	0.495/0.505
Stuxnet Manipulation Value	z	- 0.02

Appendix B

Specific Values for Experiment *4.2b*

Initial Likelihood	$P(E_1 \mid H_1)$	0.50
Initial Prior	$P(H_1)$	0.50
Initial Likelihood	$P(E_1 \mid H_2)$	0.50
Initial Prior	$P(H_2)$	0.50
Number of Evidence—E	E	$E_{2.000}$
Reevaluation Start	R	$E_{1.000}$
“Chance the Likelihood Increases”	L	50%
“Chance of Spot Inspection”	S	5%
Likelihood Increases Over Time	x	0.00055
Spot Inspection Confirms Readiness	y	.495/.505
Stuxnet Manipulation Value	z	- 0.011

Appendix C

Specific Values for Experiment *4.3b*

Initial Likelihood	$P(E_{1} \mid H_{1})$	0.50
Initial Prior	$P(H_{1})$	0.50
Initial Likelihood	$P(E_{1} \mid H_{2})$	0.50
Initial Prior	$P(H_{2})$	0.50
Number of Evidence—E	E	$E_{2.000}$
Reevaluation Start	R	$E_{1.000}$
“Chance the Likelihood Increases”	L	50%
“Chance of Spot Inspection”	S	5%
Likelihood Increases Over Time	x	0.00055
Spot Inspection Confirms Readiness	y	0.495/0.505
Stuxnet Manipulation Value	z	- .011
Stuxnet Softening Value	sx	- 0.01