Book Overview

Arithmetic Is All You Need: The Human Story of Computer Intelligence is the story of how we built machines that think — how four thousand years of making arithmetic faster led to better prediction, and how better prediction eventually enabled our ability to create intelligence. The story is told through vignettes of the most important moments and ideas along this computational arc. And to be very clear: the book is not an attempt at a full survey of the history of computation. The goal is for the reader to *feel* the acceleration of humanity's computational abilities and that they are along for the ride. Each chapter highlights a period when our computational and predictive power increased dramatically; together they form a continuous narrative and each leaves the reader with an understanding of the period's key advancement.

Several through-lines weave through the book. The most prominent is prediction: from the beginning, one of the primary drivers of better arithmetic was the desire to predict the future, especially of human affairs. Astrology was among the earliest applications of mathematical astronomy, and the demand for horoscopes funded centuries of computational progress. We now possess, through the very arithmetic those astrologers helped advance, a genuine ability to predict human behavior — and the consequences of that ability are among the book's central concerns. A second through-line is the recurring appearance of doomsday prophets who warned that some new capability would end the world; they have always been wrong. A third is the question of what it means for a human to write a book at all in an age when machines can generate text on demand. This book answers that question indirectly via a demonstration that a life produces something a machine cannot reconstruct.

The book begins with the problem of retrieving stored knowledge. Chapter 1 opens in ancient Alexandria, where the library held far more written knowledge than any one person could read in a lifetime, and outlines solutions to the problem of finding what you need — from Callimachus's catalog through card catalogs, Yahoo's directory, Google's PageRank, and finally to computer intelligence. The chapter explains how computer intelligence works — they navigate a geometric space of meaning, where every word generated requires billions of arithmetic operations. This establishes the scale of computation that modern AI demands, and poses the question the rest of the book answers: how did we get here?

The answer begins in Babylon. Chapter 2 outlines the origins of mathematical astronomy — how the Babylonians recorded the heavens on clay tablets and predicted their motions, how Ptolemy built a geometric model of the solar system that predicted planetary positions for fifteen centuries, and how the tables he constructed to support that model alleviated one computational bottleneck while creating another that would show up 1500 years later. His work would shape the next two thousand years of intellectual history. Chapter 3 starts at the next bottleneck: Tycho Brahe's observatory generated data at a prodigious rate creating a crushing amount of arithmetic work for those who sought to understand it. The search for a solution — a faster multiplication technique — led to prosthaphaeresis, Bürgi's secret tables, and ultimately Napier's logarithms. Napier's tables would eliminate one bottleneck and create their own — how does one create the tables?

From there the book follows the industrialization of computation. Chapter 4 examines the teams of human computers that mass-produced the tables that navigators, engineers and scientists depended on before pivoting to Babbage's dream of computing them by steam. Chapter 5 examines how technology similar to mechanical adding machines was used to encrypt messages and how cracking those encrypted messages spurred the development of electronic computers.

The second half of the book takes computation into the modern era — GPUs, supercomputers, and the industrial-scale data centers that now consume as much power as small cities — before pivoting to the conceptual breakthrough that turned raw arithmetic into intelligence. Optimization, the subject of Chapter 8, is how we harnessed arithmetic for creation — we started defining problems and letting the machines do the work of finding a solution. When optimization operates at sufficient scale, what emerges is something whose operation we do not fully understand — a computer intelligence that can predict, and generate, language, images, and even the perception of thought.

The final chapters explore the dangers of making humans part of an optimization loop with the machines — why we shouldn't worry about our continued existence, but we should worry what that existence will be like.

Chapter 9 explores how reinforcement learning with verifiable rewards is producing self-improving systems in domains like mathematics and code — and asks what happens in domains without verifiable answers. Chapter 10 argues that the real danger of AI is not extinction but our humanity. The same optimization techniques, deployed through the attention-industrial complex, are training us to be predictable, narrowing our choices, and fragmenting our shared reality. The existential risk crowd, I argue, has identified the wrong threat — not a runaway optimization that destroys humanity, but a local minimum that traps us with and in System 1 thinking.

The book closes with UNLESS — an afterword where I shift to the first person and reflect on my experience in the industry, why I started Adept, and how I see agents powered by computer intelligence as a path away from the attention-industrial complex with — but the window is narrowing. Since so much of the doom narrative is owned by stories, I also tell my own — a story about how arithmetic will help us colonize the heavens and how we might make it a reality.

The book will be approximately 60,000–75,000 words. Three and a half chapters and the front matter are complete; the remaining chapters are outlined in detail. The tone is narrative and accessible, with technical ideas explained through historical context rather than formalism. Extensive footnotes provide a second channel for humor, technical depth, and asides that enrich the narrative without interrupting it. Each chapter stands as its own story while advancing the book's larger argument. The book is designed to work as an audiobook: the main text contains no equations, diagrams are not necessary to follow the argument, and anything that wouldn't survive the transition to audio lives in the footnotes. Specific guidance for audiobook adaptation — such as how epigraphs should be introduced with attribution rather than read cold — will be part of the manuscript.

About the Author

Erich Elsen holds a PhD in Mechanical Engineering from Stanford University, where his thesis was among the earliest work on GPU computing — using graphics processors for general-purpose scientific calculation years before this became an industry standard. He founded Royal Caliber, a GPU computing consultancy, before joining Andrew Ng's Baidu Silicon Valley AI Lab (SVAIL) as one of its earliest members. SVAIL pioneered the use of large GPU clusters for training deep neural networks and was the laboratory that established the computational paradigm now used across the AI industry.

From Baidu, Elsen moved to work with Jeff Dean at Google Brain and then to DeepMind, where he co-led the large-scale model scaling effort. His team trained Gopher, at the time the largest language model ever built, and he co-authored the Chinchilla paper, which established the compute-optimal relationship between model size and training data and reshaped how the industry trains large language models. His earlier research spans GPU kernel optimization, sparse neural networks, neural audio synthesis, and speech recognition. His work has been cited over 29,000 times (Google Scholar) and he holds 23 patents in GPU computing and artificial intelligence.

In 2022, he co-founded Adept, an AI unicorn built on the premise that intelligent agents should work alongside people by using the same software tools humans use and act on their behalf rather than replace them. That vision, and the questions it raised about who these agents ultimately serve, informs the final chapters of this book. He is currently a Principal Research Scientist at Databricks.

Arithmetic Is All You Need is written by its author. Most books about AI are either written by technologists and polished by ghostwriters, or written by journalists who have never trained a model. This book is the rare case of both—someone who has spent two decades inside the systems the book describes, and who can write.

Audience and Platform

The primary audience for *Arithmetic Is All You Need* is the same readership that made Chris Miller's *Chip War*, Walter Isaacson's *The Innovators*, and James Gleick's *The Information* into bestsellers: intellectually curious adults who want to understand how a transformative technology actually works, where it came from, and where it is going, told as a human story rather than a technical manual. The current moment — in which AI systems are reshaping industries, provoking legislation, and generating both excitement and fear — creates urgency for a book that replaces speculation with understanding and history.

A secondary audience includes professionals working in technology, data science, and AI who have deep knowledge of current systems but little sense of the centuries of accumulated effort those systems rest on. The book offers them a richer context for the tools they use daily.

The author does not maintain a public social media presence — a deliberate choice consistent with one of the book's arguments about the attention-industrial complex and the cost of algorithmic curation on human autonomy. His platform is professional rather than public: he is well known within the AI research community, has extensive relationships across the major AI laboratories, and his published work is widely read and cited within the field. He is available for podcast appearances, interviews, and media engagements around publication. Promotional partnerships with science-focused media channels such as Veritasium, whose audience of over 16 million subscribers aligns closely with this book's readership, are a natural fit for the material. Databricks, where he currently works, would support promotion through its substantial conference and media presence, including the annual Data + AI Summit, which draws tens of thousands of attendees from across the technology industry.

Potential endorsers include Ali Ghodsi (CEO of Databricks), Matei Zaharia (CTO of Databricks and creator of Apache Spark), Ashish Vaswani (co-author of "Attention Is All You Need," the paper that introduced the Transformer architecture underlying modern AI), Jensen Huang (CEO of NVIDIA), Demis Hassabis (CEO of Google DeepMind), and Mustafa Suleyman (CEO of Microsoft AI).

Comparable Titles

Arithmetic Is All You Need occupies territory opened by several recent bestsellers while differing from each in focus and argument.

Chris Miller's *Chip War* (2022) demonstrated that a mass audience exists for serious history of computation — it was a *New York Times* bestseller and Financial Times Book of the Year. Miller's book concerns who controls the silicon; this book concerns what the silicon does. Where *Chip War* traces the geopolitics of semiconductor manufacturing, *Arithmetic Is All You Need* traces the intellectual history of computation itself — from Babylonian astronomy to the scaling laws that govern large language models — and argues that each era's computational breakthroughs enabled new forms of prediction that reshaped what humanity could build and understand.

Walter Isaacson's *The Innovators* (2014) covers overlapping territory — the path from Babbage and Lovelace to modern computing — and was a *New York Times* bestseller. But Isaacson is a biographer; his book is organized around personalities. This book is organized around ideas: that computation enables prediction, that prediction enabled computer intelligence, and that the progressive cheapening of arithmetic across four millennia connects ancient astronomy to modern AI.

Mustafa Suleyman's *The Coming Wave* (2023) argues that AI poses existential risks requiring containment of the technology itself. This book shares Suleyman's sense of urgency but focuses on a different danger: not in what computation might do to us, but in what it is already doing — narrowing our choices and optimizing our attention for someone else's benefit. Where *The Coming Wave* looks ahead to catastrophic risk, this book looks at the erosion already underway. Readers who found *The Coming Wave* compelling will find here a complementary argument built from four thousand years of evidence.

James Gleick's *The Information* (2011) is the closest in ambition and register — a sweeping intellectual history that traces information theory from African drums to the internet. *Arithmetic Is All You Need* traces a parallel thread that follows computation, not information, while updating the arc to include the AI era.

Chapter Outline

Chapter 1: As We May Think

In antiquity the Library of Alexandria already contained far more knowledge than any individual could hope to read, remember or assimilate. Humanity's accumulated knowledge totaled maybe 100,000 scrolls thanks to the "wise and useful provision of the ancients to transmit their thoughts to posterity by recording them in treatises," according to the Roman architect Vitruvius. The problem then, as now, was how to find what you were looking for.

Callimachus's Pinakes was essentially a card catalog for the Library — an organizing device that would survive for two thousand years — the author remembers being taught to use one during grade school in the 1980s. Relatively little development in information retrieval happened by the time of Vannevar Bush's 1945 dream of the Memex, a device which would sit on one's desk and help them search through all of their memos, papers, correspondence and libraries. Yahoo, one of the internet's original organizers, started with a recognizably Callimachean organization scheme. But it quickly became impossible to deal with the expanding size of the web; eventually Google mostly solved the problem of finding links to sources of information. But usually what people wanted were *answers*.

The chapter's second half explains how computer intelligence is able to retrieve information from humanity's set of knowledge, and use it to provide answers and how this relies on a vast amount of arithmetic. The key insight: treats words as points in a very high dimensional space where proximity encodes relationships. "Cat" can be close to "Dog" in some dimensions ("cute") while far apart in others ("independence"). Language generation becomes a process of moving in this high dimensional space; and each move requires billions of arithmetic operations.

The conclusion remarks that even if computer intelligence is *only* an incredible retrieval system that gives everyone access to expert human answers on any topic it would fundamentally alter society and the economy; but it has the potential to be even more – to be able to expand our frontier of knowledge. Finally, it setups a question providing a through-line for the next 6 chapters: if producing a single word requires more arithmetic than an individual could do in a lifetime, how did we get to the point where we can do it in fractions of a second?

Chapter 2: Antiquity Computes

It might at first seem unlikely that our first steps towards predicting the world would come from predicting other ones, but that is exactly how the Babylonians took the first steps in humanity's long journey of using arithmetic to predict the future. By recording the positions of heavenly bodies over decades and even centuries onto clay tablets, they were able to work out patterns and near repetitions with periods of decades. Combining these observations with arithmetic allowed them to predict the locations of the heavenly bodies in the sky. They did all this without any understanding of why — they had no model of the system they were observing. Our creation of computer intelligence is in a similar place today: we have powerful predictive tools that can work on any data, and applying those tools to language leads to something that feels like intelligence, but we're missing the "model" evolution has figured out.

The Greeks supplied the planetary model that the Babylonians were missing. The Earth was in the center of the Universe and planets revolved around it on circles circling circles. Their models evolved until Ptolemy wrote his astronomical treatise — the Almagest — which had the final say in the matter for the next 1500 years. Ptolemy's models were intricate and working with them required a great deal of arithmetic and also trigonometry. But trigonometry is useless without knowing the values of trigonometric functions (like sine) for any possible input angle.

Ptolemy included such a table at the beginning of his Almagest and it was likely the largest computational endeavor of humanity up to that point. An explanation of how he created it forms a centerpiece of the chapter. The chapter closes noting that this work would be transmitted around the world, with each civilization adding their flourishes, but it would not be until it returned to Western Europe that the next major leap forward would take place.

Chapter 3: Arithmetic Shines Light on the Heavens

In the late 16th century, a one-in-five-thousand-years run of astronomical luck gave Tycho Brahe the evidence he needed to doubt the established world order. The Church had co-opted Aristotle and Ptolemy — the Earth was the center of the Universe and the heavens were unchanging. But a supernova — the sudden appearance of a new and very bright star — put this in doubt. Tycho set about making measurements and soon discovered that comets too appeared to exist in the heavens, not the atmosphere — the Heavens were not static after all. Anticipating the entire scientific revolution, he decided the solution was to collect data and from the data find the truth.

His observatory produced tens of observations each night, each of which required an hour of arithmetic to reduce into useful form — even with the help of more advanced trigonometric tables. The tables had revealed multiplication as the next computational bottleneck. The author took three and a half minutes to multiply two six-digit numbers, but only ten seconds to add them. The solution was foreshadowed in Archimedes's *The Sand Reckoner*, in which he computes how many grains of sand would be required to fill the Universe. It is one of the only works from which we know of an ancient Greek, Aristarchus, who thought the Earth went around the Sun. Archimedes's idea — that multiplication of numbers written with exponents could be reduced to addition — would be realized as a complete solution by a Swiss polymath clockmaker, Jost Bürgi. But he kept his work secret. Kepler knew of it, but couldn't use it.

It was John Napier who arrived at a similar solution for reducing multiplication to addition – he called them logarithms, the same term we use today. His method was laborious — it took him twenty years to compute his tables vs. a few months for Bürgi — but he gave them away. It was his work that allowed Kepler to unlock the secrets of Tycho's data and discover how the planets really moved — in ellipses. Kepler used his own table of logarithms, Tycho's data and his theory to compile *The Rudolphine Tables*, the most accurate set of planetary predictions yet made.

Chapter 4: The First Computers

Inside the Rudolphine Tables there is an ornate, engraved, fold-out map of the world, as it was then known. Tucked into the corner is a scroll which foreshadows a problem that would consume Empire a hundred years later. It instructs the reader on how to use the measured position of the Moon and the tables to predict their longitude. Accurate navigation was so important to the British Empire that Parliament passed the Longitude Act in 1714 establishing rewards for methods achieving certain levels of accuracy. One of the key instigators of the act, and Isaac Newton's successor at Cambridge, William Whiston, incidentally, calculated the orbit of a comet which he predicted would end the world on October 16, 1736. The end of the world did not come, but more mathematical tables certainly did.

Kepler's tables were not accurate enough for longitude calculation because our Moon's orbit, unlike the planets, cannot be reduced to an idealized two-body orbit. After a century of progress, the Moon's motion was understood well enough to make accurate predictions, but the calculations took a skilled astronomer — Nevil Maskelyne, the Astronomer Royal himself — four hours at sea. To make the method practical, the laborious calculations had to all be done in advance. Such an undertaking was beyond any individual. When Maskelyne published the first tables suitable for the lunar distance method in 1767, it was possible because of a new profession — the computer. Human computers worked together to do the calculations and error checking required to produce these tables and all the others for which demand had exploded: logarithms, trigonometric functions, logarithms of those functions, tides, moon, planets, and more.

Charles Babbage spent considerable energy producing the most accurate logarithm table ever produced before electronic computers. He was frustrated to discover errors in every previous edition — and especially that different editions often reproduced the same errors. He even noted that the errors in a Chinese logarithm table presented to the Royal Society in 1750 contained some of the same errors as a table published in

Europe some 120 years earlier. Babbage's famous wish, upon discovering errors in a table he was checking with another mathematician, was that the work "could be done by steam." He would devote more than a decade to designing and building a machine that could completely automate the production of tables. The chapter focuses on one part of the machine — the carry mechanism, the propagation of overflow from one digit to the next — which was the primary bottleneck limiting the speed of mechanical and electronic calculation, a constraint that would haunt computing machinery for the next two centuries.

Chapter 5: Arithmetic Keeps Secrets

Babbage's machines never got built, but less ambitious adding machines did. Their design was not too different from that of Pascal and Babbage, with rotating gears and a slow, serial carry mechanism. By the end of the 19th century, mechanical calculating machines were everywhere — in offices, shops, and government buildings. Their technology would also transform cryptography. Mechanical machines using similar technology could encrypt messages. But breaking the best mechanical encryption machines would require building something new — electronic computation.

The German encryption machines worked by taking in a sequence of letters to encode and combining each letter with a random number to produce a new "enciphered" letter. The random number was generated by a series of gears; the arrangement of the gears — how they were rotated relative to each other — defined the number. Each new character to be encoded caused some of the gears to rotate, in a manner similar to addition in the adding machines, and produce a new number. If you didn't have enough compute to figure out how the gears were all positioned, then this stream of numbers looked random. If you had enough compute to figure it out, there was no randomness at all.

The encryption machines had a weakness — not all gears rotated with each new letter, which meant patterns from the original message could leak through into the encrypted one. Exploiting these patterns required calculating letter frequencies for thousands of possible gear positions and comparing them with statistical properties of language: how often the same letter appears twice in a row. The comparisons were far too numerous for human computers. The machine, Colossus, built to perform them at Bletchley Park was the first fully electronic computing device and its key component was, in essence, a special light bulb. That everyday object, mass-produced for decades to illuminate homes, turned out to be the component that made electronic computation possible.

Chapter 6: Faster Than Thought

Colossus and its vacuum tubes were only the beginning; the march towards smaller and faster digital switches had begun and it was relentless. In 1947 William Shockley invented the first transistor at Bell Labs and he bettered the design only one year later. In 1965 one of Intel's future founders, Gordon Moore, would coin his eponymous law to describe the industry's pace of progress — that the number of transistors on a chip doubled every year. But if we reframe his law as how much arithmetic a single device can perform in one second, then the exponential stretches back not just to 1965 but across the entire history traced in this book. Exponential growth always invites the question: when does it end?

Each individual technology eventually hits a physical wall, but each time a new technology picks up where the last left off. Thomas Malthus observed in 1798 that population grows exponentially while food production grows linearly, and predicted doom. He was right that contemporary agricultural technology would not continue scaling; he was wrong because he didn't predict the arrival of new agricultural technology. The question is not, "will this curve stop", it's whether another curve is waiting.

One innovation illustrates how the exponential was sustained at the level of individual arithmetic operations. The carry mechanism, the same bottleneck that limited addition in Babbage's Difference Engine, was finally made parallel. Instead of propagating the carry serially from one digit to the next, new circuits computed all the carries simultaneously. This same principle of parallelism then appeared everywhere: within operations, multiple operations at the same time, multiple cores, all the way to hundreds of thousands of chips working together. Each level of parallelism extended the curve when the previous approach hit its limits.

Increasingly all this computation was used for predicting nature. Newton, Maxwell, and their successors had compressed the laws governing how things work into compact equations, but solving those equations for real-world problems required enormous computation. Predicting tomorrow's weather means dividing the atmosphere into millions of cells and stepping the equations of fluid dynamics forward in time — the same principle as the Babylonian System A, but applied to partial differential equations with billions of variables. Recently, neural networks have begun to outperform physics-based simulations at tasks like weather prediction, despite having no knowledge of the underlying physical laws — a development that would have been familiar to the Babylonians, who predicted the heavens without any model of why they moved. The reductionist view of distilling nature into scientific laws may be the best way to "understand" nature, but it may not be the best way to predict it.

Chapter 7: Modernity Computes

Babbage's dream of computing with steam was realized in the 20th century — coal heated water to make steam to spin turbines which turned generators to produce electricity which flowed into computers and through their vacuum tubes to do arithmetic. Now the energy we use to do arithmetic "with steam" is enough to power many small countries.

It requires some of the most sophisticated infrastructure we've ever produced to turn energy into arithmetic at what is now planetary scale. We produce millions of chips, consisting of hundreds of billions of transistors each, with machines that use lasers to heat metal hotter than the Sun to produce light that will etch the circuitry into silicon. All of these chips go into massive data centers where they are all tightly interconnected to each other. The power draw of a single data center can approach that of a mid-sized nuclear power station.

At current levels, arithmetic is still a small contributor to global energy consumption — data centers consume less than half a percent of all energy produced globally. But if the trends outlined in the previous chapters continue, and four or five more orders of magnitude over the next century is consistent with historical trends, then arithmetic becomes by far the largest energy consumer by 2100. At that scale, where the energy comes from *is* existential — the waste heat alone becomes enough to warm our planet; no debate over climate change required — it's just physics. Nuclear and fossil fuels release energy that was not in the Earth's thermal budget; solar and wind merely redirect what the Sun was already providing. A civilization computing at that scale had better be doing it with sunlight.

The next decade will face hard physical constraints to continuing the growth rates of the previous one — rates which were anomalous, driven by the explosive arrival of artificial intelligence. A thousand-fold increase in computational energy would exceed total current global electricity generation many times over. We cannot build new, traditional, infrastructure fast enough to meet this demand. Either the exponential stops or we find new technology that forms the basis for the next phase of growth: fusion power, data centers in space, radical improvements in efficiency — perhaps doing a thousand times more arithmetic per watt by finding approaches closer to the paradigm evolution found in biological brains — or something else entirely. But next, we'll examine what we do with the compute we already have.

Chapter 8: By Gradual Descent

When Ptolemy needed to fit the parameters of his planetary model to his data, he made a guess, checked how well it worked, and then made a better guess until he was satisfied. Kepler did something similar to find the parameters for his ellipses. In the last few centuries, mathematics — in particular calculus — has formalized this ad hoc process into one we call optimization. The result is a meta-program: one program that creates another. Given a loss function that describes how "wrong" a particular guess is, calculus can tell you how to update your parameters to do incrementally better next time. The meta-program follows those rules over and over again and arrives at a solution: for this data and this loss function, these are the best parameters. Each step takes arithmetic, and the more parameters, the more arithmetic it requires.

Creating a computer intelligence consists of two components — parameters, that is numbers, and a traditional computer program that describes exactly how arithmetic combines those parameters with the input to produce an output. There can be trillions of parameters that require 1,000,000,000,000,000,000,000,000,000 arithmetic operations to find the optimal ones. The meta-program, the traditional code, we understand. The loss function we understand — predict the next word. We understand the underlying mathematical rules. But we don't entirely understand how the optimization or the computer intelligence it produces works. The optimization process is like navigating the hardest maze ever constructed in trillions of dimensions all at once — theory says it should get trapped, finding mediocre solutions with no path to better ones. In practice, it doesn't. The maze turns out to be surprisingly forgiving: good solutions are everywhere and connected to each other. Right now we are in the pre-thermodynamics era of making steam engines. We've figured out a lot of tricks that make it work, but we don't really have a theory that explains it all.

Predicting the next word of human-authored text is only the first step. To make these systems useful and produce output we like, a further step places humans directly in the optimization loop — we optimize for what humans prefer. Human raters compare outputs and choose which they prefer, and the optimization adjusts the computer intelligence accordingly. Because it is framed as a preference amongst options, the options presented already constrain the evolution of the intelligence. And its constrained evolution influences our preferences and even the options open to us. More problematic — these preference judgements are made quickly — raters make fast, intuitive assessments reflecting System 1 thinking, rather than slow, careful reasoning. This is how computer intelligence learns to produce writing that feels good at first and falls apart under scrutiny. We are optimizing for computer intelligence which produces outputs preferred by our ancient System 1 reaction system — and that is exactly what we get.

But there is a second optimization objective, one independent of human preferences entirely. If computer intelligence one day surpasses us, it will be because of this one.

Chapter 9: A Player of Games

The 2016 defeat of Lee Sedol by AlphaGo in the ancient game of Go was evidence to some that computers would soon eclipse humans at everything. AlphaZero, its successor, could learn to play chess and Go just by playing itself — starting by making completely random moves — it only needed the rules. But board games are a poor imitation of life. AlphaZero worked by playing itself until one side won and then nudging the parameters of its model to make winning moves more likely. The way AlphaZero learns will help in some domains, but it's unclear how broadly it can be applied.

In code or mathematics we can define the equivalent of "winning" at a board game — the code passes tests, the math produces a desired answer or formally verified proof. In general, any domain where it is much easier to verify than to produce might be amenable to this technique. We call such situations ones with verifiable rewards. An amateur chess player can know who won a game between grandmasters without understanding most of the moves.

AlphaZero worked because two players making random chess moves eventually results in someone winning and someone losing. But for most coding or math problems, generating random text will never solve the problem. The search space is just too vast. But computer intelligence that's been trained to predict human text and preferences — it can sometimes produce solutions, and once is all that's necessary for progress. We can train the computer intelligences to solve progressively harder and harder problems in this way — at first they infrequently produce solutions to easy problems, but we use those solutions to train the model to get better at solving problems. Now it infrequently produces solutions to medium problems, and so on. It's also possible to teach the computer intelligence to run a search procedure without requiring retraining — it can use its memory of past failed attempts to inform more likely future solutions.

Optimizing these systems to match our preferences is bound by us, but training them against verifiable rewards has no ceiling. The problems can keep getting harder, the opposing player better, the codebase gnarlier. Within hours AlphaZero went from making random moves to being better than any human who ever lived. Systems that can reliably write code or do mathematics better than humans are going to change the world. But how will they choose which problems to solve? That is not a question with a verifiable answer.

Chapter 10: We May Yet Live

(This chapter contains two full-length sections under a single numeral, connected by their sub-titles: "Vain Wisdom All, and False Philosophy" and "And Still Be Dead.")

The first half, "Vain Wisdom All, and False Philosophy," opens by asking why fear is always the first response to new power. Prophets peddling wares of an "AI apocalypse" grew like weeds immediately after the tree of compute grew. Unlike the wares of the prophets that form the through-line of the book, the doomsayer has evolved to peddle with the language of mathematics. No longer will the world end on such and such a date, now it is "I predict 10% chance of AI doom over the next 5 years." They've learned it has the same ability to command attention with none of the downside of actually being wrong. A prediction of a probability sounds mathematical and scientific; often there's even lots of mathematical reasoning behind each number. But these predictions are not science.

As Carl Sagan once said, "Science is a way to call the bluff of those who only pretend to knowledge. It is a bulwark against mysticism, against superstition, against religion misapplied to where it has no business being." The falsifiable claim is the bedrock of science and these probability of doom predictions are not falsifiable. If doom happens – how do we know the prediction of 10% was correct? It happened, so maybe it was much too low. If it doesn't happen – how do we know the prediction of 10% was correct? The right probability might have been 99% and we just got lucky. We cannot, unfortunately, reset the universe and try again. All such statements like this are utter hogwash and we should call their bluff. The chapter uses Milton to name such philosophy for what it is: "vain wisdom all and false philosophy."

The second half, "And Still Be Dead", says that our existence may not be threatened, but our humanity is under siege. Algorithms are optimizing *us* and have been for decades – and the goal of the optimization is not our benefit, it is someone else's. Technology has always created feedback loops, but until recently they operated at the level of society and on time scales measured in decades or centuries: the plow, steam, even cars. Now an optimization loop *of us* is the goal and it operates on timescales measured in seconds. Algorithms predict what we will click, watch, read, and buy, then serve us more of it. Our control over the shape of our lives is eroded by our very ability to use arithmetic to predict ourselves – machines use our past choices to narrow our current options – to trap us in local minimum from which we cannot escape. Our connection to a shared conversation is frayed when everyone goes to the Library of Babel and reads a different book.

This all existed before computer intelligence, but computer intelligence opens new front in the war for our freedom and attention. Prior incarnations were spies: watching from the shadows, gathering intel about our habits, desires, wealth, and secrets; but the spies rarely interacted with us directly, the feedback loop was buffered by intermediaries which slowed it down. Now the spies have become undercover agents – we talk with them directly while they predict and answer us but also redirect us to serve their own ends. If you're interested, there's one **really** important technique they use to gain your confidence and hide their suggestions that serve their interests and not yours. Tell me if you'd like to learn about it.

Finally, the chapter closes by noting that slower feedback loops have existed for much longer in the insurance industry. Predicting that the poorest people in a certain zipcode should pay more for car and health insurance because they are the highest risk can make those outcomes even *more* likely. There we asked a question: "which predictions do we, society, want to make and what inputs do we want to make them with?" Now we can make predictions about almost anything with panopticon levels of surveillance on every detail of our lives. Do we want to allow it? These are value judgements that only society can make.

Afterword: UNLESS Summary

One of my motivations in starting Adept was the hope that agents powered by computer intelligence could fundamentally alter the attention-industrial complex. They could be our Odysseus tied to the mast — visiting the isles of the web whilst being bombarded with siren calls of advertising, immune and never wavering from their pursuit of our goals.

Agents powered by computer intelligence are transforming how programming, software engineering and even research is done. It is already routine to spend ten percent or more of a worker's cost on the arithmetic powering their agents. In the future, it may even make sense to pay multiples of the human cost for their pack of agents. Because the standard in this realm is to pay directly for agents, there's a chance to change the way the attention-industrial complex functions. Individuals can pay for their agents directly and then direct them to do things on the web for them: answer this question, find me relevant news to read on the way to work, recommend some new music, research and book a trip. Knowing our interests, preferences and goals, the agents can tirelessly seek out the best options for us — while avoiding whatever the attention-industrial complex tries to distract them with.

The response of the complex will be twofold: first, to capture the agents so that they do not truly work for just us, but also for the complex; second, to make it impossible for the agents to interact with the web on our behalf. Preventing agent capture requires that agent developers remain independent from the complex — and there are already very few options here. Google, OpenAI, and Anthropic are the makers of by far the best three computer intelligence systems. Google created the complex. OpenAI has plans to introduce advertising and join the complex soon. That leaves only Anthropic as independent of the complex and capable of potentially disrupting it. Anthropic, founded as a public benefit corporation, also has the corporate charter and ideological mindset to pursue a path out of the current local minimum to one more beneficial to humanity. They were founded partially because of the fear of existential risk, and they may yet be what keeps us alive.

Arithmetic Is All You Need

The Human Story of Computer Intelligence

Erich Elsen

Contents

1 As We May Think **1**

2 Antiquity Computes **11**

3 Arithmetic Shines Light on the Heavens **21**

Chapter 1

As We May Think

> "Consider a future device … in which an individual stores all his books, records, and communications, and which is mechanized so that it may be consulted with exceeding speed and flexibility." — Vannevar Bush, "As We May Think" (1945)

> "Every one knew how laborious the usual method is of attaining to arts and sciences; whereas, by this contrivance, the most ignorant person, at a reasonable charge, and with a little bodily labour, might write books in philosophy, poetry, politics, laws, mathematics, and theology, without the least assistance from genius or study." — Jonathan Swift, *Gulliver's Travels* (1726)

Computer intelligence is the culmination of a long human project: the collection and organization of our knowledge. It has always been distributed and the bottleneck has always been how effectively any individual could access what was already known.

Even in antiquity, human knowledge was a collective enterprise spanning cultures and generations: no individual could absorb it in full, and civilization advanced by making accumulated knowledge accessible. The Library of Alexandria is said to have held up to 100,000 papyrus scrolls, far more written knowledge than any one person could hope to read in a lifetime.

The Roman architect Vitruvius, writing in the first century BC, understood the debt this implied. "It was a wise and useful provision of the ancients to transmit their thoughts to posterity by recording them in treatises," he wrote, "so that they should not be lost, but, being developed in succeeding generations through publication in books, should gradually attain in later times, to the highest refinement of learning. And so the ancients deserve no ordinary, but unending thanks, because they did not pass on in envious silence, but took care that their ideas of every kind should be transmitted to the future in their writings." The act of writing things down and refusing envious silence was the beginning. But it created its own problem: once knowledge was recorded at a scale beyond any individual's capacity to read, let alone remember, how would anyone find the specific piece they needed?

Consider the problem as it would have presented itself in practice thousands of years ago. A small collection of a few dozen scrolls needs no organizational system at all — the librarian knows every work personally — where it is stored, and what it contains. A scholar arriving with a question can simply ask, and the librarian can retrieve the relevant text and its location in moments.

But this doesn't scale very far; at a hundred scrolls, a diligent librarian might still manage, though they may need to pause and think. At a thousand, memory alone is no longer sufficient. No one person can remember the contents of a thousand works with enough detail to reliably match a specific question to a specific passage. The librarian may remember that a certain subject was treated somewhere, but not in which scroll, or may confuse two works that treated similar topics differently. At ten thousand scrolls, human memory is an island in a vast ocean. Without some external system of organization, knowledge that exists in the library is the tree in the proverbial forest — it might as well not exist at all.

The early librarians at Alexandria solved this by organizing the library's holdings physically: scrolls were stored in rooms or alcoves grouped by subject, with tags attached to the end of each scroll identifying the author and title. Within a subject area, works were sometimes arranged alphabetically by author — much better than memory alone, but far from perfect. A work that touched on multiple subjects could only be stored in one place. And not all questions mapped obviously to a particular category. As the collection grew, the subject groupings themselves became unwieldy: "philosophy" might encompass thousands of scrolls spanning logic, ethics, natural philosophy, and metaphysics, with no further subdivision to guide the searcher.

Callimachus, a scholar and poet who worked at the library in the third century BC, recognized that physical organization alone could never solve the problem at this scale. Rather than relying on the arrangement of the scrolls themselves, he created a separate work, the Pinakes, a catalog of the entire collection. The Pinakes, reportedly 120 scrolls in its own right, listed the library's holdings by subject and author, noted the opening lines of works, provided biographical information about writers, and grouped everything into categories such as rhetoric, law, epic poetry, and medicine. It was humanity's first large-scale bibliographic database: a system of metadata that existed apart from the collection it described. A scholar no longer needed to browse the shelves or rely on a librarian's memory; they could consult the Pinakes to learn what the library held on a given topic and where to find it.

It was an extraordinary achievement, but the core problem that would persist for two millennia remained: even if knowledge existed somewhere, finding a specific piece of it was slow and uncertain. The Pinakes could tell you that a work existed and roughly where to look for it, but it could not tell you whether that work contained the particular fact or argument you needed. One would navigate to a subject area, browse promising titles, retrieve scrolls, and read substantial portions to see whether they contained what one was looking for. The process was measured in hours and days.

Difficulty in searching was a serious problem, but not without its own benefits. By browsing, one is exposed to ideas not explicitly sought. Discovering a new world is as exciting in a library as sighting land from a ship at sea.

For nearly two thousand years, this basic pattern barely changed. Unable to figure out how to make searching scale further, we found a different way to cope with ever accumulating knowledge. Each generation summarized and condensed the knowledge of the last in encyclopedias and textbooks that compressed centuries of learning into single volumes. And much was simply forgotten — works that were wrong, or superseded, or no longer relevant quietly dropped out of circulation. The number of books in a typical library stayed roughly comparable to the number of scrolls in Alexandria not because knowledge stopped growing, but because it was continually being re-summarized and culled.

Across two millennia tools had improved only modestly with the introduction of indices, card catalogs, and the Dewey Decimal System. Information finding remained slow and manual. As a child in the 1980s, the author was still taught to use card catalogs as the primary interface to human knowledge. You approached a wall of small wooden drawers, each labeled with a range of letters, pulled open the drawer and flipped through index cards scanning titles and subject headings. If you were lucky, you found a potentially useful book and noted the call number. You then walked to the shelves, found the right section, and browsed spines until you found your book. If it had been checked out, or didn't have what you needed, you tried browsing nearby books or started all over. The entire process, from question to answer, might take an hour for a simple search. For a serious research question, it could take days or weeks, and success was never guaranteed.

The frustration with the sheer friction of finding what you knew must exist somewhere led Vannevar Bush, in 1945, to imagine the device described in this chapter's epigraph. Bush called it the Memex: a desk-sized machine in which a researcher could store all of his books, records, and correspondence on microfilm, and retrieve any item by a system of codes and cross-references. The Memex was never built, but Bush had identified a problem that was about to get much worse: the volume of knowledge was beginning to grow so fast that summarization and curation could no longer hold the tide.

The real inflection came with computers and the internet. Early web directories, typified by Yahoo's hierarchical taxonomy, were recognizably Callimachean. Yahoo did not search the web so much as organize it: human editors placed websites into nested categories such as Science → Astronomy → Observatories or Arts → Literature → Poetry. To find information, users navigated these branches, browsing downward in the hope that the right page had been filed in the right place.

Using Yahoo in, say, 1996 was an experience closer to browsing a library shelf than to what we now think of as searching the internet. You arrived at a page of broad categories. You clicked on one — say, "Science" — and were presented with subcategories: Biology, Chemistry, Physics, Astronomy. You clicked again, and again, each time narrowing the scope but also committing to a path. If what you were looking for had been filed under a different branch, you might never find it. A page about the chemistry of interstellar dust might appear under Astronomy or under Chemistry, but probably not both, and the choice was made not by any algorithm but by a human editor applying judgment. This worked when the web was new—when a team of editors could plausibly visit and categorize most sites of consequence. But quickly, the system collapsed under its own success, and for exactly the reason Callimachus would have recognized: the knowledge grew faster than the catalog.

Full-text search engines such as AltaVista introduced a radically different idea: you could search directly for words. This was revolutionary, but brittle. Queries required choosing the right terms carefully and in advance; ambiguity was punishing. Searching for "apple" returned orchards and computers indiscriminately while "java" yielded islands and programming languages alike. A medical researcher searching for "treatment resistant depression" might find clinical papers, pharmaceutical advertisements, patient forums, and self-help blogs, all jumbled together with no way to distinguish authority from anecdote. The system retrieved pages that contained the right words, but it had no understanding of what those words meant or which sources were trustworthy. The serendipity of library browsing, of stumbling on related ideas nearby on the shelf, was replaced by something much less useful: stumbling over anything that happened to share a common word.

Advertising and search engine optimization made the problem worse. As incentives shifted, pages were increasingly written not for humans but for algorithms, polluting results with keyword-stuffed irrelevance. The web was becoming a library in which many of the books were written to trick the catalog into placing them on the most prominent shelf.

Google exploited a unique property of the web — hyperlinks — to mostly solve this problem. By analyzing how pages referenced one another, PageRank inferred which sources were likely to be valuable. A page that was linked to by many other pages was probably important itself. This applied the idea of academic citations — highly cited papers are more likely to be relevant — at much larger scale. But it was more subtle than mere vote-counting. A link from a highly ranked page counted for more than a link from an obscure one, creating a recursive definition of authority that could only be resolved by computing across the web's entire link graph simultaneously. The result was a ranking system that, for the first time, could distinguish between a personal homepage mentioning a topic in passing and an authoritative reference work treating it in depth—even when both pages contained the same keywords.

For two decades, this approach defined the state of the art. If relevant information existed somewhere on the internet, Google could usually point you toward it. The combination of full-text search with link-based ranking created a system that could handle both the scale of the web and the ambiguity of natural language queries with reasonable success. Two generations learned to begin almost any intellectual task by typing a few words into a search box; the interface was so successful that it became invisible, you simply expected it to work. It became a default and we stopped noticing what was still missing.

What was missing was the answer itself. Search could point you in the right direction, but the information you needed was rarely presented exactly in the form you needed it. You still had to read and interpret material spread across multiple sources. If you were trying to understand how a system worked or diagnose a problem search could direct you to relevant pages, but the work of actually using that information remained manual. Someone trying to get an old printer working with a new laptop would find a manufacturer's support page that hadn't been updated in three years, a forum thread where the solution was marked "resolved" with no explanation, a driver download page for a different operating system, and a YouTube video that covered a similar but not identical model. Bringing those fragments together into an actual solution remained entirely the user's responsibility.

Large language models represent the next, and possibly final, evolution of information finding. They can still retrieve documents, but the important difference is what happens next: rather than handing the user a list of sources, they construct answers. We compress an enormous fraction of humanity's written knowledge into these language models and equip them to search for what they might be missing. The resulting computer intelligence can synthesize knowledge on demand, tailored to your context and intent. Ask a question about contract law, or the chemistry of bread-making, or the best way to debug a memory leak — provide the specifics of your situation — and the system generates an answer tailored to your circumstances, drawing on patterns learned from the billions of texts on which it was trained.

This is a better way to find information, but one that fundamentally breaks the tradition of knowing where knowledge came from. When answering factual questions, responses arrive without the sources that let us evaluate them, and can be confidently wrong in ways a list of references cannot. But it is also a fundamentally different interface, one that opens possibilities search never could. You can ask the system to write code, draft a document, or even delegate tasks entirely, letting the system act on your behalf. None of this was conceivable when the interface was a search box returning a list of links. For much of what these systems now do — writing, coding, reasoning through problems — the concept of a source doesn't even apply. Their power comes from the feedback loops they create between humans and computer intelligences; those loops expand what is possible for us to accomplish, but the computer intelligences also now shape us in return. There is no free lunch.

How were these systems built? The first generation learned by reading the existing internet. The models were fed vast quantities of text[1] — books, articles, forums, encyclopedias, technical documentation, conversations — and learned patterns of language by predicting, word by word, what came next. The resulting systems could generate fluent, knowledgeable text across an extraordinary range of subjects, because they had absorbed the statistical structure of how humans write about those subjects.

But very quickly it became clear, based on user interactions, that much valuable knowledge was still missing. It was embedded in interactions that had never before been worth recording in a widely searchable medium. A senior engineer's instinct for debugging, a teacher's sense of which explanation will land with a struggling student, a doctor's clinical judgment built from thousands of patient

[1]The text used to train these models was not always obtained with the consent of its authors. Large-scale data collection swept up (and even intentionally targeted) copyrighted works often without permission or compensation. The legal and ethical questions surrounding this practice remain unresolved at the time of writing. It is also unclear how to properly compensate sources whose knowledge is disintermediated. The standard advertising model of the internet will not work here — perhaps we can find something better.

encounters — the general principles behind these skills had been written down in textbooks, but actual transcripts of their application in specific situations had not. So they were created. Companies hired large numbers of people to produce exactly the kinds of documents that hadn't existed before: written demonstrations of expert judgment applied to specific situations.

Building the most capable models became as much a problem of data collection as of raw computation; the data collection required the coordinated work of millions of people and the expenditure of billions of dollars, making it one of the largest deliberate attempts ever undertaken to render tacit human knowledge explicit and reusable by machines. It was, in a sense, the largest bibliographic project since Alexandria — except that instead of cataloging existing texts, it was creating new ones, extracting knowledge that had previously existed only in human heads and human habits.

The result is something that feels like intelligence, but it's strange and savant-like: immensely impressive, shockingly fallible, possessing vast knowledge and instant recall, but little grounding. They are extraordinarily good at reorganizing what humanity already knows, and conspicuously bad in ways humans are not.

The crucial step was to make words, and eventually ideas, amenable to arithmetic. Once language could be represented numerically, systems built to manipulate numbers could begin to operate over meaning as well as symbols.

This was not the first attempt to formalize language. For decades, researchers in artificial intelligence had tried to represent meaning using rules and logic, so-called "expert systems": hand-crafted dictionaries that defined words in terms of other words, grammar systems that parsed sentences into tree structures, knowledge bases that encoded facts as relationships between labeled concepts. These approaches worked in narrow domains like airline reservation systems but they were brittle. Every new domain required new rules, and the rules interacted in ways that were difficult to predict. Natural language, with its ambiguity, context-dependence, and sheer volume, defeated every attempt to contain it in a rule-based framework.

The breakthrough came from a different direction entirely. The insight, anticipated by the linguist J.R. Firth's famous observation that "you shall know a word by the company it keeps," was that meaning could be inferred from context. Words that appeared in similar contexts probably meant similar things. If "dog" and "cat" appeared in many of the same sentences — near words like "pet," "veterinarian," "fed" — then they were probably related in meaning. This was not a definition of meaning in the philosophical sense, but it was something more useful: a way to measure it.

If we associate words with numbers, then we can have a computer program work through large volumes of text and optimize the assignment of numbers to words so that nearby words can be predicted from each other. (We'll discuss this process in more detail later.) At first we might try to associate each word with a single number: "dog" $\rightarrow$ 3.1, "leopard" $\rightarrow$ 2.7, "wolf" $\rightarrow$ 1.6.

But this immediately breaks down. Words are related to one another in too many different ways for a single number to capture. A dog may be closest to a wolf in an evolutionary sense, while a leopard may be closer to a wolf in terms of behavior or habitat. Dogs are domesticated; wolves and leopards are not. Leopards and wolves may be perceived as more dangerous than dogs. Dog can also be a verb, not just a noun. These relationships shift across contexts, uses, and even languages.

To represent all of this, we need more than one number per word.

The solution is to represent each word not as a number, but as a point—a position in a space with many dimensions. Distance in that space encodes similarity: words that are used in similar ways, or that share related meanings, end up near one another. Words that are similar in some ways, like tomato and apple (red and round), might be close in some dimensions but far apart in others (like taste).

To visualize the idea, imagine placing words within a three-dimensional space, like rooms in a house. Already we can do much more than with a single number. "Dog" and "wolf" might be close vertically, while "wolf" and "leopard" might be close horizontally. Notice that in two and three dimensions there are already more than two and three types of relationships that are possible.

To see why, consider a simplification: along each dimension, two words can be either "near" or "far". In two dimensions that gives four possible relationships: (near, near), (near, far), (far, near) and (far, far). In three dimensions this expands to eight. As the number of dimensions increases, the number of possible relationships expands *very* fast. Modern computer intelligences use thousands or even tens of thousands of dimensions, allowing an enormous number of relationships to coexist simultaneously.

Our intuition is tempted to label these dimensions: this one for size, that one for danger, another for cuteness and so on, but the instinct is misguided. The dimensions themselves do not correspond neatly to human concepts, they are simply degrees of freedom the system uses to arrange words so that useful relationships emerge. Meaning lives in the geometry of the whole which is not at all intuitive, and difficult for us to understand even after years of dedicated research.

Once words are represented this way, as points, language generation becomes a matter of movement through the space. Starting from a point corresponding to a word or phrase, the system repeatedly transforms that point — nudging it in directions suggested by context, grammar, and prior usage. Each transformation is millions, perhaps billions of numerical operations: additions and multiplications applied across many dimensions at once.

After enough of these steps, the point arrives in a region of the space near words that would plausibly come next. The system then selects among those nearby words, favoring closer ones but retaining some variability. For computer intelligence, language emerges as a result of navigating a high-dimensional landscape shaped by prior text.

Representing words, ideas, and meaning as points in space was a truly revolutionary shift. Rarely in intellectual history has a single abstraction collapsed many previously separate phenomena into one: Newton showed that the motion of the heavens, revealed by Kepler, and the motion of falling objects, measured by Galileo, obeyed the same laws; Darwin showed that the diversity of life could be understood as variations of a single process unfolding over time. The same is true here: a single abstraction unifies a wide range of problems that once appeared unrelated.

Concepts that share no words in common can now be recognized as related. The point in meaning space representing "food keeps getting stuck in my throat and it feels like it won't go down" would find itself near a cluster of related conditions — esophageal stricture, eosinophilic esophagitis, achalasia — in the geometry of meaning, they live in the same neighborhood. This is what it means to represent knowledge geometrically: relationships that were previously implicit, requiring a trained human mind to recognize, become explicit in the structure of the space itself.

Translation, for example, emerges naturally once words from different languages are embedded within a common meaning space. A sentence is mapped into that space, where its content, its meaning, is expressed independently of any particular language. From there, it can be rendered back into words under the constraints of another language. What had been an enormously difficult problem for computers becomes a geometric operation: move from one region of the space (the source language) to a nearby region (the target language) where the same meaning is expressed differently.

Images can be treated in much the same way. Visual patterns are mapped into the same shared meaning space as language. Describing an image then becomes a matter of referencing the points representing the image, and rendering them into words. Generating an image reverses the process: take points representing the words, move to the points describing the image, and finally generate the pixels.

The same principle extends to speech, audio, video, and other modalities. What once required distinct systems, features, and hand-crafted pipelines now reduces to a single operation of navigating a shared space of meaning. But navigating this meaning space is computationally demanding; each step of generation is an expensive transformation applied across thousands of dimensions, and even a single such transformation involves millions of additions and multiplications.

Producing a word requires repeating this process dozens, even hundreds of times. The price of unification is compute.

We can pay this price only because of the long accumulation of computational capacity humanity has built over millennia. For most of history, arithmetic was performed by individuals and was therefore slow and expensive. Then mechanical calculators, electronic computers, and industrial-scale infrastructure gradually compressed more and more computation into smaller amounts of time. That a machine can now perform, in a fraction of a second, the amount of arithmetic required to traverse a space of meaning at this scale is the culmination of a civilizational project to make computation abundant — one whose full history, and full cost, the following chapters will outline.

And yet an open question remains: can a system that navigates a space shaped by existing knowledge move beyond retrieval to discovery? Some of the greatest scientific advances have not required any new data, but only a reorganization of what was already known. Given Tycho Brahe's observations of planetary motion, Johannes Kepler discovered his empirical laws that predicted the heavens with unprecedented accuracy. Then, less than a century later, Newton required no new data beyond that available to Kepler to create a new framework — his laws of motion and universal gravitation — that explained how Kepler's empirical laws worked.

What Kepler and Newton did was extrapolation, but what experts mostly do and what generally makes expertise valuable is interpolation: combining known facts, methods, and patterns in ways suited to a new situation. A doctor diagnosing a patient and an engineer sizing a beam — they each draw on established knowledge applied to particular circumstances. Nearly every act of professional judgment is an interpolation: recognizing which known patterns apply here, and adapting them to the specifics at hand; acts of true extrapolation are rare and even undesirable in most jobs. A system that could reliably interpolate across the full breadth of human knowledge — combining insights from medicine with engineering, or from law with economics, at a speed and scale no individual could match — would already be transformative, even if it never produced a single genuinely new idea.

Whether computer intelligence can make a genuinely extrapolative leap remains an open question — one we'll return to later. What we can answer is a prior question: how did we get here? The arithmetic required to produce a single sentence of machine-generated text would have consumed the working lives of a city of human calculators. That it now happens in a fraction of a second is not magic, but history.

Chapter 2

Antiquity Computes

> "I conceive that these things, King Gelon, will appear incredible to the great majority of people who have not studied mathematics, but that to those who are conversant therewith and have given thought to the question of the distances and sizes of the earth, the sun and moon and the whole universe, the proof will carry conviction." — Archimedes, *The Sand Reckoner* (c. 250 BC), translated by Thomas L. Heath

It is, in many ways, surprising that humanity's first successful systematic predictions about our world were predictions of other ones. The solar system and the stars are incongruous with anything encountered in daily life — yet there we made our first steps towards prediction and not closer to home.

Their size, distance and relatively small number are exactly the things which made them targets for our desire to predict. Absent the influence of the heavens themselves, things on Earth rarely repeat. The wind never blows quite the same way twice. And even relatively repeatable things are hard to quantify. A stone falls faster than our ability to time it with any accuracy without modern timepieces[1]. But the sun, the moon, the stars and the planets are always there on a leisurely walk across the sky waiting to be measured.

Perhaps equally surprising to us is that for thousands of years the prediction of the movements of celestial bodies was intertwined with the prediction of Earthly events — horoscopes and fortunes. Astronomy and astrology were born together, one in service of the other. We used the one thing we *could* predict to make predictions about the things we *wanted* to predict, but could not.

The Babylonians were the first people to systematically observe and record their observations of the night sky. And they needed arithmetic to make sense of their

[1]Galileo's insight to put falling things on a very shallow ramp so that they fall slowly enough he could time them with his heartbeat was a brilliant insight thousands of years in the making. We should not be hard on the ancients for not thinking of it.

observations. Before the adoption of Arabic numerals, Roman numerals (I-one, V-five, X-ten, L-fifty, …) were the standard way to write numbers in Europe. One problem with them is that writing nearby numbers can require greatly different numbers of letters. It is hard to parse the scale of the number from its representation: 3888 = MMMDCCCLXXXVIII whereas 3900 is MMMCM. Another problem with them is that they make calculation clunky. Addition is not too bad, you can mostly just merge the symbols from the two numbers together.

XLVII + XXIX = 47 + 29
~~X~~LVII + ~~X~~XIX cancel the subtracted X in 47 with an X in 29
LVII + XIX = 57 + 19
LVI~~I~~ + X~~I~~X do the same for the subtracted one in 29
LVI + XX = 56 + 20 merge all the remaining symbols together
LXXVI = 76

But multiplication is really cumbersome; it's not impossible to use something like long multiplication with Roman numerals[2], but it's error prone and inefficient. The Romans worked around this by not writing down their arithmetic like we do today; they used an abacus or other counting device, did their sums and then just wrote the answer down using Roman numerals.

An abacus has some advantages — it is certainly faster for adding many numbers compared with writing things out. But one big disadvantage is that you have no record of your work; writing things down allows you to see all of your intermediate calculations and verify them individually and if there was a mistake, to correct it and only redo the portion that comes afterwards. An abacus in contrast is destructive — intermediate results are erased by the process of using it.

It would be natural to think that as the Babylonians were working a millennium

[2] Here is worked example of long multiplication with Roman numerals. You need to expand everything and then figure out how to simplify. Knowing a "Roman multiplication table" would make things a bit faster (e.g. VX=L, VC=D, …).

```
XIII   13
 VII    7
=========
   XIII
+  XIII
+ XXXXXVVV simplify XXXXX -> L and VV -> X
==========
   XIII 13
   XIII 13
+ LXV   65
===========
LXXXVIIIIII  simplify IIIII -> VI
LXXXVVI      simplify VV -> X
LXXXXI       simplify LXXXX -> XC
XCI     91
```

before the Romans, they were using an even more primitive number system – but the opposite was true. The Babylonian number system was much closer to our own than to the Romans – it was a place value system that facilitated multiplication; the main differences were that they lacked a decimal point and used base 60 instead of base 10. The lack of a decimal point meant that a number like 1234 could be interpreted as 1.234 or 12.34 or 123.4 and so on; which one needed to be inferred from the context. Their base of 60 is what gives their number system its name — the sexagesimal system. Unlike our 10 digits, they had sixty different symbols to represent each number from 1 to 59[3]. And their multiplication table had 1,770 entries instead of 45.

With a multiplication table so large, memorization would be difficult, so it is not surprising that multiplication tables are a common occurrence on the Babylonian clay tablets archaeologists have found. We have also found tables for reciprocals, squares, square roots, and even Pythagorean triples; lookup tables were used extensively to reduce the amount of work in solving common problems. Their tables and their number system made serious arithmetic possible — and that arithmetic, applied to data recorded over decades and even centuries, is what enabled prediction.

The basis of all Babylonian prediction was discovering repetitive patterns in the positions of the Sun, Moon, and planets. Consider the Sun — its position in the Zodiac, that is, which stars appear behind it[4], repeats after 1 year — in fact, the Sun's position repeating is essentially the *definition* of a year. The planets, in contrast, all take much longer than a year to repeat their positions in the Zodiac. From a modern point of view: if the Earth orbits the Sun with a period of 365 days and Venus orbits the Sun with a period of 225 days, then after 8 years Earth will have completed 8 revolutions and Venus will have done just shy of 13. That means that Venus will appear in almost the exact same place in the sky every 8 years. Even without understanding the mechanism, it is possible to observe these repetitions.

Although the Babylonians did not make plots — they dealt entirely with tabulated numbers — plotting their data is illustrative of what they were doing. If we plot the position of the Sun against days, then it forms a nearly straight line going from 0 degrees to 360 degrees over the course of 365.25 days. Then it starts all over again. This happens because Earth's orbit is *almost* circular, so that

[3]The symbols themselves were actually constructed a bit like Roman numerals, so it wasn't quite as bad as having to remember sixty completely different glyphs.

[4]You might wonder how to measure the Sun's position against the stars when the stars aren't visible while the sun is shining. One answer is to observe what stars are visible right on the horizon just before the Sun becomes visible in the same spot. Or to avoid the problem of needing to look towards the Sun (which you should NEVER do!), you can note what stars are exactly *opposite* of where the Sun rises. Then from your knowledge of the night sky, you can figure out later which stars were behind the Sun.

each day traverses nearly the same angle around the Sun. The planet Mercury which orbits close to the Sun roughly follows this straight line, but undulates around it due to its orbit, sometimes running ahead and sometimes falling behind; however, whereas the Sun returns to the same place after one year, the undulations of Mercury don't — it takes them 46 years to start repeating.

{Note, there needs to be a plot there illustrating what was just described.}

Determining these periods of repetition takes decades, even centuries of systematic record keeping across generations. But if you can determine the period of the cycle, and you have observations of the previous cycle, then you can make predictions about the next one. The Babylonians discovered relationships that repeated over very long periods of time. Jupiter returns to nearly the same place in the sky every *71 years* and Mars every *79 years*. Even though the Babylonians had no idea *why* this happened, they knew from the data that it did, and that was enough to enable prediction.

This idea was taken to an extreme by a 3rd century BCE Babylonian priest named Berossus. As related by Seneca, he described the ultimate period: the point when all the planetary cycles simultaneously complete and reset; not just when is Mars in the same place in the sky, but when is every planet back in the same place. As the starting and ending point of this cycle, called The Great Year, he chose the time when all the planets were exactly in alignment. He predicted that if this happened with the planets in Cancer the world would end in fire and if in Capricorn, a great flood. Such an alignment hasn't happened yet, so the jury remains out on this doomsday prophecy[5].

These periods, while close, were not exact repetitions. For example the Babylonians knew that after 8 years Venus returned to almost the same place in the sky, but it would be off by about 2.5 degrees. Almost but not quite. Predictions that just copied what happened the previous cycle would not be accurate enough, so it was important to note the small but real difference from an exact repetition.

This worked — figure out the period of repetition, note the slight differences from one cycle to the next, and use previous records to predict the future — but it must have been quite cumbersome to keep 79 years of observational records around on clay tablets! Reducing the number of numbers needed was the next step, but it came at the cost of doing more arithmetic.

The Babylonians realized from looking at the tables of observations that the amount by which the position changed from one observation to the next — the differences between successive entries in their tables — was nearly constant

[5]If such an event were to happen, it would not happen for millions of years. Long enough to not be too concerned about this potential end of the world.

across large stretches of the Zodiac. Their idea was to reduce the massive set of position observations to a much smaller set of differences and then to use those differences to reconstruct the positions on-demand.

They had two ways of approximating these differences — known by the aggressively unhelpful names System A and System B. System A changed the difference value infrequently but in large increments when it did change. System B varied the difference up and down like a zig-zag — it was always changed, but by smaller amounts. In either case the update procedure was much the same — take the current position, look up the difference, and add them together to get the next position. System B is theoretically more sophisticated than System A, but it was harder to deal with in practice, so both systems ended up being used side-by-side for thousands of years.

In all the tablets we've found of the Babylonian astronomer priests, we haven't found one that posits a mechanism, a model, for *why* the heavens behaved the way they did. The heavens moved, the priests observed, and from those observations, using arithmetic, they were able to make accurate predictions about the future. It would be left to the Greeks to ask the question — what system of motion could produce our observations?

Claudius Ptolemy lived in the 2nd century AD, perhaps in Alexandria, and with his astronomical treatise, the *Almagest*, had the final say on the matter for the next 1500 years.

His system placed the Earth at the center of the Universe, which, with hindsight, was quite a large blunder. But even today it is not so easy for a layman to refute his two arguments for why the Earth must not move: the lack of stellar parallax and that objects not attached to the Earth stay motionless relative to the ground. Parallax is best understood by placing your thumb at arms length, covering one eye and pointing it at an object a few arm lengths away; swap which eye is covered and note that your thumb has moved and is no longer pointing at the object. Stellar parallax means that we should observe the same thing when looking at stars with our "eyes" being the Earth at opposite points in its orbit around the Sun. But Ptolemy could not see it; indeed nobody was able to directly observe it until the 19th century. The reason is both simple and incomprehensible — the stars are absurdly far away. Given the choice between the Earth not moving and the stars being nonsensically far, it was an easy choice for Ptolemy and his contemporaries[6].

The system he describes in his *Almagest* is far more complicated than the Sun, Moon and planets moving in circles around the Earth. *That* system would have produced horrendous predictions about the positions of celestial bodies – certainly less accurate than the Babylonian system based purely on observation and

[6] Well, except Aristarchus.

arithmetic. The actual system Ptolemy devised required sophisticated mathematics and arithmetic to fit the model to observations and then use the model to make predictions.

To fit the observations, in particular "retrograde" motion — when the planets move backward opposite their normal direction across the sky — the planets move in circles, called epicycles, whose center itself moves around a bigger circle which encircles the Earth. Then, to account for the non-uniform speed throughout the Zodiac that the Babylonians also observed, he had to place the center of the larger circle away from the Earth itself. This offset, called an equant, meant that sometimes the planets were closer to Earth and appeared to move faster than when they were farther away. The equant was not easily accepted by all — to some, it violated the notion that the Cosmos must consist only of perfectly uniform circular motions. Fifteen hundred years later Copernicus's distaste for the idea drove him to seek a new model of the Cosmos with the Sun at the center.

The equant also made the arithmetic particularly complicated. No longer was simple arithmetic enough — Ptolemy needed trigonometry. Trigonometry allows the astronomer to measure what is tractable and then calculate that which cannot be measured. Measuring angles was the astronomer's bread and butter; with trigonometry and angles alone, the positions of the heavenly bodies could be predicted.

Euclid, based in Alexandria a few hundred years earlier, had provided the theorems with his *Elements*. In it he laid out all the known results of plane geometry and number theory, proving ever more general results from a small set of basic axioms. The basic ingredients necessary for solving triangles are there, but a very practical piece was missing — how to deal with arbitrary angles like 37 or 52 degrees that came from real astronomical observations.

Ptolemy's solution was a table of chords[7], a function closely related to our modern *sine* function. Imagine a circle with a radius going from the center to the circle's edge. Now imagine a line perpendicular to this radius going from one side of the circle across the radius to the other — that perpendicular line is a chord. The two points of the chord form a triangle with the circle's center. If the chord is drawn very near the edge of the circle it will be quite small and the angle at the center of the triangle will also be quite small. If we slide the chord along the radius until it's very close to the center, it will almost be as big as the diameter of the circle and the angle will be almost 180 degrees.

[7] We know from Theon of Alexandria's *Commentary on the Almagest* that before Ptolemy, Menelaus composed a book titled *On the calculation of the chords in a circle* and that Hipparchus also composed a similar work even earlier, but both are lost. Hipparchus's table contained entries only every $7\frac{1}{2}$ degrees compared with Ptolemy's $\frac{1}{2}2$.

He tabulated the relationship between the angle of this triangle and the length of the resulting chord for every angle from $\frac{1}{2}$ of a degree all the way up to 180 — a fine enough division to be used for precise astronomical calculations. This table, combined with Euclid's plane geometry and Menelaus's spherical trigonometry, would allow him to create a model of the movement of the Heavens and use it to generate predictions of the future. It was also the largest computational undertaking man had yet performed.

To build this table, Ptolemy started from what geometry *could* provide: exact chord lengths for a handful of special angles. Values of the chord function were known, due to Euclid, for special angles related to regular polygons: equilateral triangles, squares, pentagons and hexagons. They gave you values for the chords of 120, 90, 72, and 60 degrees, but what about the rest? Getting the rest relied on a remarkable theorem, today known eponymously as Ptolemy's Theorem, which is proved in the *Almagest* and then used to make practical geometry and trigonometry possible. The theorem relates the lengths of the sides and diagonals of a quadrilateral inscribed in a circle. It allowed Ptolemy to derive two critical results. The first, the relationship between two angles with known chords, and chords for the sum and difference of those angles. The second, which was derived from the first, was a way to generate the chord value for half of a given angle with a known chord.

With just these tools — a few exact starting values, the ability to add and subtract angles, and the ability to halve them — he had a clear if laborious path to a table with fine spacing. Starting from the hexagon's 60 degrees, he could halve his way down: 30, 15, $7\frac{1}{2}$, $3\frac{3}{4}$ degrees, and so on, getting finer with each step. Combined with addition and subtraction, this would eventually populate a table.

But Ptolemy was more clever than this. The regular pentagon gives the chord of 72 degrees. From his addition formulas, he could calculate the chord of 75 degrees (since $75 = 45 + 30$, and he already had both). The subtraction formula then gives $75 - 72 = 3$ degrees. Now he only needed to halve twice — from 3 to $1\frac{1}{2}$ to $\frac{3}{4}$ of a degree — rather than grinding down from 60 through many more halvings.

There was one gap he could not close. Halving can take you from 3 degrees to $\frac{3}{4}$ of a degree, but it cannot give you exactly 1 degree — for that, you would need to trisect an angle, which the Greeks had proven to be impossible by geometric construction alone and did not know how to do with algebraic tools. To reach the chord of 1 degree, and from there to fill in his table at half-degree intervals, Ptolemy needed a different approach. He proved a pair of inequalities that bounded the chord of 1 degree between two values he *could* calculate, and showed that the bounds were tight enough — agreeing to the precision he needed

— that the estimate was as good as exact for practical purposes.

Finding the value for the chord of $\frac{1}{2}$ was just the beginning — most of the work still remained to be done. Each entry in the table required many multiplications and divisions and was dependent on the ones before it; any introduced errors would propagate and render the table useless. It would have taken months of doing arithmetic all day long to produce the final table which occupied several pages of the *Almagest*. It gave chord values for every half degree from $\frac{1}{2}$ to 180 degrees, accurate to the equivalent of about five decimal places — all expressed, of course, in the sexagesimal system inherited from the Babylonians.

With this table it was now possible to use triangles and trigonometry to make sense of the Heavens. Ptolemy could take the geometric scaffolding of his model — the epicycles, the eccentric circles, the offsets — and compute its parameters: how big each circle was, how fast things moved and with those compute actual predicted positions for the Sun, Moon, and planets at any given time. Every prediction required trigonometry and a fair amount of arithmetic — multiple lookups in the chord table, combined with multiplications and additions.

Ptolemy's system needed 20-30 observations to accurately determine the parameters of each planet. Once those parameters were established, you could just run the model forward and predict the future as far forward as you wanted to go — if you were willing to do the math. Contrast this with the Babylonian system: without a model of the underlying system they needed records of the entire 79 year cycle of Mars, and ideally multiple such cycles — thousands of observations in all — to be able to predict its motion.

Having solved one problem — the prediction of the Heavens — Ptolemy created another. Astronomy was still done significantly in the service of astrology — predicting the heavens to predict earthly events. But if the heavens could be predicted completely, what did that mean for us? If the positions of the planets shaped your life, and those positions are determined by a mechanical system that grinds inexorably forward, then the future is already written. What room is left for choice?

Ptolemy confronted the problem directly in his *Tetrabiblos*, a companion work to the *Almagest* that applied his astronomical system to astrology. His answer had two parts. The first was that celestial influences are tendencies, not commands. Just as the Sun drives the seasons and the Moon governs the tides, the planets shape conditions — but a farmer who knows a drought is coming can store water. Foreknowledge does not eliminate agency; it enhances it. The stars incline, Ptolemy argued, but do not compel.

The second was complexity. Even if celestial influences were in principle deterministic, they interact with local conditions — geography, climate, upbringing, the circumstances of the moment — in ways too intricate to predict exactly. The prediction is real but approximate; certainty is unattainable in practice.

This same problem would arise again, more seriously, when Newton gave precise laws to describe how *everything* moved — not just the heavens but objects on Earth. If the universe is a machine whose future state follows inevitably from its present one, then free will becomes difficult to defend. Laplace laid out the following thought experiment: he imagined an intelligence vast enough to know the position and velocity of every particle in the universe, and concluded that for such an intelligence "nothing would be uncertain and the future, as the past, would be present to its eyes." In this view, our sense of choice is merely ignorance — we feel free only because we cannot see the machinery that determines us.

Ptolemy's complexity defense turned out to be durable. Even if Newton's laws are deterministic, the equations governing three or more bodies interacting gravitationally cannot be solved exactly. Many systems exhibit what we now call chaos — an extreme sensitivity to initial conditions in which tiny differences in starting points lead to wildly different outcomes. To predict the future of such a system you would need to know its present state with *infinite* precision, and no measurement can provide that. Laplace's demon would need not just vast computational power but perfect knowledge of the present, down to the last decimal place — and there is no last decimal place.

There is a deeper insight here, one that has only recently been made precise[8]. Whether something is predictable depends not just on the thing itself, but on the computational resources of the one doing the predicting. Consider a simple example: a computer program that takes a single number as input and produces a long sequence of numbers as output. The program is entirely deterministic — given the same input, it always produces the same output. But the sequence it produces *looks* random; it passes every statistical test for randomness you might think to apply and no pattern can be found in it.

Such programs, called pseudorandom number generators, are so effective that they are the foundation of modern cryptography. Our ability to keep secrets rests on the fact that it would take thousands of years to figure out the input to the pseudorandom number generator program that reproduces a sequence of known outputs, even though it is theoretically possible. An observer who did figure it out, who "cracked" the code so to speak, would be able to predict every element in the sequence — the randomness would be gone. But an observer who has only the output, and not enough computational power to work backward to the input, cannot distinguish it from a sequence produced by flipping a coin.

[8] https://arxiv.org/abs/2601.03220 *From Entropy to Epiplexity: Rethinking Information for Computationally Bounded Intelligence* This is a new enough idea, that some readers might find a direct reference useful.

The “randomness” is not in the sequence; it is in the gap between the sequence’s complexity and the observer’s capacity to analyze it.

This reframes the ancient question. Determinism that cannot be exploited — whether because chaos demands impossible precision in the initial conditions or because the computation required exceeds what any real observer can perform — is operationally indistinguishable from genuine freedom. Whether the universe is “truly” random at its foundations or merely intractable to predict may be a distinction without a difference for any intelligence that actually exists within it.

With the nature of free will still on the scales, the practical work of predicting the heavens continued. Ptolemy’s model was so successful it superseded everything that came before it. It was disseminated by Arab astronomers across the Islamic world and even farther East. While each civilization it touched would refine and extend its ideas, the core would remain the same until it made its way back to Western Europe.

Arithmetic Shines Light on the Heavens

The Multiplication Bottleneck

> "I also ask you, my friends, not to condemn me entirely to the mill of mathematical calculations, and allow me time for philosophical speculations, my only pleasures." — Johannes Kepler, letter to Vincenzo Bianchi (1619)

> "Since nothing is more tedious, fellow mathematicians, in the practice of the mathematical arts, than the great delays suffered in the tedium of lengthy multiplications and divisions…" — John Napier, *Mirifici Logarithmorum Canonis Description* (1614)

One and a half millennia after Ptolemy, in the late 16th century, the prevailing view of the heavens had not changed from the static geocentric world of Aristotle and Ptolemy. Copernicus had just reignited Aristarchus's ancient claim that the Earth moved around the Sun by publishing "On the Revolutions of the Heavenly Spheres." Scholars defended geocentrism and the ancients vigorously from attacks by the upstart heliocentrists inspired by Copernicus. We know now that Copernicus was right. But at the time, his theory, while elegant, did not actually offer more *predictive* power than Ptolemy: he still used circles and epicycles — circles going round circles — and surprisingly, had little more data than Ptolemy despite the passage of 1500 years.

Tycho Brahe, a Danish nobleman born in 1546, grew up in this intellectual environment. He also grew up in an era of great astronomical luck — he would observe a solar eclipse, great conjunction, supernova, and great comet within the span of 17 years. That's not likely to happen again for another approximately 5,000 years.

The first event, a solar eclipse in 1560, an uncanny event to observe regardless, particularly impressed the young astronomer because it had been *predicted*.[1] The prediction being off by a day did not seem to lessen his excitement or wonder that we could predict the heavens. Suitably awed, he set about learning to make the same kinds of predictions and promptly acquired all the available material he could on Ptolemy's techniques from the experts of the time.

It was with his new predictive powers and a discerning eye that he observed the great conjunction of 1563 — an event where Jupiter and Saturn appear very near to each in the sky. It too, had been predicted by the almanacs and ephemerides of the time, but by his judgement — poorly. It was this event that convinced him more and better *data* was needed to make better predictions, a truly novel thought at the time.

In November of 1572 a truly rare astronomical event occurred – a new star appeared in the sky. A star in the constellation Cassiopeia had exploded, i.e. gone supernova, an event that released so much energy the star's dying moment 8,000 light years away was one of the brightest objects in the sky. This event shook Brahe to his core. It is hard for us moderns to appreciate what a constant presence the night sky was, equanimous in its ability to be unperturbed by human events. It was there, the same, every night, slowly rotating across the sky; some stars were only visible at certain times of year, but they always came back and new ones never arrived.

According to Aristotle, the heavens were perfect and unchanging. Yet this nova ("new star"), a term Brahe coined in the title of his 1573 publication, *De nova stella*, was brighter than Venus, the brightest object in the night sky, and visible for months[2]. According to his observations it lay beyond the Moon, in the supposedly immutable sphere of the fixed stars. Something about the old theories was wrong.

[1] The author has been unable to find records of exactly *who* made this prediction. It is relatively easy to predict that an eclipse will occur every 18 years, due to the so-called Saros cycle of the sun, moon and earth. This was known to the ancients. But predicting exactly *where* a solar eclipse will happen is far harder and was not mastered until well into the 17th century. Perhaps whoever made this prediction just got lucky that the solar eclipse happened to be observable in western europe. Or maybe they were far ahead of their time.

[2] There are conflicting accounts as to whether it was brighter than Venus (the otherwise brightest object in a moonless night sky), or just as bright. Tycho claims brighter, other sources, the same. It would only have been at maximum brightness for 1-2 weeks and faded in and out.

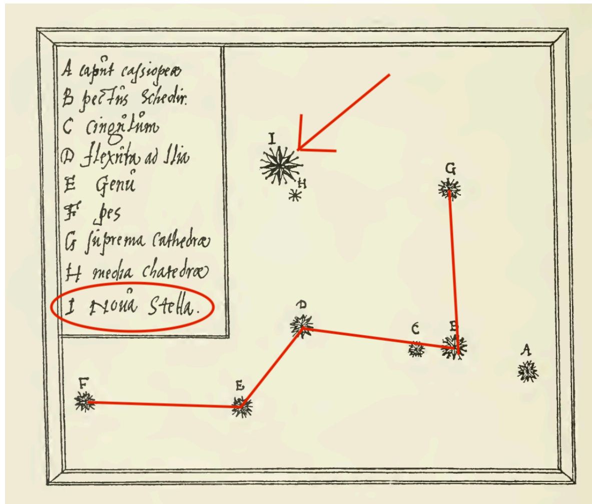


Figure 3.1: Tycho Brahe *De Stella Nova* Location of New Star

Tycho's response was not to speculate, but to measure[3]. Aristotle had reasoned about the heavens without measurement. Ptolemy had made just enough observations to constrain his models. Medieval scholars debated cosmology largely without data at all. Tycho, in contrast, collected far more information than he, or anyone else, could immediately explain. It isn't hyperbole to say that Tycho initiated the entire scientific revolution with his approach.

In 1577, his measurements of a great comet that appeared in the sky for 3 months convinced him further that the old Aristotelean wisdom must be wrong. The data implied that it too must lie beyond the moon. The answers lay in data and already his new observatory, Uraniborg, was being built on a small island off the coast of Denmark; it was to be the largest scientific data collection operation the world had yet seen.

Turning each observation of the night sky into a usable measurement involved a staggering amount of arithmetic. Observations measured positions relative to the horizon; but to be useful, they needed to be transformed into absolute coordinates on the celestial sphere — a process called reduction. Tycho and his assistants made their observations at exactly the right time to minimize the amount work involved and it still required about ten multiplications, twenty trig table lookups and a few divisions.

[3]Or at least, to measure *in addition* to speculating.

The accuracy at Uraniborg was the equivalent of measuring the width of a human hair at arm's length. That level of precision meant the trigonometric tables needed 6 or more digits of accuracy, and each multiplication needed to operate on the same. Ptolemy's table of chords had been improved upon during the fifteen intervening centuries, and the 16th century tables of Regiomontanus, Rheticus and Viète now possessed sufficient accuracy, both in terms of spacing and digits, to match.

To a modern reader, the difficulty of multiplication is easy to underestimate. We memorize the multiplication table for single-digit numbers and learn a set of rules for extending it to larger ones. Beyond a few classroom exercises, perhaps extending to three-digit problems worked once and forgotten, multiplication rarely intrudes into everyday life. When it does, we are surrounded by machines that make it effectively free. Typing the inputs now takes far longer than getting the result.

To give you, the reader, an idea of the difficulty, the author timed himself adding and then multiplying the same two six digit numbers[4]. It took him 10 seconds to do the addition and 3 minutes and 30 seconds to do the multiplication, or over twenty times as long. He also made an error, which took a further 5 minutes to discover.

Now imagine needing to do this ten times *per* observation. It could take an hour to produce one error-free reduced observation. Dozens of observations could be recorded each evening; the amount of effort to reduce them all the next day would require many scribes working dawn to dusk. Tycho had built the world's first data center and for the first time the speed of arithmetic was the limiting factor in scientific progress.

In 1580, a mathematician named Paul Wittich arrived with a solution – he possessed a way to calculate faster than everyone else. He did not fully understand where it came from; he had found it buried in the work of a little-known mathematician from the early sixteenth century, Johannes Werner. He could not even prove that it worked. But he knew that it did; and more importantly, he knew where this discovery would be most valuable. Tycho recognized its use immediately and put it to work[5].

[4] The author used the numbers 931605 and 284917. He had originally intended to use 10 digit numbers, but scaled back his ambition someway through the multiplication when he was sure he had made errors due to misaligning some columns. Certainly, with more practice and some basic learnings (like start the multiplication problems on the right edge of the paper, so there is room for the values to grow to the left and leave spacing between columns so that they remain *exactly* aligned), the author would get faster. But the main point remains – multiplication is a lot slower than addition and only becomes more so the bigger the numbers become.

[5] It is likely that Wittich took something in return – the designs of Tycho's astronomical

What was Wittich's secret? He could multiply numbers by doing only addition[6]. More specifically, he could multiply the sines of two angles by adding and subtracting the angles and then looking up the cosines of those values in a trigonometric table. Since the tables already existed this was an order of magnitude reduction in work and time for the practical problems they were solving at the observatory. An additional benefit is that since addition is so much less work than multiplication, it is also much less error prone. Speed went up and errors went down.

However, the reduction process required multiplying sines together *and* cosines together. Wittich's method only handled the former; products of cosines could not be simplified with what Wittich brought to Uraniborg. Wittich had done well to find this obscure bit of knowledge and bring it to the right place and the right time, but further progress required a mathematical brilliance which neither Wittich nor Tycho possessed.

The solution to the cosine problem and a much more general one would come from a Swiss clockmaker, Jost Bürgi. Tycho maintained a correspondence with Bürgi's aristocratic patron who called Bürgi a "second Archimedes" for his mechanical and mathematical brilliance. It is likely from this relationship that Bürgi learned of Wittich's technique and the cosine problem. Reportedly, it took him little time to produce a proof of the sine formula and from this proof only a small modification was required to derive the formula for the product of cosines via addition. Armed with these two formulas, Bürgi was able to remove most of the arithmetic labor from the process of reduction[7].

These formulas were actually waiting there patiently since Ptolemy wrote down his theorem in the Almagest. He used them to *construct* his tables, but the recognition that once the tables had been painstakingly constructed, the formulas could be used backwards together with the tables to avoid multiplication took 1500 years. The leap required both need and brilliance.

Some readers will recognize these as the "product-to-sum" formulas from high school trigonometry. In the author's experience they were memorized and promptly forgotten without context as to why they were once useful or that

instruments and some of his methods. It was the beginning of a long-running dispute over who deserved credit for what.

[6] The technique was later named prosthaphaeresis, from the Greek for addition and subtraction

[7] Much of the work in the reduction process comes from applying the spherical law of cosines — which requires three multiplications. Prosthaphaeresis eliminates all of them:

$$\cos a = \cos b \cos c + \sin b \sin c \cos \alpha \tag{3.1}$$

$$= \cos b \cos c + \tfrac{1}{2}\left[\cos(b-c) - \cos(b+c)\right] \cos \alpha \tag{3.2}$$

$$= \cos b \cos c + \tfrac{1}{2}\cos(b-c)\cos\alpha - \tfrac{1}{2}\cos(b+c)\cos\alpha \tag{3.3}$$

$$= \tfrac{1}{2}\left[\cos(b-c) + \cos(b+c)\right] + \tfrac{1}{4}\left[\cos(b-c-\alpha) + \cos(b-c+\alpha) - \cos(b+c-\alpha) - \cos(b+c+\alpha)\right] \tag{3.4}$$

they were once among the most prized tools of European science. The author might have been more likely to remember if our math and science curriculum told stories about our collective intellectual journey rather than just leaving random artifacts out in a field all jumbled about.

It is possible to generalize this technique to the multiplication of arbitrary numbers, even those not arising from the reduction calculations. And while it's certainly better than doing a brute-force calculation most of the time, it's actually still rather cumbersome for arbitrary numbers.

The solution for the general case was actually foreshadowed by Archimedes himself in his work *The Sand Reckoner*. In his time, it was thought that there was not a number big enough to count all the grains of sand on Earth. Archimedes goes further — not only will he name a number large enough to count the number of grains on the Earth, he will name the number required to fill the Earth, and then the number of grains required to fill the *Universe*. This incredible work is one of the only sources we have for the existence of Aristarchus's heliocentric theory — that the Earth goes round the Sun. Archimedes assumes Aristarchus's theory because the heliocentric universe is significantly larger than one with the earth at the center, which makes his feat of counting the grains of sand required to fill it all the more impressive. He estimates the size of the sun, moon, and earth, then uses those to estimate the size of the universe itself.

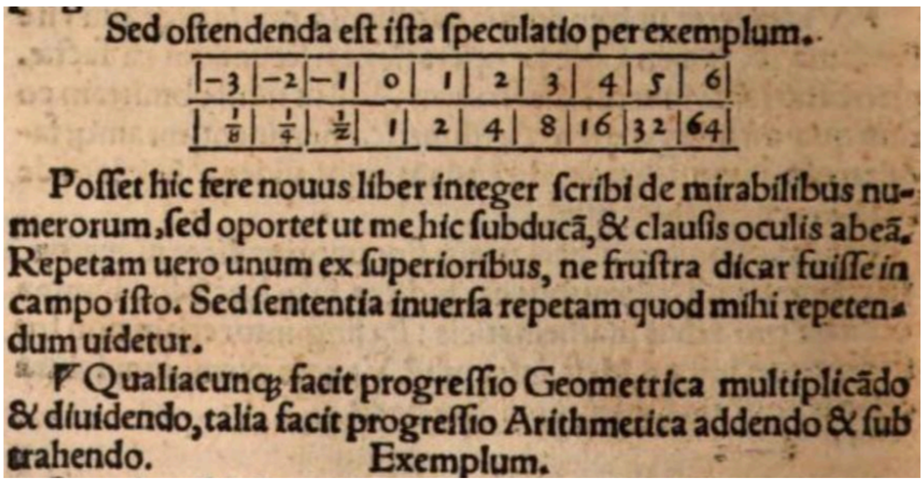
But Archimedes had to solve another problem. In ancient Greece, the largest named number was a myriad — 10,000; they literally did not have words or notation to systematically express bigger numbers and the number he needed to express was far, far larger. His solution was to invent what we would call exponents.

An exponent tells you how many times a number is multiplied by itself. For example, $10^2 = 10 * 10 = 100$ and $2^3 = 2 * 2 * 2 = 8$. The number being multiplied by itself is called the *base*; in the first example the base is 10 and the exponent 2 and in the second the base is 2 and the exponent is 3. In order to manipulate these exponentiated numbers, Archimedes proved that when you multiply two exponentiated numbers together, you *add* their exponents to get the result. Archimedes's first big number was a myriad-myriad, or myriad multiplied by itself, $10^4 * 10^4 = 10^8$, which is one hundred million. His estimate for the size of the universe was about 10^{63} grains of sand.

It is this result — that multiplying exponentiated numbers requires only *adding* their exponents — that connects sand reckoning back to simplifying multiplication. But it would take nearly two thousand years for anyone to say so explicitly. In 1544, the German mathematician Michael Stifel published *Arithmetica integra*. Eleven years earlier Stifel had achieved some notoriety for using numerology on a verse from the Revelation of John to predict the end of the world for October 19, 1533; when it failed to arrive, he required protective custody from those he had deceived — prison[8]. But his *Arithmetica integra* includes this remarkable passage (the original Latin is in fig. 3.2):

"Here almost a whole new book could be written about the wonders of numbers, but it is necessary that I withdraw myself here, and go away with closed eyes. I will repeat indeed one thing from the above, lest I be said to have been in this field in vain…

Whatever things a Geometric progression does by multiplying and dividing, such things an Arithmetic progression does by adding and subtracting."


Sed oſtendenda eſt iſta ſpeculatio per exemplum.

-3	-2	-1	0	1	2	3	4	5	6
1/8	1/4	1/2	1	2	4	8	16	32	64

Poſſet hic fere nouus liber integer ſcribi de mirabilibus numerorum, ſed oportet ut me hic ſubducã, & clauſis oculis abeã. Repetam uero unum ex ſuperioribus, ne fruſtra dicar fuiſſe in campo iſto. Sed ſententia inuerſa repetam quod mihi repetendum uidetur.

Qualiacunq; facit progreſſio Geometrica multiplicãdo & diuidendo, talia facit progreſſio Arithmetica addendo & ſubtrahendo. Exemplum.

Figure 3.2: Stifel's *Arithmetica Integra* folio 249

Stifel clearly knew that this connection would allow for the replacement of multiplication by addition and even gave some examples of it, but he didn't (or couldn't) extend it to work for arbitrary numbers. In Stifel's table, each number in the bottom row is double the one before it, and each number in the top row increases by one. To multiply 1/8 by 64, you find their exponents in the top row: -3 and 6. Add those to get 3, then look up 3 in the top row to find 8 in the bottom row. Multiplication has become addition.

[8]The German expression *einen Stiefel rechnen* (to calculate a boot), meaning to calculate nonsense, is said to derive from this incident. Like Plato and his tale of the world's creation, it is a *likely* story.

But what do you do if you want to multiply 7×3.5? Stifel's table is no help here. In Stifel's original table each number on the bottom row was the previous one multiplied by 2. If instead, he replaced it with 1.5, then the first few numbers would be $1, 1.5, 2.25$ — closer together. If 1.1, then $1, 1.1, 1.21$ — even closer! The closer the ratio is to 1, the finer the divisions — and the more likely we are to find any number we need in the table. But it also means creating such a table would require a Sisyphean number of multiplications — perhaps this is what Stifel meant when he wrote that 'a whole new book could be written about the wonders of numbers' before closing his eyes and walking away.

Jost Bürgi not only understood the principle but had the brilliance to make the calculation tractable. He constructed it entirely through addition, exploiting a multiplication trick sometimes taught to grade school students. To multiply a number by eleven, you just shift it one digit right and then sum the original and new number. For example, 11×13:

```
13
 13
---
143
```

But he needed a number very close to 1, so he chose 1.0001. Instead of shifting by one digit, he needed to shift by four digits. The first three numbers in his sequence would be: $1, 1.0001, 1.00020001$.

```
10001
    10001
---
100020001
```

That's it. He just needed to perform additions like that over and over and over again; in the end 23,027 times until his sequence ended at ten[9]. That's a non-trivial amount of work, certainly — but modern estimates suggest it might have only taken him a few months of calculating. If multiplication was needed at each step instead of addition it would have taken a lifetime. And after months of calculations which resulted in the first general purpose table which would allow humanity avoid the endless mill of calculation, he proceeded to share it with almost no one; it was a closely guarded secret for two decades.

One man knew what Bürgi had. Johannes Kepler arrived in Prague in 1600 to work as an assistant to Tycho Brahe. Tycho had been forced out of Denmark when the new king, Christian IV, whose gaze never turned skyward, cut his

[9] He could stop at 10 because once he had all the "bottom" numbers between 1 and 10, he could scale numbers bigger than 10 down and numbers smaller than 1 up to still use the table.

funding, and had resettled at the court of the Holy Roman Emperor Rudolf II, bringing with him the most valuable scientific dataset in the world — decades of planetary observations of unprecedented precision. Tycho's first task for Kepler was to figure out how to predict the orbit of Mars. His unassailable measurements disagreed with the predictions of Ptolemy which meant the true nature of the solar system was yet to be discovered.

He died just a year later with the task far from complete. On his deathbed he is reported to have said, "Let me not seem to have lived in vain," to Kepler before entrusting him with the data[10] and tasking him with shining light not just on Mars but the Heavens. Tycho's favored theory of the solar system was not correct in the end, but his approach of measuring to find better theories and truth laid the foundations of the scientific revolution. Few men can claim to have led lives *less* in vain.

Kepler's initial attack was to assume that the Ptolemaic model was essentially correct, but just that its parameters, like the size of the circles and how fast they moved, hadn't been determined accurately enough. He modified Ptolemy's model slightly, allowing more freedom in the orbital parameters to account for why previous models didn't fully predict Mars's motion.

To determine the parameters for his modified Ptolemaic model he relied on a special kind of observation called oppositions. They are taken when the Sun, Earth and Mars all lie on the same line; if you flew in a straight line from the Sun to the Earth and kept going you would get to Mars[11]. An Earth-Mars opposition occurs every approximately 2 years and Tycho had 12 recorded observations. Kepler made an initial guess for the values informed by Ptolemy and then would compare that prediction against 4 of the 12 measured values; based on the differences, he would update the guess and start again. Each guess involved a chain of trigonometric calculations — multiplying sines and cosines, solving triangles on the celestial sphere.

In chapter 16 of his *Astronomia Nova*, Kepler addressed the reader directly regarding this effort: "If this wearisome method has filled you with loathing, it should more properly fill you with compassion for me, as I have gone through it at least seventy times at the expense of a great deal of time." Kepler and Bürgi worked in the same building at Prague Castle, and Kepler knew how much effort Bürgi's tables could save. But Bürgi would not share them. Years later, Kepler would write of him in the preface to his *Rudolphine Tables*: "Justus Byrgius was led to these very logarithms many years before Napier's system appeared; but he, a hesitant man and a guardian of secrets, abandoned the child at birth, and did not raise it for the common benefit."

[10] Kepler would be embroiled in a dispute with Brahe's heirs for years over who really owned the data. The skeptical might think it convenient that Kepler, the only narrator of Brahe's final wishes, was also their main benefactor.

[11] If you imagine a typical top-down 2D representation of the solar system. In three dimensions, this isn't quite true a straight line from the sun to the earth almost never actually goes through Mars.

All this effort was not wasted. He found a set of parameters that matched Tycho's observations within his observational error of 2 arc-minutes. But Kepler did not stop at this apparent vindication of Ptolemy; he went on to check if the distances implied by this theory also agreed with the observations — and they didn't. Correcting the parameters to make the distances agree increased the error of the predictions to eight arc-minutes.

Eight arc-minutes is less than a quarter of the apparent width of the Moon. Previous astronomers could have shrugged it off; their data was not precise enough to detect so small an error. But Kepler knew that Tycho's observations were accurate to within two minutes, and he refused to ignore the discrepancy. "These eight minutes alone," he wrote, "will lead us along a path to the reform of the whole of Astronomy."

And so they did — though not without further years of the same brutal arithmetic. Kepler tried curve after curve, each attempt requiring fresh rounds of iterative calculation, until finally he tried an ellipse. It fit. The orbit of Mars was not a circle, but an ellipse with the Sun at one focus. It swept out equal areas in equal times, moving faster when closer to the Sun and slower when farther away. These first two laws appeared in his *Astronomia Nova* of 1609, wrested from Tycho's data with prosthaphaeresis and years of sheer computational force.

But to extend that discovery to the rest of the solar system and build the prediction tables that astronomers and navigators actually needed would require the very tool that Bürgi had hoarded. Kepler and Bürgi had worked side by side at Prague Castle for nearly a decade, so Kepler knew about the tables and had almost certainly seen them. But fifty-eight pages of nine-digit numbers — twenty-three thousand entries — were not something one could borrow for an afternoon and copy. There was, almost certainly, only one manuscript, and it was Bürgi's.

It was a Scottish nobleman, John Napier, who gave it to the world. Unlike Bürgi, Napier believed, as he wrote in his preface, that "the secret is best made common to all, as all good things are." Napier's formulation was different from Bürgi's: not a table of progressions but an elegant thought experiment of a point moving along a line with a speed proportional to how far it had left to travel. Imagine Zeno's arrow, trying to cross the remaining distance but at each moment covering only a fraction of what remains — moving slower and slower, never quite arriving without an infinitude of time.

His tables took nearly two decades to compute, compared with the months Bürgi had needed. But Napier was willing to give the world the artifact produced by those two decades of his life. Bürgi produced a solution to the multiplication bottleneck decades before Napier published his, yet it is Napier who changed the world while Bürgi's works collected dust.

Kepler received a copy of Napier's *Mirifici Logarithmorum* in 1617 and immediately grasped their power. He, however, disapproved of Napier's formulation so much that he later wrote his own book, the *Chilias Logarithmorum*, replacing the kinematic foundation with one built solidly on the Greek geometrical tradition even though the practical result was much the same. Whether grounded in motion or in Euclid, Laplace would later say that logarithms "doubled the life of the astronomer." Every astronomer, for the next four centuries.

Armed with these tables, Kepler could finish the work that the ellipse had begun. His third law — that the square of each planet's orbital period is proportional to the cube of its distance from the Sun — appeared in 1619, an insight likely inspired by logarithms. And in 1627, eighteen years after *Astronomia Nova*, he published the *Rudolphine Tables*: comprehensive predictions for every planet, computed using logarithms, with Napier's tables bound into the volume as an essential tool. In the preface, he reflected on encountering logarithms a decade earlier; he called it a "happy calamity" — it had made predicting the motions of all the planets possible.