1 Introduction to Computer Science

Figure 1.1 Computing is everywhere, affecting everyone, for better and for worse. (credit: modification of "Whereas design is expansive, engineering is narrowing" by Jessie Huynh/Critically Conscious Computing, CC0)

Chapter Outline

1.1 Computer Science

1.2 Computer Science across the Disciplines

1.3 Computer Science and the Future of Society

Introduction

This textbook will introduce you to the exciting and complex world of computer science. In this chapter, you'll review the history of computer science, learn about its use in different fields, and explore how computer science will impact the future of society. Computer science is a powerful tool, and computer scientists have used their vast knowledge of technology to create and implement technology that has transformed societies around the world.

This book will also introduce the computational thinking aspects of problem-solving and analytical thinking that enable the study of algorithms, which are step-by-step instructions for solving specific problems or carrying out computations. Therefore, this book also covers algorithms and their realization via programming languages, computer systems architectures, networks, and operating systems. The book subsequently delves into computer science areas that enable the design and development of software solutions using high-level programming languages (i.e., coding languages designed to be more intuitive for humans), architectural styles and related models, data management systems, and software engineering. Finally, the book demonstrates how to leverage computer science realizations and areas to build modern end-to-end solutions to business and social problems. In particular, the book focuses on modern web applications development, cloud-native applications development, and hybrid Cloud/on-premise digital solutions. The various chapters emphasize how to achieve software solution qualities such as performance and scalability. The last chapter explains how to secure software applications and their applications in the context of various cyber threats. It also explains how to make the right decisions about using computers and information in society to navigate social, ethical, economic, and political issues that could result from the misuse of technology. To conclude this textbook, we'll introduce you to cybersecurity and help you understand why responsible computing is essential to promote ethical behavior in computer science. The book is designed to help students grasp the full meaning of computer science as a tool that can help them think, build meaningful solutions to complex problems, and motivate their careers in information technology (IT).

You're already familiar with computer science. Whenever you use a laptop, tablet, cell phone, credit card reader, and other technology, you interact with items made possible by computer science. Computer science is a challenging field, and the outputs of computer science offer many benefits for society. At the same time, we have to be cautious about how we use computer science to ensure it impacts society in ethical ways. To help you understand this, the next section will explain how computer science came to be and discuss the field's potential.

1.1 Computer Science

Learning Objectives

By the end of this section, you will be able to:

- Discuss the history that led to the creation of computer science as a field
- Define computer science
- Assess what computer science can do, as well as what it should not do

The field of computer science (CS) is the study of computing, which includes all phenomena related to computers, such as the Internet. With foundations in engineering and mathematics, computer science focuses on studying algorithms. An algorithm is a sequence of precise instructions that enables computing. This includes components computers use to process information. By studying and applying algorithms, computer science creates applications and solutions that impact all areas of society. For example, computer science developed the programs that enable online shopping, texting with friends, streaming music, and other technological processes.

While computers are common today, they weren't always this pervasive. For those whose lives have been shaped by computer technology, it can sometimes seem like computer technology is ahistorical: computing often focuses on rapid innovation and improvement, wasting no time looking back and reflecting on the past. Yet the foundations of computer science defined over 50, and as much as 100, years ago very much shape what is possible with computing today.

The Early History of Computing

The first computing devices were not at all like the computers we know today. They were physical calculation devices such as the abacus, which first appeared in many societies across the world thousands of years ago. They allowed people to tally, count, or add numbers (Figure 1.2). Today, abaci are still used in some situations, such as helping small children learn basic arithmetic, keeping score in games, and as a calculating tool for people with visual impairments. However, abaci are not common today because of the invention of number systems such as the Arabic number system (0, 1, 2, 3, . . .), which included zero and place values that cannot be computed with abaci. The concept of an algorithm was also invented around this time. Algorithms use inputs and a finite number of steps to carry out arithmetic operations like addition, subtraction, multiplication, and division, and produce outputs used in computing. Today's computers still rely on the same foundations of numbers, calculations, and algorithms, except at the scale of billions of numbers and billions of calculations per second.

To introduce a concrete example of an algorithm, let us consider binary search algorithm, which is used to locate a number in a sorted array of integers efficiently. The algorithm operates by repeatedly dividing the search interval in half to perform the search. If the number being searched is less than the integer in the middle of the interval, the interval is narrowed to the lower half. In the alternative, the interval is narrowed to the upper half. The algorithm repeatedly checks until the number is found or the interval is empty.

Algorithms may sound complicated, but they can be quite simple. For example, recipes to prepare food are algorithms with precise directions for ingredient amounts, the process to combine these, and the temperatures and cooking methods needed to transform the combined ingredients into a specific dish. The dish is the output produced by following the algorithm of a recipe.

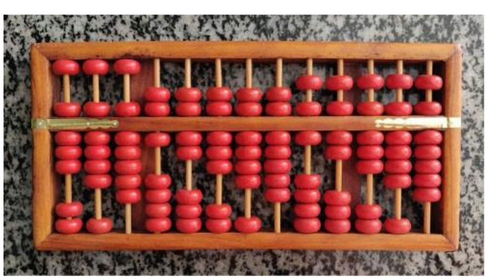

Figure 1.2 An abacus is one of the first calculators. (credit: “Traditional Chinese abacus illustrating the suspended bead use” by

Jccsvq/Wikimedia Commons, CC0)

The next major development in the evolution of computing occurred in 1614 when John Napier, a Scottish mathematician, developed logarithms, which express exponents by denoting the power that a number must be raised to obtain another value. Logarithms provided a shortcut for making tedious calculations and became the foundation for multiple analog calculating machines invented during the 1600s.

Scientists continued to explore different ways to speed up or automate calculations. In the 1820s, English mathematician Charles Babbage invented the Difference Engine with the goal of preventing human errors in manual calculations. The Difference Engine provided a means to automate the calculations of polynomial functions and astronomical calculations.

Babbage followed the Difference Engine with his invention of the Analytical Engine. With assistance from Ada Lovelace, the Analytical Engine was program-controlled and included features like an integrated memory and an arithmetic logic unit. Lovelace used punched cards to create sequencing instructions that could be read by the Analytical Engine to automatically perform any calculation included in the programming code. With her work on the Analytical Engine, Lovelace became the world’s first computer programmer.

The next major development in computing occurred in the late 1800s when Herman Hollerith, an employee of the U.S. Census Office, developed a machine that could punch cards and count them. In 1890, Hollerith’s invention was used to tabulate and prepare statistics for the U.S. census.

By the end of the 1800s and leading into the early 1900s, calculators, adding machines, typewriters, and related machines became more commonplace, setting the stage for the invention of the computer. In the 1940s, multiple computers became available, including IBM’s Harvard Mark 1. These were the forerunners to the advent of the digital computer in the 1950s, which changed everything and evolved into the computers and related technology we have today.

Around this time, computer science emerged as an academic discipline rooted in the principles of mathematics, situated primarily in elite institutions, and funded by demand from the military for use in missile guidance systems, airplanes, and other military applications. As computers could execute programs faster than humans, computer science replaced human-powered calculation with computer-powered problemsolving methods. In this way, the earliest academic computer scientists envisioned computer science as a discipline that was far more intellectual and cognitive compared to the manual calculation work that preceded it.

Richard Bellman was a significant contributor to this effort. A mathematics professor at Princeton and later at Stanford in the 1940s, Bellman later went to work for the Rand Corporation, where he studied the theory of multistage decision processes. In 1953, Bellman invented dynamic programming,[1] which is a mathematical optimization methodology and a technique for computer programming. With dynamic programming, complex problems are divided into more manageable subproblems. Each subproblem is solved, and the results are stored, ultimately resulting in a solution to the overall complex problem.[2] With this approach, Bellman helped revolutionize computer programming and enable computer science to become a robust field.

What Is Computer Science?

The term computer sciencewas popularized by George E. Forsythe in 1961. A mathematician who founded Stanford University's computer science department, Forsythe defined computer science as "the theory of programming, numerical analysis, data processing, and the design of computer systems." He also argued that computer science was distinguished from other disciplines by the emphasis on algorithms, which are essential for effective computer programming.[3]

Computer science is not only about the study of how computers work, but also everything surrounding computers, including the people who design computers, the people who write programs that run on computers, the people who test the programs to ensure correctness, and the people who are directly and indirectly affected by computers. In this way, computer science is as much about people and how they work with computers as it is about just computers.

Not everyone agrees with this definition. Some people argue that computer science is more about computers or software than the people it affects. However, even if we were to study just the "things" of computer science, the people are still there. When someone designs a computer system, they are thinking about what kinds of programs people might want to run. Typically, effort is made to design the computer system so it is more efficient at running certain kinds of programs. A computer optimized for calculating missile trajectories, for example, won't be optimized for running social media apps.

Many computing innovations were initially developed for military research and communication purposes, including the predecessor to the Internet, the ARPANET (Figure 1.3).

---

1 S. Golomb, "Richard E. Bellman 1920–1984," n.d. https://www.nae.edu/189177/RICHARD-E-BELLMAN-19201984

2 Geeks for Geeks, “Dynamic Programming or DP,” 2024. https://www.geeksforgeeks.org/dynamic-programming/

3 D. E. Knuth, “George Forsythe and the Development of Computer Science,” Communications of the ACM, vol. 15, no.8, pp.

722–723. 1972. https://dl.acm.org/doi/pdf/10.1145/361532.361538

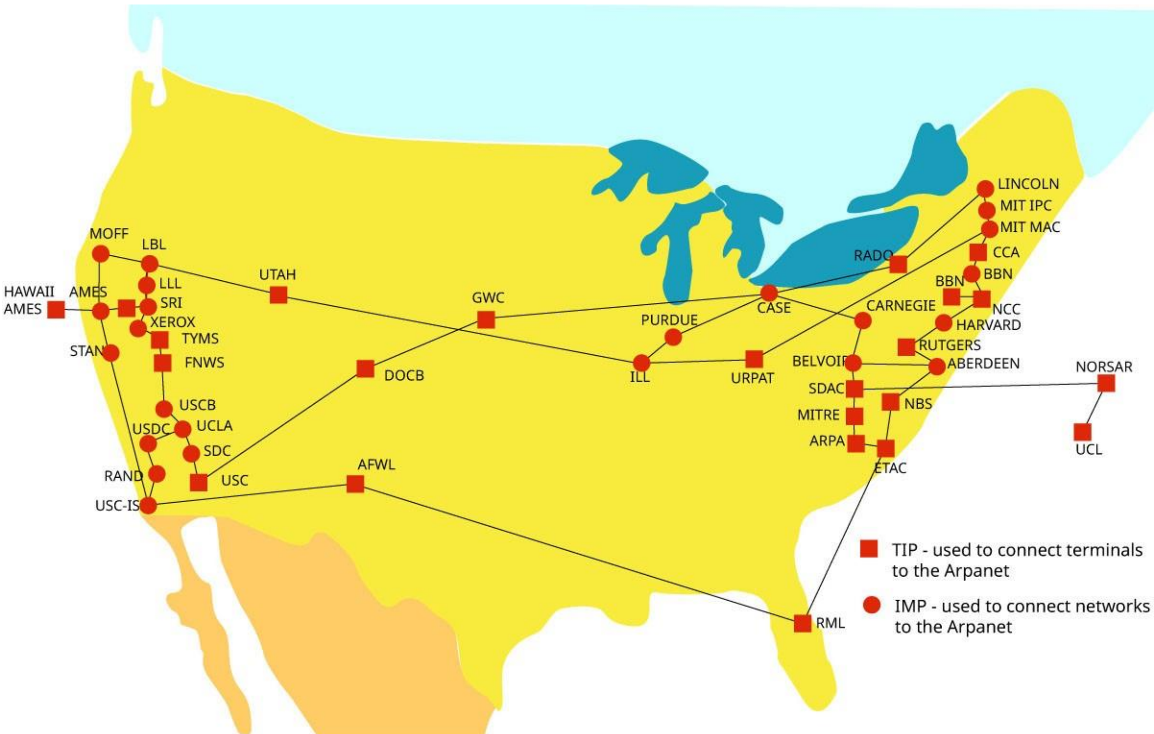

Figure 1.3 The ARPANET, circa 1974, was an early predecessor to the Internet. It allowed computers at Pentagon-funded research facilities to communicate over phone lines. (credit: modification of "Arpanet 1974" by Yngvar/Wikipedia, Public Domain)

What Is a Computer?

While computer science is about much more than just computers, it helps to know a bit more about computers because they are an important component of computer science. All computers are made of physical, realworld material that we refer to as hardware. Hardware—which has four components, including processor, memory, network, and storage—is the computer component that enables computations. The processor can be regarded as the computer’s “brain,” as it follows instructions from algorithms and processes data. The memory is a means of addressing information in a computer by storing it in consistent locations, while the network refers to the various technological devices that are connected and share information. The hardware and physical components of a computer that permanently house a computer’s data are called storage.

One way to understand computers is from a hardware perspective: computers leverage digital electronics and the physics of materials used to develop transistors. For example, many of today's computers rely on the physical properties of a brittle, crystalline metalloid called silicon, which makes it suitable for representing information. The batteries that power many of today's smartphones and mobile devices rely on lithium, a soft, silvery metal mostly harvested from minerals in Australia, Zimbabwe, and Brazil, as well as from continental brine deposits in Chile, Argentina, and Bolivia. Computer engineers combine these substances to build circuitry and information pathways at the microscopic scale to form the physical basis for modern computers.

However, the physical basis of computers was not always silicon. The Electronic Numerical Integrator and Computer (ENIAC) was completed in 1945, making it one of the earliest digital computers. The ENIAC operated on different physical principles. Instead of silicon, the ENIAC used the technology of a vacuum tube, a physical device like a light bulb that was used as memory in early digital computers. When the "light" in the vacuum tube is off, the vacuum tube represents the number 0. When the "light" is on, the vacuum tube represents the number 1. When thousands of vacuum tubes are combined in a logical way, we suddenly have memory. The ENIAC is notable in computer history because it was the first general-purpose computer, meaning that it could run not just a single program but rather any program specified by a programmer. The ENIAC was often run and programmed by women programmers (Figure 1.4). Despite its age and differences in hardware properties, it shares a fundamental and surprising similarity with modern computers. Anything that can be computed on today's computers can also be computed by the ENIAC given the right circumstances—just trillions of times more slowly.

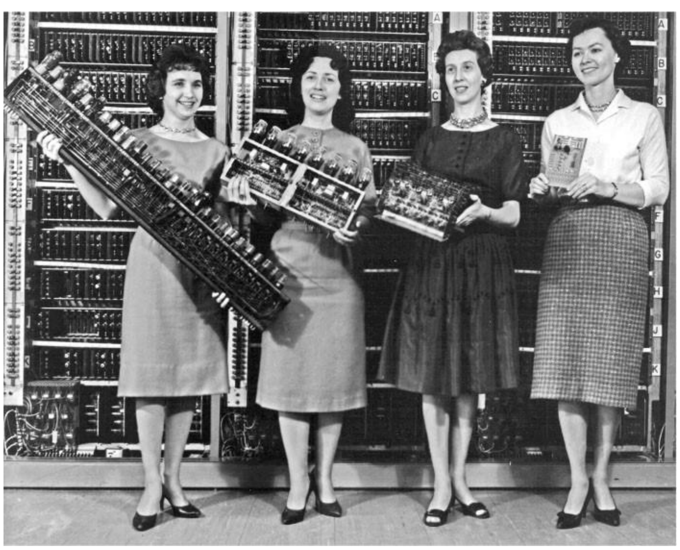

Figure 1.4 This image depicts women programmers holding boards used in computers such as the ENIAC, many of which were designed expressly for ballistics and ordinance guidance research. Today, these room-size computers can be reproduced at a literally microscopic

scale—basically invisible to the human eye. (credit: modification of "Women holding parts of the first four Army computers" by U.S. Army/Wikimedia Commons, Public Domain)

How is this possible? The algorithmic principles that determine how results are computed makes up software. Almost all computers, from the ENIAC to today's computers, are considered Turing-complete (or

Computationally Universal, as opposed to specialized computing devices such as scientific calculators) because they share the same fundamental model for computing results and every computer has the ability to run any algorithm. Alan Mathison Turing was an English mathematician who was highly influential in the development of theoretical computer science, which focuses on the mathematical processes behind software, and provided a formalization of the concepts of algorithm and computation with the Turing machine. A Turing-complete computer stores data in memory (either using vacuum tubes or silicon) and manipulates that data according to a computer program, which is an algorithm that can be run on a computer. These programs are represented using symbols and instructions written in a programming language consisting of symbols and instructions that can be interpreted by the computer. Programs are also stored in memory, which allows programmers to modify and improve programs by changing the instructions.

While both hardware and software are important to the practical operation of computers, computer science's historical roots in mathematics also emphasize a third perspective. Whereas software focuses on the program details for solving problems with computers, theoretical computer science focuses on the mathematical processes behind software. The idea of Turing-completeness is a foundational concept in theoretical computer science, which considers how computers in general—not just the ENIAC or today's computers, but even tomorrow's computers that we haven't yet invented—can solve problems. This theoretical perspective expands computer science knowledge by contributing ideas about (1) whether a problem can be computed by a Turingcomplete computer at all, (2) how that problem might be computed using an algorithm, and (3) how quickly or efficiently a computer can run such an algorithm. The answers to these questions suggest the limits of what we can achieve with computers from a technical perspective: Using mathematical ideas, is it possible to use a computer to compute all problems? If the answer to a problem is yes, how much of a computing resource is needed to get the answer?

Clearly both humans and computers have their strengths and limitations. An example of a problem that humans can solve but computers struggle with is interpreting subtle emotions or making moral judgments in complex social situations. While computers can process data and recognize patterns, they cannot fully understand the nuances of human emotions or ethics, which often involve context, empathy, and experience.

On the flip side, there are tasks that neither computers nor humans can perform, such as accurately predicting chaotic systems like long-term weather patterns. Despite advancements in artificial intelligence (AI), computer functions that perform tasks, such as visual perception and decision-making processes that usually are performed by human intelligence, these problems remain beyond our collective reach due to the inherent unpredictability and complexity of certain natural systems.

Theoretical computer science is often emphasized in undergraduate computer science programs because academic computer science emerged from mathematics, often to the detriment of perspectives that center on the social and technical values embodied by applications of computer technology. These perspectives, however, are gradually changing. Just as the design of ARPANET shaped the design of the Internet, computer scientists are also learning that the physical aspects of computer hardware determine what can be efficiently computed. For example, many of today's artificial intelligence technologies rely on highly specialized computer hardware that is fundamentally different at the physical level compared to the general-purpose programmable silicon that has been the traditional focus of computer science. Organizations that develop human-computer interaction (HCI), a subfield of computer science that emphasizes the social aspects of computation, now host annual conferences that bring together thousands of researchers in academia and professionals in the industry. Computer science education is another subfield that emphasizes the cognitive, social, and communal aspects of learning computer science. Although these human-centered subfields are not yet in every computer science department, their increasing representation reflects computer scientists' growing desire to serve not only more engaged students, but also a more engaged public in making sense of the values of computer technologies.

The Capabilities and Limitations of Computer Science

Computers can be understood as sources, tools, and opportunities for changing social conditions. Many people have used computer science to achieve diverse goals beyond this dominant vision for computer science. For example, consider computers in education.

Around the same time that the ARPANET began development in the late 1960s, Wally Feurzeig, Seymour Papert, and Cynthia Solomon designed the LOGO programming language to enable new kinds of computermediated expression and communication. Compared to contemporary programming languages such as FORTRAN (FORmula TRANslation System) that emphasized computation toward scientific and engineering applications, LOGO is well known for its use of turtle graphics, whereby programs were used to control the actions of a digital turtle using instructions such as moving forward some number of units and turning left or right some number of degrees. Papert argued that this turtle programming enabled body-syntonic reasoning, a kind of experience that could help students more effectively learn concepts in mathematics such as angles, distance, and geometric shapes by instructing the turtle to draw them, and physics by constructing their own understandings via reasoning through the physical motion of turtle programs by showing concepts of velocity, repeated commands to move forward the same amount; acceleration, by making the turtle move forward in increasing amounts; and even friction, by having the turtle slow down by moving forward by decreasing amounts. In this way, computers could not only be used to further education in computer science, but also offer new, more dynamic ways to learn other subjects. Papert's ideas have been expanded beyond the realm of mathematics and physics to areas such as the social sciences, where interactive data visualization can help students identify interesting correlations and patterns that precipitated social change and turning points in history while also learning new data fluencies and the limits of data-based approaches.[4]

Yet despite these roots in aspirations for computers as a medium for learning anything and everything, the study of computer science education emerged in the 1970s as a field narrowly concerned with producing more effective software engineers. Higher-education computer science faculty, motivated by the demand for

4 B. Naimipour, M. Guzdial, and T. Shreiner. 2019. Helping Social Studies Teachers to Design Learning Experiences Around Data:

Participatory Design for New Teacher-Centric Programming Languages. In Proceedings of the 2019 ACM Conference on International Computing Education Research (ICER '19). Association for Computing Machinery, New York, NY, USA, 313. DOI: https://doi.org/10.1145/3291279.3341211

software engineers, designed their computer science curricula to teach the concepts that early computer companies such as IBM desperately needed. These courses had an emphasis on efficiency, performance, and scalability, because a university computer science education was only intended to produce software engineers. We live with the consequences of this design even today: the structure of this textbook inherits the borders between concepts originally imagined in the 1970s when university computer science education was only intended to prepare students for software development jobs. We now know that there are many more roles for computer scientists to play in society—not only software engineers, but also data analysts, product managers, entrepreneurs, political advisors or politicians, environmental engineers, social activists, and scientists across every field from accounting to zoology.

Although the role of computers expanded with the introduction of the Internet in the late 1990s, Papert's vision for computation as a learning medium has been challenging to implement, at least partly because of funding constraints. But as computers evolve, primary and secondary education in the United States is striving for ways to help teachers use computers to more effectively teach all things—not just computers for their own sake, but using computers to learn everything. Computers and Racial Justice

Our histories so far have centered the interests of White American men in computer science. But there are also countless untold, marginalized histories of people of other backgrounds, races, ethnicities, and genders in computing. The book and movie Hidden Figuresshares the stories of important Black women who were not only human computers, but also some of the first computer scientists for the early digital computers that powered human spaceflight at NASA (Figure 1.5).

Figure 1.5 Katherine Johnson, a Black computer scientist, recalculated the computations done by early digital computers for space flight planning at NASA. Her contributions were portrayed in the book and movie Hidden Figures. (credit: “Katherine Johnson at NASA, in 1966” by NASA/Wikimedia Commons, Public Domain)

In one chapter of Black Software, Charlton McIlwain shares stories from “The Vanguard” of Black men and women who made a mark on computer science in its early years from the 1950s through the 1990s through the rise of personal computing and the Internet, but whose histories have largely been erased by the dominant Silicon Valley narratives. Their accomplishments include leading computer stores and developing early Internet social media platforms, news, and blog websites. For example, Roy L. Clay Sr., a member of the Silicon Valley Engineering Hall of Fame, helped Hewlett-Packard develop its first computer lab and create the company’s first computers. Later, Clay provided information to venture capitalists that motivated them to invest in start-ups such as Intel and Compaq.[5] In another example, Mark Dean was an engineer for IBM whose work was instrumental in helping IBM develop the Industry Standard Architecture (ISA) bus, which created a method of connecting a computer’s processor with other components and enabling them to communicate. This led to the creation of PCs, with Dean owning three of the nine patents used to create the original PC.[6]

Yet their efforts were often hampered by the way that computer science failed to center, or even accommodate, Black people. Historically, American Indians and Hispanic people did not have the same access as even Black Americans to computers and higher education. Kamal Al-Mansour, a technical contract negotiator at the NASA Jet Propulsion Lab, worked on space projects while Ronald Reagan was president. He recounts:

"It was conflicting . . . doing a gig . . . supporting missiles in the sky, (while) trying to find my own identity and culture . . . JPL was somewhat hostile . . . and I would come home each day [thinking] What did I accomplish that benefited people like me? And the answer every day would be 'Nothing.'"[7]

Al-Mansour would go on to start a new company, AfroLink, finding purpose in creating software that centered on Black and African history and culture. This story of computer technologies in service of African American communities is reflected in the creation of the Afronet (an early social media for connecting Black technologists) and the NetNoir (a website that sought to popularize Black culture). These examples serve as early indicators of the ways that Black technologists invented computer technologies for Black people in the United States. Yet Black Softwarealso raises challenging political implications of the historical exclusion of

Black technologists. Black culture on the Internet has greatly influenced mainstream media and culture in the United States, but these Black cultural products are ultimately driving attention and money to dominant platforms such as X and TikTok rather than those that directly benefit Black people, content creators, and entrepreneurs. Computer technologies risk reproducing social inequities through the ways in which they distribute benefits and harms.

The digital divide has emerged as a significant issue, as many aspects of society -- including education, employment, and social mobility -- become tied to computing, computer science, and connectivity. The divide refers to the uneven and unequal access and distribution of technology across populations from different geographies, socioeconomic statuses, races, ethnicities, and other differentiators. While technological access generally improves over time, communities within the United States and around the world have different levels of access to high-speed Internet, cell towers, and functioning school computers. Unreliable electricity can also play a significant role in computer and Internet usage. And beyond systemic infrastructure-based differences, individual product or service access can create a divide within communities. For example, if powerful AI-based search and optimization tools are only accessible through high-priced subscriptions, specific populations can be limited in benefiting from those tools.

5 J. Dreyfuss, "Blacks in Silicon Valley," 2011. https://www.theroot.com/blacks-in-silicon-valley-1790868140

6 IBMers, "Mark Dean," n.d. https://www.ibm.com/history/mark-dean

7 C. D. McIlwain, (2019). Black software: The Internet and racial justice, from the AfroNet to Black Lives Matter, New York: Oxford University Press.

GLOBAL ISSUES IN TECHNOLOGY

H-1B Visas Address Worker Shortages

According to the U.S. Bureau of Labor Statistics (BLS), by 2033, the number of jobs available for computer and information research scientists is expected to increase by 26%. This is much faster job growth than the average expected in total for all occupations. BLS predicts that this will result in about 3,400 job openings per year in technology, including computer science.[8]

To fill some of these jobs, U.S. employers likely will continue to rely on H-1B visas. This visa enables employers to recruit well-educated professionals from other countries. These professionals temporarily reside in the United States and work in specialty occupations, like computer science, that require a minimum education of a bachelor's degree or its equivalent.[9] To participate in the visa program, employers must register and file a petition to hire H-1B visa holders. Each year, the U.S. Citizenship and Immigration Services accepts applications from individuals from other countries who compete for a pool of 65,000 visa numbers, as well as an additional pool of 20,000 master's exemption visa numbers awarded that year and valid for a period of three years. At the end of three years, employers can petition to have each worker's visa extended for a period of three additional years.[10] This program helps U.S. employers fill vacancies in many fields, including computer science while providing job opportunities for highly skilled workers around the world.

Computers and Global Development

Computer technology, like any other cutting-edge technology, changes the balance of power in society. But access to new technologies is rarely ever equal. Computer science has improved the quality of life for many people who have access to computer technology and the means of controlling it to serve their interests. But for everyone else in the world, particularly people living in the Global South, computer technologies need context-sensitive designs to meet their needs. In the 1990s, for instance, consumer access to the Internet was primarily based on "dial-up" systems that ran on top of public telephone network systems. Yet many parts of the world, even today, lack telephone coverage, let alone Internet connectivity. Research in computers for global development aims to improve the quality of life for people all over the world by designing computer solutions for lowincome and underserved populations across the world—not just those living in the wealthiest countries.

Computer technologies for global development require designing around unique resource constraints such as a lack of reliable power, limited or nonexistent Internet connectivity, and low literacy. Computer scientists employ a variety of methods drawing from the social sciences to produce effective solutions. However, designing for diverse communities is difficult, particularly when the designers have little direct experience with the people they wish to serve. In The Charisma Machine, Morgan Ames criticizes the One Laptop Per Child (OLPC) project, a nonprofit initiative announced in 2005 by the Massachusetts Institute of Technology Media Lab. The project attempted to bring computer technology in the form of small, sturdy, and cheap laptops that were powered by a hand crank to children in the Global South. Based on her fieldwork in Paraguay, Ames argues that the project failed to achieve its goals for a variety of reasons, such as electricity infrastructure problems, hardware reliability issues, software frustrations, and a lack of curricular materials. Ames argues that “charismatic technologies are deceptive: they make both technological adoption and social change appear straightforward instead of as a difficult process fraught with choices and politics.” When the computers did

---

8 U.S. Bureau of Labor Statistics, “Computer and Information Research Scientists: Job Outlook,” 2024. https://www.bls.gov/ooh/ computer-and-information-technology/computer-and-information-research-scientists.htm#tab-6

9 U.S. Citizenship and Immigration Services, “H-1B Specialty Occupations,” 2024. https://www.uscis.gov/working-in-the-unitedstates/h-1b-specialty-occupations

10 American Immigration Council, “The H-1B Visa Program and Its Impact on the U.S. Economy,” 2024. https://www.americanimmigrationcouncil.org/research/h1b-visa-program-fact-sheet

work, OLPC’s vision for education never truly materialized because children often used the computers for their own entertainment rather than the learning experiences the designers intended. Though Ames’s account of the OLPC project (Figure 1.6) itself has been criticized for presenting an oversimplified narrative, it still represents a valuable argument for the risks and potential pitfalls associated with designing technologies for global development: technology does not act on its own but is embedded in a complicated social context and history.

Figure 1.6 In 2005, MIT's Media Lab started the OLPC initiative to bring laptops to children in the Global South. An unexpected outcome they discovered was that designing technologies for global communities is not as straightforward as designers may initially believe. (credit: "One Laptop per Child" by OLE Nepal Cover/Flickr, CC BY 2.0)

THINK IT THROUGH

Internet Commerce

Many products and companies offer services or products over the Internet. While online shopping provides additional sales opportunities for businesses, while offering consumers a convenient shopping option, it is not without risks. For example, online businesses and their shoppers may be victims of data breaches and identity theft. Other risks include fake reviews that motivate consumers to make a purchase, phishing that leads to hacking, and fake online stores that take consumers' money without delivering a product. What can we do to mitigate the risks and dangers of online shopping?

Addressing these risks is not as simple as practicing humility and including communities in the design process. Many challenges in computing for global development are sociopolitical or technopolitical rather than purely technical. For example, carrying out a pilot test to evaluate the effectiveness of a design can appear as favoritism toward the pilot group participants. These issues and social tensions are especially exacerbated in the Global South, where the legacies of imperialism and racial hierarchies continue to produce or expand social inequities and injustices.

The identities of people creating computer technologies for global development are ultimately just as important as the technologies they create. In Design Justice, Sasha Costanza-Chock reiterates the call for computer scientists to "build with, not for," the communities they wish to improve. In this way, Design Justice seeks to address the social justice tensions raised when asking the question, “Who does technology ultimately benefit?” by centering the ingenuity of the marginalized “user” rather than the dominant “designer.”

In some cases, underdeveloped countries can quickly catch up without spending the money that was invested to develop the original technologies. For example, we can set up ad hoc networks quickly today and at a portion of the cost in Middle Eastern and African countries using technology that was developed (at a high cost) in the United States and Europe over the past several decades. This means that sometimes, progress in one part of the world can be shared with another part of the world, enabling that area to quickly progress and advance technologically.

LINK TO LEARNING

The Design Justice Network (https://openstax.org/r/76DesignJust) is an organization that aims to advance the principles of design justice and to include people who are marginalized in the technology design process.

1.2 Computer Science across the Disciplines

Learning Objectives

By the end of this section, you will be able to:

- Differentiate between discovery and invention
- Describe how science, mathematics, and engineering each play a role in computer science
- Discuss how data science, computational science, and information science each relate to computer science
- Explain why the various areas of computer science are synergistic

Computer science is an incredibly diverse field not because of what it can achieve on its own but because of how it contributes to every other field of human knowledge and expertise. From its early days, it was understood that there would be cross-collaboration between computer scientists and colleagues in other disciplines. Today, almost all modern technologies either depend on computer technologies or benefit significantly from them. Computer technologies and the study of computer science have reshaped almost all facets of life today for everyone.

Data Science

Across business, financial, governmental, scientific, and nonprofit workplaces, millions of people are programming, and most of the time, they don't even know it! A spreadsheet is an example of a data-centric programming environment where data is organized into cells in a table. Instead of presenting programs as primarily about algorithms, spreadsheets present programs as primarily about data. Spreadsheets are often used for data analysis by offering a way to organize, share, and communicate ideas about data. Spreadsheets are uniquely effective and accessible because they allow for the visual organization of data in whichever structure makes the most sense to the user. Instead of hiding data behind code, spreadsheets make data as transparent and up to date as possible.

Although spreadsheets make computation accessible for millions of people across the world, they have several shortcomings. Unless limits are removed, many popular spreadsheet software products such as Microsoft Excel may have a limitation to the number of rows of data they can store that is less than the data of modern computers. One such example occurred in October 2020 when Public Health England failed to report 15,841 positive cases of COVID-19 in the United Kingdom due to mismanaged row limits in the spreadsheet used. This shortcoming attests not only to the technical limit on the number of rows supported by spreadsheets, but also to the design limitations of software that fails to communicate data loss, irregularities, or errors to users. Errors in spreadsheet software data entry can often go unnoticed because spreadsheets do not enforce data types. Cells can contain any content: numbers, currencies, names, percentages, labels, and legends. The meaning of a cell is determined largely by the user rather than the software. Spreadsheets are an expressive and accessible technology for data analysis, but this creative power that spreadsheets afford to users is the very same power that limits spreadsheets as a data management and large-scale data analysis tool. The more data and the more people involved in a spreadsheet, the greater the potential for spreadsheet problems.

The interdisciplinary field that applies computing to managing data and extracting information from data is called data science. Data scientists are practitioners who combine computing and data analysis skills with the domain knowledge specific to their field or business. The demand for data scientists is becoming increasingly important as more and more research and business contexts involve analyzing big data, or very large datasets that are not easily processed using spreadsheets. These datasets often involve high-volume measurements of user interactions on the Internet at a very fine grain, such as tracking a customer's web browser history across an online storefront. Data scientists can then analyze browser patterns using machine learning methods in order to recommend related products, target advertisements to specific customers over social media, and reengage customers over email or other means. Machine learning (ML) is a subset of artificial intelligence that relies on algorithms and data to make it possible for artificial intelligence to learn, actually mimicking the way humans learn. For example, ML is used to identify fraudulent versus legitimate banking transactions. Once a computer learns how to distinguish fraudulent transactions, it can be alert and call attention to suspicious banking activity.

GLOBAL ISSUES IN TECHNOLOGY

Targeted Advertising

Although data scientists can produce immense value for business and research alike, their work also raises significant social concerns. For example, web browser history tracking enables companies to target advertising of products to people and also allows for targeting of political advertisements. In Antisocial Media, Siva Vaidhyanathan argues that the "impact of Facebook on democracy is corrosive [because political campaigns] can issue small, cheap advertisements via platforms like Facebook and Instagram that may target "groups as small as twenty, and then disappear, so they are never examined or debated." This undermines the process of discussions among voters in democracies like the United States, as well as countries like Germany, United Kingdom, and Spain, which spent the most on targeted political advertising on Facebook in the spring of 2019.[11] Given their lack of transparency, such ads are a questionable practice.

Another interesting topic worth mentioning here is targeted advertising toward children.[12] It is important to consider the ethical implications of using the data collected through tracking, especially when it comes to targeting at-risk populations. It raises questions about the accountability of platforms and advertisers in safeguarding users' rights and ensuring transparency in how data is used for these purposes.

Computational Science

Beyond data science, computer science can also fundamentally change how science is researched and developed. The field of computational science refers to the application of computing concepts and technologies to advance scientific research and practical applications of scientific knowledge in a wide range of fields, including civil engineering, finance, and medicine (among many others). For example, algorithms and computer software play a key role in enabling numerical weather prediction (Figure 1.7) or the use of mathematical models to forecast weather based on current conditions in order to assist peoples' everyday lives and contribute to our understanding of the climate models, climate changes, and climate catastrophes. These algorithms may rely on a large amount of computer hardware power that might not be available in a

[11] Statista, "European Elections: Countries that spent the most on targeted political advertising on Facebook from March 1 to May

26, 2019*," 2019. https://www.statista.com/statistics/1037329/targeted-political-ad-spend-on-facebook-by-eu-countries/

[12] Maya Brownstein. Harvard study is first to estimate annual ad revenue attributable to young users of these platforms. January 2,

2024. https://news.harvard.edu/gazette/story/2024/01/social-media-platforms-make-11b-in-ad-revenue-from-u-s-teens/

single system, so the work may need to be distributed across many computers. Computational science studies methods for realizing these algorithms and computer software.

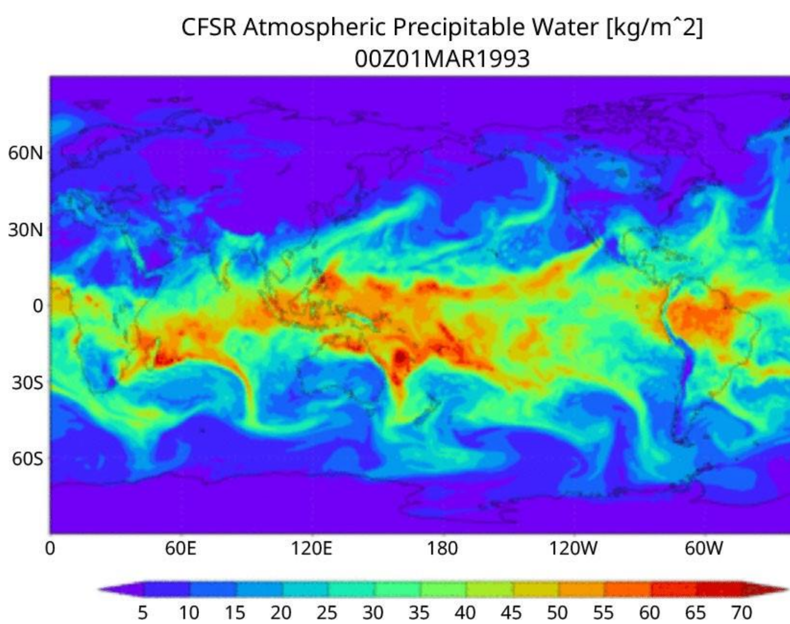

Figure 1.7 Meteorologists collect data from a variety of sources and use the data, algorithms, and computers to predict the weather.

(data source: Climate Forecast System, National Centers for Environmental Information, National Oceanic and Atmospheric

Administration, https://www.ncei.noaa.gov/products/weather-climate-models/climate-forecast-system; credit: modification of "CFSR

Atmospheric Precipitable Water" by NOAA/ncei.noaa.gov, Public Domain)

INDUSTRY SPOTLIGHT

Computer Science and Climate Change

Computer science fights climate change and limits the impacts of climate catastrophes by enabling technologies for decarbonization through power consumption optimization and advancing renewable energy sources. Numerical weather forecasting not only supports our everyday lives but also helps climate scientists determine the precise locations for wind turbines and simulate how they should be designed to enable the greatest energy production. To support the power grid, data science methods can help predict peak power consumption, optimize power sources to produce exactly the right amount of power needed, and adjust power storage to reduce the amount of energy that needs to be generated from nonrenewable sources. Computer models and algorithms assist energy engineers in optimizing building air conditioning and power demands so that they efficiently serve the people living, working, and playing within them.

Although computer science has been used to support scientific discovery, the theory of knowledge of computer science has historically been considered quite different from that of the natural sciences, such as biology, physics, and chemistry. Computer science does not study natural objects, so to most, it would not be considered a natural science but rather an applied science. Unlike natural sciences such as biology, physics, and chemistry, which emphasize the discoveryof natural phenomena, computer science often emphasizes

inventionor engineering.

However, computer science is today deeply interdisciplinary and involves methods from across science, mathematics, and engineering. Computer scientists design, analyze, and evaluate computational structures, systems, and processes.

- Mathematics plays a key role in theoretical computer science, which emphasizes how a computational problem can be defined in mathematical terms and whether that mathematical problem can be efficiently

solved with a computer.

- Engineering plays a key role in software engineering, which emphasizes how problems can be solved with computers as well as the practices and processes that can help people design more effective software solutions.
- Science plays a key role in human-computer interaction, which emphasizes experimentation and evaluation of the interface (boundary) between humans and computers, often toward designing better computer systems.

Information Science

Not only is computation interdisciplinary, but other disciplines are also becoming more and more computational. In The Invisible Future, Nobel Laureate biologist David Baltimore defines DNA in computational terms. He states that biology is an information science because DNA encodes for the outputs of biological systems. The interdisciplinary field studying information technologies and systems as they relate to people, organizations, and societies is called information science. The role of information in natural sciences can also be found in the physics of quantum waves that carry information about physical effects, in the chemical equations that specify information about chemical reactions, in the information flows that drive the evolution of economies and political organizations, and in the information processes underlying social, management, and communication sciences.[13]

CONCEPTS IN PRACTICE

Computer Science and DNA

Research into DNA sequencing and indexing is opening new ways of helping medical providers offer personalized treatments for patients. Large-scale genome sequencing of not only the human genome, but also the DNA signatures for viruses has made it possible for medical providers to take human fluid samples and analyze them for the presence of infectious diseases. This research requires computer science concepts, including specialized medical computer devices to sequence the billions of nucleotides that form a DNA sequence, data structures and algorithms to efficiently process and identify DNA signatures, and the miniaturization of computer hardware so that this technology is accessible (both in terms of price and physical size) in more and more care centers.

Although information science has its roots in information classification, categorization, and management in the context of library systems, information science today is a broad field that encompasses the many diverse ways information shapes society. For example, today's social media networks provide more personable and instantaneous information communication compared to traditional news outlets—billions of people around the world are using social media to engage with information about the world. For many people, social media may be the primary way that they learn about and make sense of the world (Figure 1.8). Yet, we've already seen risks associated with information technologies such as the Internet. In today's "information age," information has more power than ever before to reshape society. Information scientists, data scientists, computational scientists—and, therefore, computer scientists—have a social responsibility: “One cannot reap the reward when things go right but downplay the responsibility when things go wrong.”[14]

---

13 P. J. Denning, “Computing is a natural science,” Commun. ACM, vol. 50, no. 7, pp. 13–18, July 2007. https://doi.org/10.1145/ 1272516.1272529.

14 R. Benjamin, “Race after technology: Abolitionist tools for the new Jim code,” 2019, Polity.

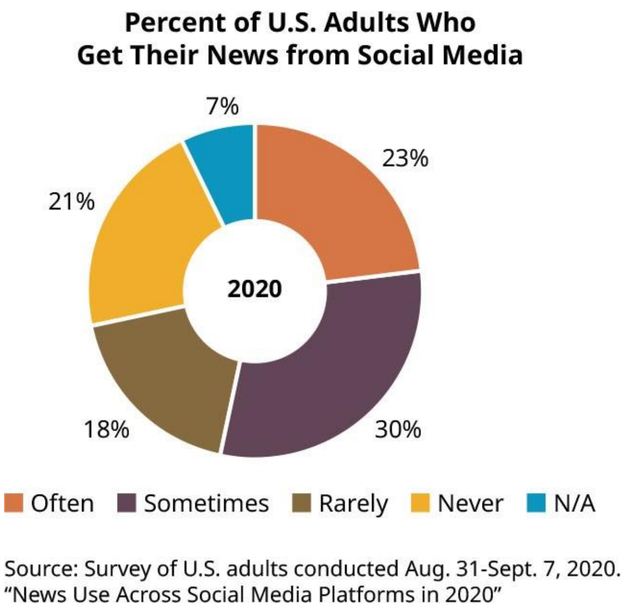

Figure 1.8 While Americans used to primarily get their news from newspapers, as technology has advanced their primary source of news media has shifted. As of 2020, 53% of American adults surveyed stated they got their news from social media at least some of the time. (data source: Elisa Shearer and Amy Mitchell, Pew Research Center. "About Half of Americans Get News on Social Media at

Least Sometimes." From Survey of U.S. adults conducted Aug. 31–Sept. 7, 2020. In: E. Shearer, A. Mitchell, "News Use Across Social Media Platforms in 2020," Jan 12, 2021.; attribution: Copyright Rice University, OpenStax, under CC BY 4.0 license)

Despite the centrality of information to decision-making and social change, dominant approaches to computer science tend to focus on computational structures, systems, and processes (such as algorithms) that describe one kind of information by focusing on the whator howof solving problems with computers, but less often the whyor whoquestions. Information science broadly centers people, organizations, and society in the study of information technologies.

Computer Science Is an Interdisciplinary Field

In presenting data science, computational science, and information science, we’ve introduced the idea that computer science can shape other disciplines. But we’ve also raised questions about what computer science is today. If computer science is the study of all “phenomena surrounding computers,” it could also involve data science, computational science, bioinformatics, cheminformatics, computational social science, medical informatics, and information science. As you will learn in Chapter 13 Hybrid Multicloud Digital Solutions Development, another aspect of computer science is responsible computing, which includes the appropriate management of cyber resources as well as robust cybersecurity. It is difficult to define computer science today because it is so widely used by people across the world in diverse capacities. Definitions are about defining boundaries and excluding practices, which may be helpful for understanding the practices of a certain culture or group that is “doing” computer science, but it can never truly represent everyone and all the things that people are doing with computer science. However, computer science’s historical roots in mathematics shape the way it categorizes subfields:

- Theoretical computer science ◦ Theory of computation

◦ Information representation

◦ Data structures and algorithms

◦ Programming language and formal methods

- Computer systems ◦ Architecture

◦ Artificial intelligence

◦ Networks

◦ Security

◦ Databases

◦ Distributed computing

◦ Graphics

- Applied computer science

◦ Scientific computing

◦ Human-computer interaction

◦ Software engineering

In this hierarchy, theoretical computer science and computer systems are treated separately from applied computer science and human-computer interaction, suggesting that the mathematics of computing are pure and separate from social questions. Yet we've seen several examples that question this paradigm and instead point to a structure where human-computer interaction is infused throughout the study of computer science and all its subfields.

Today, computer science is a field that is just as much about people as it is about computer technology because each of these subfields is motivated by the real-world problems that people ultimately want to solve. The subfields of artificial intelligence and machine learning have applications that directly influence human decision-making, ranging from advertisement targeting to language translation to self-driving cars. Effective computational solutions to research or business problems require combining specific knowledge with computer science concepts from a combination of areas. For example, the computational science application of weather prediction combines knowledge about various subfields of computer science (algorithms, distributed computing, computer systems) with knowledge about climate systems. Theoretical computer scientists are increasingly interested in asking questions such as, "How do we design, analyze, and evaluate algorithms or information systems for fairness? How do we even define fairness in a computer system that strips away the complexities of the real world? What ideas or information are encoded in the data? And what are the limits of our approaches?" Computer science is a complex field, and its synergistic nature means that when computer science is used in an interdisciplinary manner that shapes other disciplines, its impact on society is much greater than when each discipline functions on its own.

1.3 Computer Science and the Future of Society

Learning Objectives

By the end of this section, you will be able to:

- Discuss how computer scientists develop foundational technologies
- Discuss how computer scientists evaluate the negative consequences of technologies
- Discuss how computer scientists design technologies for social good

As noted earlier, computer science is a powerful tool, and computer scientists have vast technological knowledge that continues transforming society. Computer scientists have an obligation to be ethical and good stewards of technology with an emphasis on responsible computing. Written code influences daily life, from what we see on social media to the news stories that pop up in a Google search and even who may or may not receive a job interview. When computer scientists don't consider the ramifications of their code, there can be unintended consequences for people around the world. The Y2K problem, also known as the "millennium bug," is a good example of shortsighted decisions that allowed computer scientists to only store the last two digits of the year instead of four. This made sense at a time when memory was expensive on both mainframe computers and early versions of personal computers. The Y2K problem was subsequently coined by John Hamre, the United States Deputy Secretary of Defense, as the "electronic equivalent of the El Niño."[15] The future of computer science will highly affect the future of the world. Although we often think of computer technologies as changing the way the world works, it is actually people and their vision for the future that are amplified by computing. The relationship between computer science and people is about how computer technologies can bias society and how the choices made through computer systems can both promote and discourage social inequities. Computer technologies can encode either value or both values in their designs. In this section, we'll introduce three ways that computer science can shape the future of society: developing foundational technologies, evaluating negative consequences of technologies, and designing technologies for social good.

Developing Foundational Technologies

We've seen how foundational technologies like artificial intelligence, algorithms, and mathematical models enable important applications in data science, computational science, and information science.

As noted previously, artificial intelligence (AI) is the development of computer functions to perform tasks, such as visual perception and decision-making processes, that usually are performed by human intelligence. AI refers to a subfield of CS that is interested in solving problems that require the application of machine learning to human cognitive processes to achieve goals. AI research seeks to develop algorithm architectures that can make progress toward solving problems. One such example is image recognition, or the problem of identifying objects in an image. This problem is quite difficult for programmers to solve using traditional programming methods. Imagine having to define very precise rules or instructions that could identify an object in an image regardless of its position, size, lighting conditions, or perspective. As humans, we have an intuitive sense of the qualities of an object. However, representing this human intelligence in a machine requiring strict rules or instructions is a much harder task. AI methods for image recognition involve designing algorithm architectures that can generalize across all the possible ways that an object can appear in an image. INDUSTRY SPOTLIGHT

Agricultural Robots

Agricultural robots help large-scale industrial farmers produce crops more efficiently and support sustainability efforts. One agricultural robot is now being used to improve fertilizer and pesticide treatments by taking pictures of plants as a farmer drives a tractor over the field. Artificial intelligence techniques are used to recognize and identify the lettuce plants and weed plants in the image. For each identified lettuce or weed plant, the robot makes a personalized decision about the best chemical treatment for the plant in real time as the tractor moves to the next row of crops. This ability to personalize chemical treatments improves yields and plant quality for large-scale industrial agriculture by producing more crops with fewer chemicals.

---

15 “Looking at the Y2K bug,” portal on CNN.com. Archived 7 February 2006 at the Wayback Machine. https://web.archive.org/web/20060207191845/http://www.cnn.com/TECH/specials/y2k/

Recent approaches to AI for image recognition draw on a family of methods called neural networks instead of having programmers craft rules or instructions by hand to form an algorithm. In humans, the neural network is a complex network in the human brain that consists of neurons, or nerve cells, connected by synapses that send messages and electrical signals to all parts of the body, enabling us to think, move, feel, and function.

In computer science, a neural network (Figure 1.9) is an AI algorithm architecture that emphasizes connections between artificial nerve cells whose behavior and values change in response to stimulus or input. These neural networks are not defined by individual neurons but by the combination of all the neurons in the network. Typically, artificial neurons are arranged in a hierarchy that aims to capture the structure of an image. Although the first level of neurons might respond to individual pixels, later levels of artificial neurons might respond in aggregate to the arrangement of several artificial neurons in the preceding layer. This is similar to how the human visual system responds to edges at the lower levels, then responds in aggregate to the specific arrangement of several edges in later levels, and ultimately identifies these aggregated arrangements as objects.

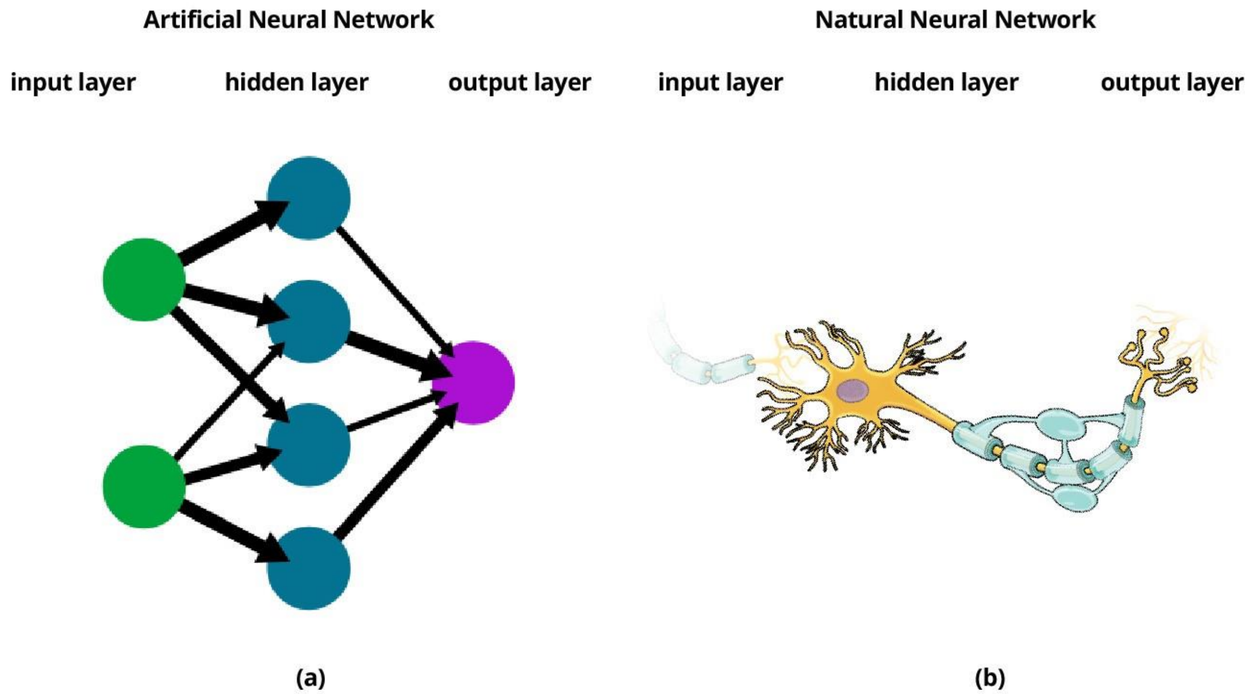

Figure 1.9 (a) An artificial neural network consists of three key layers: the input layer, where raw data enters the system; the hidden layer, where information is processed and patterns are identified; and the output layer, where results are presented. (b) A natural neural network, such as those in the human body, mirrors this structure. The input layer represents sensory receptors, like those in the retina. The hidden layer corresponds to the synapse, where partial processing of the sensory data occurs. Finally, the output layer represents the information sent to the brain for final processing and interpretation. (credit a: modification of "Neural network example" by Wiso/Wikipedia, Public Domain; credit b: modification of work from Psychology 2e. attribution: Copyright Rice

University, OpenStax, under CC BY 4.0 license)

The idea of neural networks, however, is not as new as it might seem. Artificial neural networks were first imagined in the mid-1900s alongside contemporary research efforts in the cognitive sciences. The ideas of multilayered, hierarchical networks of neurons and the mathematical optimization methods for learning were all there, but these early efforts were limited by the computational processing power available at the time. In addition, the large datasets that drive neural network learning were not nearly as available as they are today with the Internet. Developments in foundational technologies such as computer architecture and computer networks paved the way for the more recent developments in neural network technologies. The broad area of computer systems investigates these architectures and networks that enable new algorithms and software. Without these technologies, neural networks would not be nearly as popular and revolutionary as they are today. Yet the relationship between computer systems and AI development is not one-directional. Today, computer scientists are using neural networks to help design new, more efficient computer systems. The development of foundational computer technologies not only creates opportunities for direct and indirect applications, but also supports the development of other computer technologies.

Just as we saw how technological fixes embodied a powerful belief about the relationship between computer solutions and social good, a similar cultural belief exists about the relationship between foundational technologies and their social values. The belief that technologies are inherently neutral and that it is the people using technology who ultimately make it "good" or "bad" is considered social determination of technology.

THINK IT THROUGH

Social Determination of Technology

Do you agree with the social determination of technology? Is it possible for technology—before it is used by people to solve certain problems—to encode social values? Try to come up with an example that would support this belief. What about an example that refutes this belief? What are the social implications of agreeing or disagreeing with the social determination of technology?

Today's neural networks are designed to identify patterns and reproduce existing data. It is widely accepted that many big datasets can encode social preferences and values, particularly when the data is collected from users on the Internet. A social determination of technology accepts this explanation of AI bias and leaves the design of AI algorithms and techniques as neutral: the bias in an AI system is attributed to the social values of the data rather than the design of the AI algorithms. Critics of social determination point out that the way AI algorithms learn from big data represents a social value, one that encodes a default preference for reproducing the biases inherent in big data. This applies whether the AI application is about fair housing, medical imaging, ad targeting, drone strikes, or another topic. This is an issue that computer scientists must consider as they practice responsible computing and strive to ensure that data is gathered and handled as ethically as possible.

Evaluating Negative Consequences of Technology

Today's AI technologies work by reproducing existing patterns rather than imagining radically different futures. As much as neural networks are inspired by the human brain, it would be a stretch to suggest that AI systems have any semblance of general intelligence. Though these systems might be quite effective at identifying lettuce plants from weed plants in an image, their capacity for humanlike intelligence is limited by design. A neural network learns to recognize similar patterns that appear across millions or billions of sample images and represent these patterns with millions or billions of numbers. Mathematical optimization methods are used to choose the numeric values that best encode correlations across the sample images. However, current approaches lack a deeper, conceptual representation of objects. One criticism of very large neural networks is that there are often more numeric values than there are sample images—the network can effectively memorize the details of a million sample images by encoding them in a billion numbers. Many of today's neural networks recognize objects in images not by relying on some intrinsic idea or concept of objects but by memorizing every single configuration of edges as they appear in the sample images.

This limitation can lead to peculiar outcomes for image recognition systems. Often, neural network approaches for image recognition have certain examples of images where objects are misidentified in unusual ways: a person's face might be recognized in a piece of toast or in a bunch of clouds in the sky. In these examples, the pattern of edges might coincidentally trigger the neural network values so that it misidentifies objects. These are among the more human-understandable examples; there are many other odd situations that are less explainable. An adversarial attack is a sample input (e.g., an image) that is designed to cause a system to behave problematically. Researchers have found that even tweaking the color of just a single point in an image can cause a chain reaction in the neural network, leading it to severely misidentify objects. The adversary can choose the color of the point in such a way as to almost entirely control the output of some neural networks: changing a single specific point in an image of a dog might cause the system to recognize the object as a car, airplane, human, or almost anything that the adversary so desires. Moreover, these adversarial attacks can often be engineered to cause the neural network to report extremely high confidence in its wrong answers. Self-driving cars that use neural networks for image recognition can be at risk of realworld adversarial attacks when specially designed stickers are placed on signs that cause the system to recognize a red light as a green light (Figure 1.10). By studying adversarial attacks, researchers can design neural networks that are more robust and resilient to these attacks.

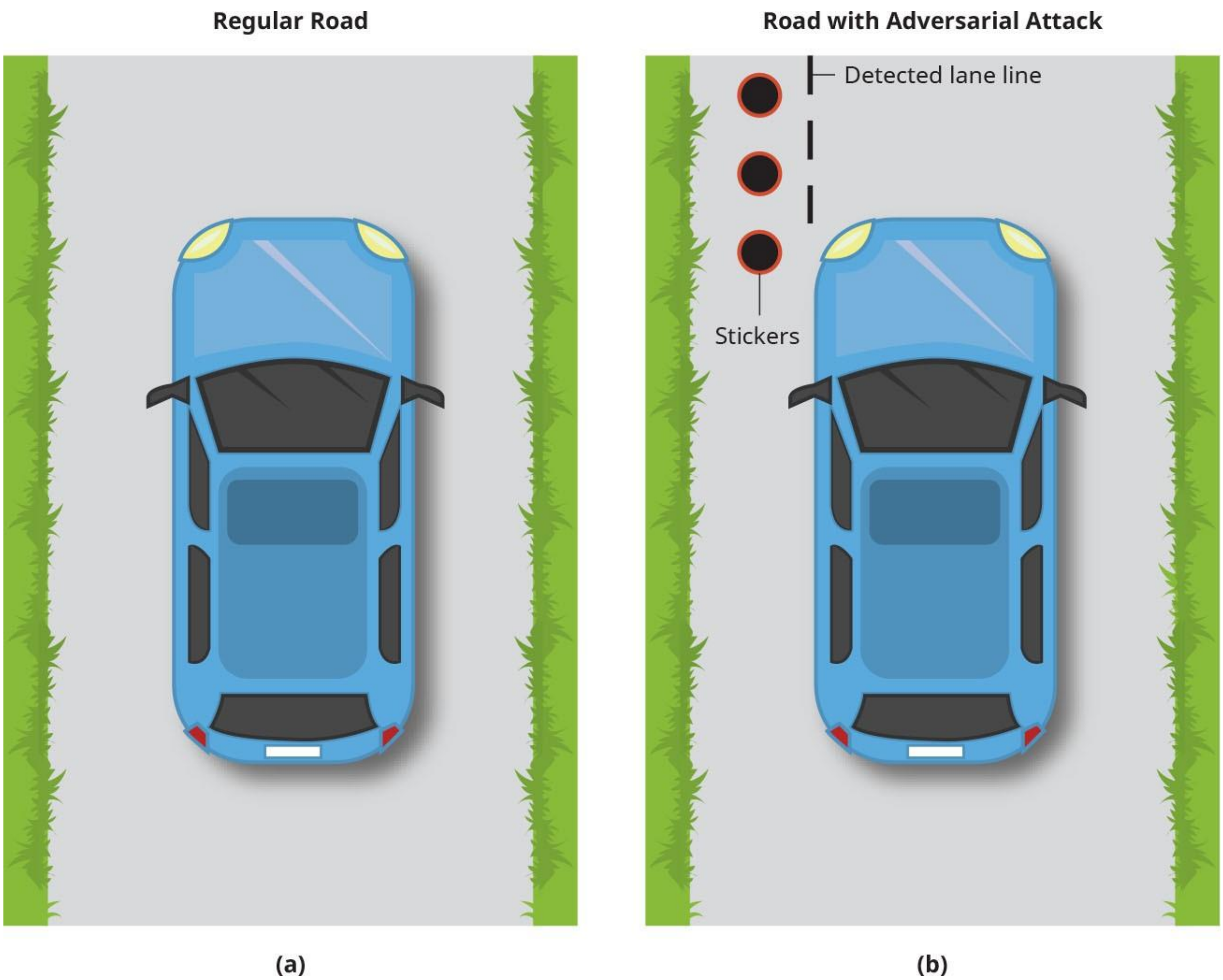

Figure 1.10 (a) Autopilot functions in self-driving cars generally identify roads and lanes using artificial intelligence to “see” road markings. (b) Researchers were able to trick these cars into seeing new lanes by using as few as three small stickers, to confuse the neural networks and force the cars to change lanes. (attribution: Copyright Rice University, OpenStax, under CC BY 4.0 license) In general, research is an important part of computer science. Through research, computer scientists analyze ways that technology can be used and gain insight and answers to address issues and improve various aspects of society. Research enables computer scientists to make advancements like the design of new algorithms, development of new hardware and software, and applications for emerging technologies such as AI.

One important use of research is to investigate adversarial attacks to gather answers needed for computer scientists to improve foundational technologies by evaluating the negative consequences of technologies. Computer technologies offer a unique medium for learning things (not just learning computer science), connecting with each other, and enhancing the lives of people all around the world. Yet, in each of these examples, we also raised concerns about how these technologies unfolded and affected people’s lives in both positive and negative ways. While we can rarely, if ever, paint any one technology as purely “good” or “bad,” computer scientists are interested in studying questions around how technologies are designed to center social values. Social scientists are not solely responsible for answering questions about technology, but computer scientists can also contribute important knowledge and methods toward understanding computer technologies.

Designing Technologies for Social Good

Computer science can advance social good by benefiting many people in many different areas, including public health, agricultural sustainability, climate sustainability, and education.

Computer technologies accelerate medical treatments for public and personal health from initial research and development to clinical trials to large-scale production and distribution. In January 2020, Chinese officials posted the genetic sequence of the coronavirus SARS-CoV-2. This helped pharmaceutical companies to begin developing potential vaccines for the virus at a significantly faster rate than for any other virus in the past (Figure 1.11).

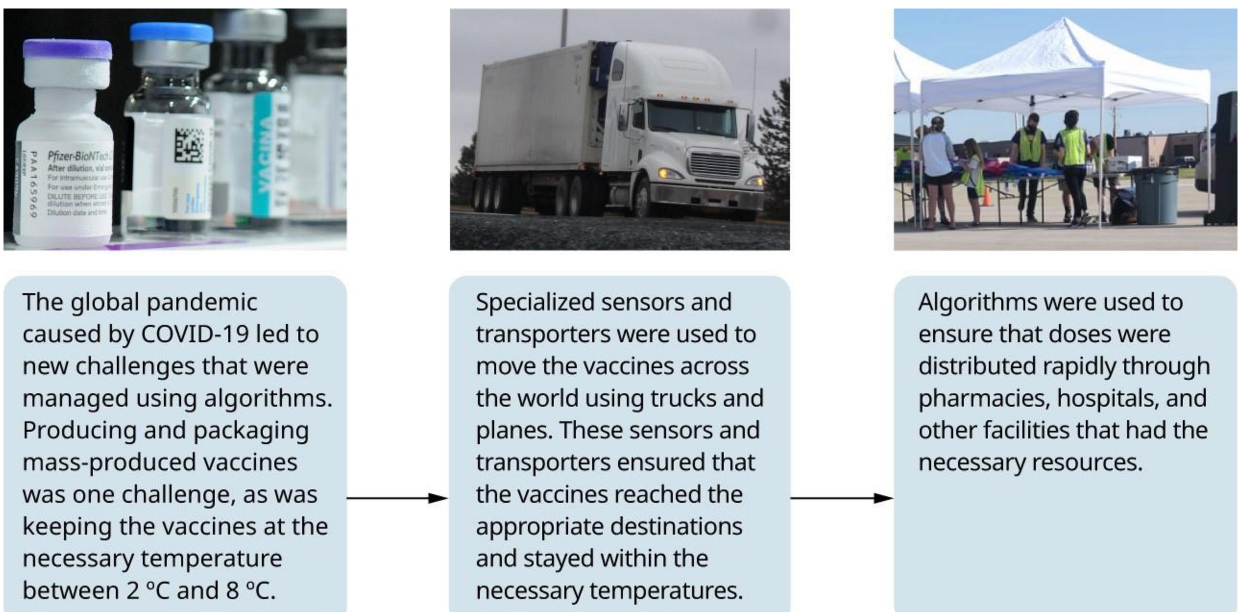

Figure 1.11 The SARS-CoV-2 outbreak that began in 2020 displayed how quickly computer science could be harnessed by governments, medical facilities, and scientists to decode the virus, develop treatments, and distribute vaccinations around the world. What would have been a very difficult feat to manage manually was simplified through the use of algorithms and computer technology. (credit left: modification of "COVID-19 vaccines" by Agência Brasília/Flickr, CC BY 2.0; credit center: modification of "T04" by Sarah Taylor/Flickr, CC BY 2.0; credit right: modification of “Back2School Brigade prepares families for upcoming school year” by

Thomas Karol/DVIDS, Public Domain)

Computational science enables the miracles of modern medicine. Viral sequences can be digitized and rapidly shared between researchers across the world via the Internet. Computer algorithms and models can simulate the human immune system responses to particular treatments within hours rather than years. The first treatments can then be produced at a small scale using computer-engineered cells in less than a month from the initial sequencing. To ensure the treatments are safe and effective, clinical trials are held at disease transmission “hot spots” predicted using data science methods drawing on data aggregated and monitored from across the world. Once a treatment is proven safe and effective, it is mass-produced with the help of computer-controlled robots and automated assembly lines. Algorithms manage the inventory supply and demand and control the transportation of treatments on trucks and planes guided by computer navigation systems. Web apps and services notify people throughout the process.

Yet the use of computer technology throughout modern medicine is anything but politically neutral. Computers, algorithms, and mathematical models solve the problems that their creators wish to solve and encode the assumptions of their target populations. Supply and demand data for the data models are determined by various factors, at least partly in response to the money and relationships between countries that control the technology, the Global North, and countries that don’t, the Global South. Within local communities, the uptake of medical treatments is often inequitable, reflecting and reinforcing historical inequities and disparities in public health. Computer technology alone often doesn’t address these issues. In fact, without people thinking about these issues, computer technologies can often amplify disparities. Consider datasets, which can be biased if they overrepresent or underrepresent specific groups of people. If decisions are made on the basis of biased data, people in the groups that are not represented fairly may receive inequitable treatment. For example, if a local government agency is working with a biased dataset, political leaders may make decisions that result in certain citizens receiving inadequate funding or services. This is an example of why responsible computing, which we will cover in Chapter 14 Cyber Resources Qualities and Cyber Computing Governance, is so important.

These problematic histories are not only aggravated in medicine and public health, but also reflected in housing. Redlining refers to the inequitable access to basic public services based on residents’ neighborhoods and communities, which includes the practice of withholding financial services from areas with a large underrepresented population. In the United States, these communities reflect the histories of racial segregation and racial wealth inequalities. Fair housing laws are intended to prevent property owners from discriminating against buyers or renters because of race, color, ability, national origin, and other protected classes. But computer technologies also present new kinds of challenges.

Microtargeted ads on social media platforms contribute to not only political polarization, but also discrimination in housing. This can be a particular problem when combined with redlining. Even if the ad targeting is not explicitly designed to discriminate, microtargeted ads can still reinforce historical redlining by incorporating data such as zip codes or neighborhoods. This may result in digital redlining, which is the practice of using technology, such as targeted ads, to promote discrimination. In 2021, a Facebook user filed a class-action lawsuit that argued nine companies in the Washington, D.C., area deliberately excluded people over the age of 50 from seeing their advertisements for housing because they wanted to attract younger people to live in their apartments.[16] This is an example of an issue in technology that should be addressed by responsible computing with an emphasis on ethical behavior.

With good intentions and attention to personal biases, technologies can be designed for social good. For example, a hypothetical algorithm for fair housing could evenly distribute new housing to people across protected classes and marginalized identities, such as older populations. Of course, algorithmic control and automated decision-making is challenged to consider the underlying conditions behind social problems. Still, algorithms can be important tools to enable us to distribute outcomes more fairly from a statistical perspective, and this can be an important step in addressing the larger societal systems and inequities that produce social problems.

LINK TO LEARNING

Review the [Parable of the Polygons (https://openstax.org/r/76polygons)](https://openstax.org/r/76polygons) by Vi Hart and Nicky Case. In it, the authors show how a segregated world where people simply prefer living near other people who are like themselves (a “small individual bias”) can re-create and reproduce “large collective bias.”

As part of responsible computing, computer scientists must be aware of technological fix, which refers to the idea that technologies can solve social problems, but is now often used to critique blind faith in technological solutions to human problems. Unless the process is handled responsibly, the “fix” may cause more problems than it resolves. When considering how to address social and political problems, computer scientists must take care to ensure that they select the appropriate technology to address specific problems.

16 C. Silva, “Facebook ads have a problem. It’s called digital redlining,” 2022. https://mashable.com/article/facebook-digitalredlining-ads-protected-traits-section-230

To address social problems and advance social good, recall that human-centered computing emphasizes people rather than technologies in the design of computer solutions. A human-centered approach to fair housing might begin by centering local communities directly affected by redlining. Rather than replacing or disrupting the people and organizations already working on a problem, a human-centered approach would center them in the design process as experts. A human-centered approach requires that the designer ask why they are not already working with people in the community impacted by their work.

LINK TO LEARNING

Anatomy of an AI System (https://openstax.org/r/76AIanatomy) illustrates how an AI system like the Amazon Echo does not just involve computer technology, but also involves a vast and deeply interconnected web of human labor, data, and physical resources that are often taken for granted. Evaluation of the negative consequences of technology does not end at the technology itself, but also considers its broad-reaching impacts and implications for people around the world.

Chapter Review

Key Terms

adversarial attack sample input (e.g., an image) that is designed to cause a system to behave problematically

algorithm sequence of precise instructions

artificial intelligence (AI) development of computer functions to perform tasks, such as visual perception and decision-making processes, that usually are performed by human intelligence

big data very large datasets that aren't easily processed using spreadsheets

computational science application of computing concepts and technologies to advance scientific research and practical applications of science knowledge

computer program algorithms that can be run on a computer computer science (CS) study of the phenomena surrounding computers computing all phenomena related to computers

data science interdisciplinary field that applies computing toward managing data and extracting information from data

hardware physical, real-world materials that enable computation

human-computer interaction (HCI) subfield of computer science that emphasizes the social aspects of computation

image recognition problem of identifying objects in an image

information science interdisciplinary field studying information technologies and systems as they relate to people, organizations, and societies

memory means of addressing information in a computer by storing it in consistent locations network various technological devices that are connected and share information

neural network AI algorithm architecture that emphasizes connections between artificial "neurons" whose behavior and values change in response to stimulus or input

processor computer's "brain," that follows instructions from algorithms and processes data programming language language consisting of symbols and instructions that can be interpreted by a computer

social determination of technology belief that technologies are inherently neutral, and that it is the people who use a technology who ultimately make it "good" or "bad"

software algorithmic principles that determine how results are computed software engineeringsubfield of computer science that emphasizes how problems can be solved with computers as well as the practices and processes that can help people design more effective software solutions

spreadsheet data-centric programming environment where data is organized into cells in a table storage hardware and physical components of a computer that permanently house a computer's data technological fix idea that technologies can solve social problems, but now often used to critique blind faith in technological solutions to human problems

theoretical computer science mathematical processes behind software

Turing-complete fundamental model for computing results and every computer has the ability to run any algorithm

vacuum tube physical device that works like a light bulb used as memory in early digital computers

Summary

1.1 Computer Science

- Computer science is pervasive in our daily lives, business and industry, scientific research and development, and social change.
- Computer science (CS) is the study of computing, which includes all phenomena related to computers,

such as the Internet. With foundations in engineering and mathematics, computer science focuses on studying algorithms, which are instructions that enable computing. This includes computer hardware and software and the way these are used to process information. Three perspectives on computers include the hardware perspective, software perspective, and theoretical perspective. These perspectives each emphasize different aspects of computation, and they're often centered in undergraduate computer science because of the history of computer science, but there are other perspectives on computer science.

- People have used computer science to advance many more diverse goals beyond making war or making money. Computing was imagined as: a new medium for helping people learn everything; a new technology that could enable anti-racism; a means of enabling global development for peoples across the world. Yet these visions and promises are still taking hold in a world largely focused on the dominant history of computer science.

1.2 Computer Science across the Disciplines

- By contributing tools and resources to handle tasks and improve operations, computer science enables many other fields and areas of research or development.

- Data science is an interdisciplinary field that applies computing to managing data and extracting information from data. Many millions of people engage in data science work by using spreadsheets. Still, data science also often emphasizes larger-scale problems involving big data that are hard to manage using spreadsheets alone.

- Computational science refers to applying computing concepts and technologies to advance scientific research and practical applications of science knowledge. Computer science's emphasis on creating things can help other sciences by, for example, contributing new models or simulations that can enable the discovery of new kinds of scientific knowledge previously inaccessible to scientists.

- Information science is an interdisciplinary field studying information technologies and systems as they relate to people, organizations, and societies. As computing is now so central to information management and information exchange, information science has significant overlap with computer science. Still, it tends to emphasize the social value of information, whereas computer science has (historically) emphasized algorithms and computation over people or information.

- Today, computer science is an interdisciplinary field that contributes to all other fields. Effective computational solutions to research or business problems require combining domain-specific knowledge with computer science concepts from a combination of areas.

1.3 Computer Science and the Future of Society

- Computer science is shaping the future of society. There are three ways in which computer science can shape the future of society: developing foundational technologies, evaluating negative consequences of technologies, and designing technologies for social good.

- As one example of developing foundational technologies, computer science's rapid development of artificial intelligence technologies (and the current trend around neural networks) has enabled many new applications like image recognition. These developments often do not occur in isolation: the popular use of neural networks, for example, depended on new computer architectures and advancements in the Internet (computer networks). Technologies can encode social values: neural networks are designed to learn from big data, so they encode a preference for the contemporary social realities that produced the data.

- As one example of evaluating negative consequences, computer science considers the philosophical and practical limitations of neural networks. Research into adversarial attacks can enable computer scientists to develop more robust neural networks that are safer and more effective.

- As one example of designing technologies for social good, computer science contributes to the research, development, mass production, delivery, and logistics of modern medicine from beginning to end. Yet applications for social good are often embedded in broader social and political dynamics that computer science has difficulty addressing. Even though computer technologies can be designed for social good, they can cause harm when their design processes fail to center on human values and diverse users.