grokking

# Artificial Intelligence Algorithms

Rishal Hurbans

MANNING
Shelter Island

For online information and ordering of this and other Manning books, please visit www.manning.com. The publisher offers discounts on this book when ordered in quantity. For more information, please contact

Special Sales Department
Manning Publications Co.
20 Baldwin Road, PO Box 761
Shelter Island, NY 11964
Email: orders@manning.com



Manning Publications Co.
20 Baldwin Road
Shelter Island, NY 11964

Development editor: Elesha Hyde
Technical development editor: Frances Buontempo
Review editor: Ivan Martinović
Production editor: Deirdre Hiam
Copy editor: Keir Simpson
Proofreader: Jason Everett
Technical proofreader: Krzysztof Kamyczek
Typesetter: Jennifer Houle
Cover designer: Marija Tudor

ISBN: 9781617296185
Printed in the United States of America

# contents

# preface



This preface aims to describe the evolution of technology, our need to automate, and our responsibility to make ethical decisions while using artificial intelligence in building the future.

## Our obsession with technology and automation

Throughout history, we have had a hunger to solve problems while reducing manual labor and human effort. We have always strived for survival and conservation of our energy through the development of tools and automation of tasks. Some may argue that we are beautiful minds that seek innovation through creative problem-solving or creative works of literature, music, and art, but this book wasn't written to discuss philosophical questions about our being. This is an overview of artificial intelligence (AI) approaches that can be harnessed to address real-world problems practically. We solve hard problems to make living easier, safer, healthier, more fulfilling, and more enjoyable. All the advancements that you see in history and around the world today, including AI, address the needs of individuals, communities, and nations.

To shape our future, we must understand some key milestones in our past. In many revolutions, human innovation changed the way we live, and shaped the way we interact with the world and the way we think about it. We continue to do this as we iterate and improve the tools we use, which open future possibilities (figure 0.1).

This short high-level material on history and philosophy is provided purely to establish a baseline understanding of technology and AI, and to spur thought on responsible decision-making when embarking on your projects.

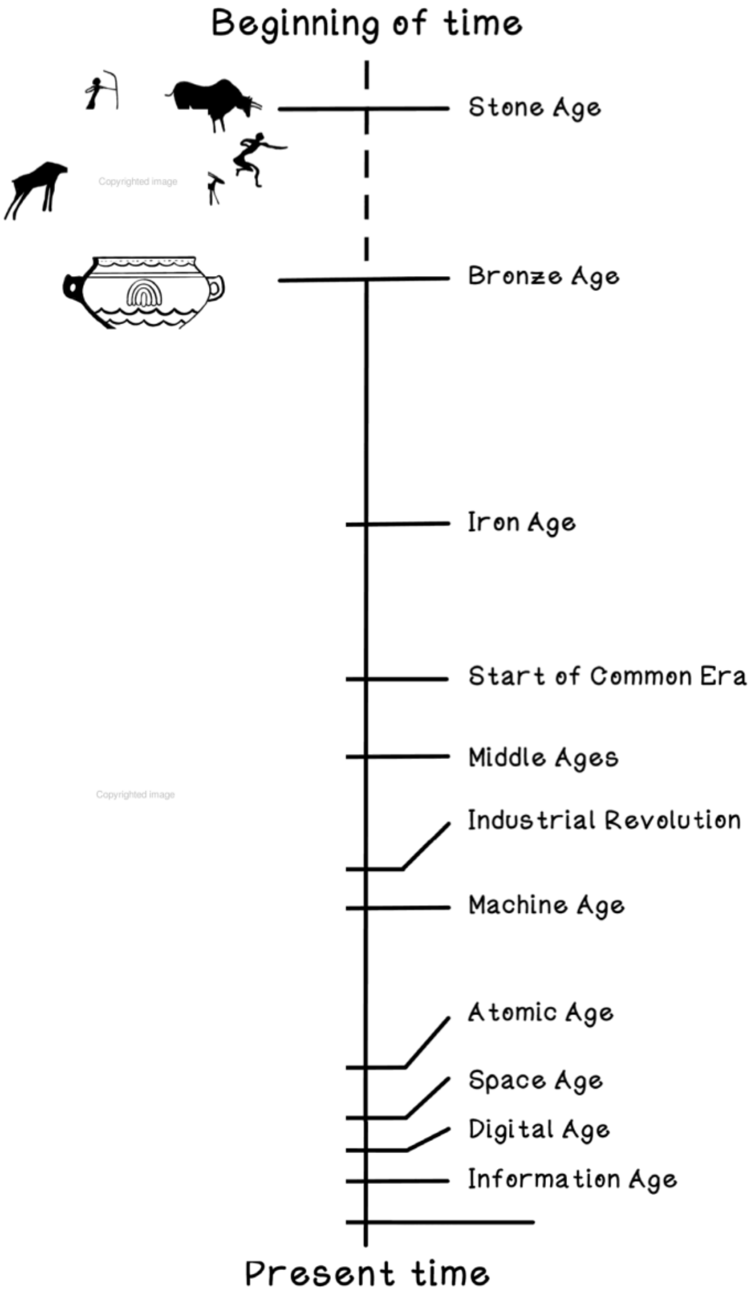



**Figure 0.1 A brief timeline of technological improvements in history**


Figure 0.4 Aim for ethical and legal applications of technology

## Intention and impact: Understanding your vision and goals

When you develop anything—such as a new physical product, service, or software—there's always a question about the intention behind it. Are you developing software that affects the world positively, or is your intention malevolent? Have you thought about the broader impact of what you're developing? Businesses always find ways to become more profitable and powerful, which is the whole point of growing a business. They use strategies to determine the best ways to beat the competition, gain more customers, and become even more influential. That said, businesses must ask themselves whether their intentions are pure, not only for the survival of the business, but also for the good of their customers and society in general. Many famous scientists, engineers, and technologists have expressed a need to govern the use of AI to prevent misuse. As individuals, we also have an ethical obligation to do what is right and establish a strong core set of values. When you're asked to do something that violates your principles, it is important to voice those principles.

## Unintended use: Protecting against malicious use

It is important to identify and protect against unintended use. Although this may seem obvious and easy to accomplish, it is difficult to understand how people will use whatever you are creating, and even more difficult to predict whether it aligns with your values and the values of the organization.

An example is the loudspeaker, which was invented by Peter Jensen in 1915. The loudspeaker was originally called Magnavox, which was initially used to play opera music to large crowds in San Francisco, which is quite a benevolent use of the technology. The Nazi regime in Germany had other ideas, however: they placed loudspeakers in public places in such a way that everyone was subjected to hearing Hitler's speeches and announcements. Because the monologues were unavoidable, people became more susceptible to Hitler's ideas, and after this point in time, the Nazi regime gained the majority of its support in Germany. This is not what Jensen envisioned his invention being used for, but there's not much he could have done about it.

Times have changed, and we have more control of the things we build, especially software. It is still difficult to imagine how the technology you build may be used, but it is almost guaranteed that someone will find a way to use it in a way that you did not intend, with positive or negative consequences. Given this fact, we, as professionals in the technology industry and the organizations we work with must think of ways to mitigate malevolent use as far as possible.

## Unintended bias: Building solutions for everyone

When building AI systems, we use our understanding of contexts and domains. We also use algorithms that find patterns in data and act on it. It can't be denied that there is bias all around us. A bias is a prejudice against a person or group of people, including, but not limited to their gender, race, and beliefs. Many of these biases arise from emergent behavior in social interactions, events in history, and cultural and political views around the world. These biases affect the data that we collect. Because AI algorithms work with this data, it is an inherent problem that the machine will "learn" these biases. From a technical perspective, we can engineer the system perfectly, but at the end of the day, humans interact with these systems, and it's our responsibility to minimize bias and prejudice as much as possible. The algorithms we use are only as good as the data provided to them. Understanding the data and the context in which it is being used is the first step in battling bias, and this understanding will help you build better solutions—because you will be well versed in the problem space. Providing balanced data with as little bias as possible should result in better solutions.

## The law, privacy, and consent: Knowing the importance of core values

The legal aspect of what we do is hugely important. The law governs what we can and cannot do in the interest of society as a whole. Due to the fact that many laws were written in a time when computers and the internet were not as important in our lives as they are today, we find many gray areas in how we develop technology and what we are allowed to do with that technology. That said, laws are slowly changing to adapt to the rapid innovation in technology.

We are compromising our privacy almost every hour of every day via our interactions on computers, mobile phones, and other devices, for example. We are transmitting a vast amount of information about ourselves, some more personal than others. How is that data being processed and stored? We should consider these facts when building solutions. People should have a choice about what data is captured, processed, and stored about them; how that data is used; and who can potentially access that data. In my experience, people generally accept solutions that use their data to improve the products they use and add more value to their lives. Most important, people are more accepting when they are given a choice and that choice is respected.

## Singularity: Exploring the unknown

The *singularity* is the idea that we create an AI that is so generally intelligent that it is capable of improving itself and expanding its intelligence to a stage where it becomes super intelligence. The concern is that something of this magnitude cannot be understood by humans which could change civilization as we know it for reasons we can't even comprehend. Some people are concerned that this intelligence may see humans as a threat; others propose that we may be to a super intelligence what ants are to us. We don't pay explicit attention to ants or concern ourselves with how they live, but if we are irritated by them, we deal with them in isolation.

Whether these assumptions are accurate representations of the future or not, we must be responsible and think about the decisions we make, as they ultimately affect a person, a group of people, or the world at large.

# acknowledgments



Writing this book has been one of the most challenging yet rewarding things I've done to date. I needed to find time where I had none, find the right headspace while juggling many contexts, and find motivation while being caught up in the reality of life. I couldn't have done it without a number of amazing people. I have learned and grown through this experience. Thank you, Bert Bates, for being a fantastic editor and mentor to me. I've learned so much about effective teaching and written communication from you. Our discussions and debates, and your empathy throughout the process has helped mold this book into what it is. Every project needs someone organized with a finger on the pulse making sure things are happening. For this, I'd like to thank Elesha Hyde, my development editor. Working with you has been an absolute pleasure. You always provide direction and interesting insights about my work. We always need people to bounce ideas off, and who better to annoy than your friends. I'd like to especially thank Hennie Brink for being a great sounding board and pillar of support always. Next, I'd like to thank Frances Buontempo and Krzysztof Kamyczek for providing constructive criticism and objective feedback from a writing and technical perspective. Your input has helped fill gaps and make the teaching more accessible. I would also like to thank Deirdre Hiam, my project manager; my review editor, Ivan Martinovic; my copyeditor Kier Simpson; and my proofreader, Jason Everett.

Finally, I'd like to thank all the reviewers who took the time to read my manuscript throughout development and provided invaluable feedback that has made the book better in some or other way: Andre Weiner, Arav Agarwal, Charles Soetan, Dan Sheikh, David Jacobs, Dhivya Sivasubramanian, Domingo Salazar, GandhiRajan, Helen Mary Barrameda, James Zhijun Liu, Joseph Friedman, Jousef Murad, Karan Nih, Kelvin D. Meeks, Ken Byrne, Krzysztof Kamyczek, Kyle Peterson, Linda Ristevski, Martin Lopez, Peter Brown, Philip Patterson, Rodolfo Allendes, Tejas Jain, and Weiran Deng.

- *Chapter 1—Intuition of artificial intelligence*, introduces the intuition and fundamental concepts that surround data, types of problems, categories of algorithms and paradigms, and use cases for artificial intelligence algorithms.
- *Chapter 2—Search fundamentals*, covers the core concepts of data structures and approaches for primitive search algorithms, and their uses.
- *Chapter 3—Intelligent search*, goes beyond primitive search algorithms and introduces search algorithms for finding solutions more optimally, and finding solutions in a competitive environment.
- *Chapter 4—Evolutionary algorithms*, dives into the workings of genetic algorithms where solutions to problems are iteratively generated and improved upon by mimicking evolution in nature.
- *Chapter 5—Advanced evolutionary approaches*, is a continuation of genetic algorithms but tackles advanced concepts involving how steps in the algorithm can be adjusted to solve different types of problems more optimally.
- *Chapter 6—Swarm intelligence: Ants*, digs into the intuition for swam intelligence and works through how the ant colony optimization algorithm uses a theory of how ants live and work to solve hard problems.
- *Chapter 7—Swarm intelligence: Particles*, continues with swarm algorithms while diving into what optimization problems are, and how they're solved using particle swarm optimization—as it seeks good solutions in large search spaces.
- *Chapter 8—Machine learning*, works through a machine learning workflow for data preparation, processing, modeling, and testing—to solve regression problems with linear regression, and classification problems with decision trees.
- *Chapter 9—Artificial neural networks*, uncovers the intuition, logical steps, and mathematical calculations in training and using an artificial neural network to find patterns in data and make predictions; while highlighting its place in a machine learning workflow.
- *Chapter 10—Reinforcement learning with Q-Learning*, covers the intuition of reinforcement learning from behavioral psychology, and works through the Q-Learning algorithm for agents to learn good and bad decisions to make in an environment.

The chapters should be read from start to end sequentially. Concepts and understandings are built up along the way as the chapters progress. It is useful to reference the Python code in the repository after reading each chapter to experiment and gain practical insight into how the respective algorithm can be implemented.

## About the Code

This book contains Pseudocode to focus on the intuition and logical thinking behind the algorithms, as well as to ensure that the code is accessible to anyone, regardless of programming language preferences. Pseudocode is an informal way to describe instructions in code. It is intended to be more readable and understandable; basically more human-friendly.

With that said, all algorithms described in the book have examples of working Python code available on Github (http://mng.bz/Vgr0). Setup instructions and comments are provided in the source code to guide you as you learn. One potential learning approach would be to read each chapter then reference the code after to cement your understanding of the respective algorithms.

The Python source code is intended to be a reference for how the algorithms could be implemented. These examples are optimized FOR LEARNING and NOT PRODUCTION use. The code was written to serve as a tool for teaching. Using established libraries and frameworks is recommended for projects that will make their way into production, as they are usually optimized for performance, well tested, and well supported.

## liveBook discussion forum

Purchase of *Grokking Artificial Intelligence Algorithms* includes free access to a private web forum run by Manning Publications where you can make comments about the book, ask technical questions, and receive help from the author and from other users. To access the forum, go to http://mng.bz/xWoe. You can also learn more about Manning's forums and the rules of conduct at https://livebook.manning.com/#!/discussion.

Manning's commitment to our readers is to provide a venue where a meaningful dialogue between individual readers and between readers and the author can take place. It is not a commitment to any specific amount of participation on the part of the author, whose contribution to the forum remains voluntary (and unpaid). We suggest you try asking the author some challenging questions lest his interest stray! The forum and the archives of previous discussions will be accessible from the publisher's website as long as the book is in print.

## Other online resources

Source code for Grokking Artificial Intelligence Algorithms:
  http://mng.bz/Vgr0
Author website:
  https://rhurbans.com

# about the author



Rishal has been obsessed with computers, technology, and crazy ideas since childhood. Throughout his career he has been involved in the leadership of teams and projects, hands-on software engineering, strategic planning, and the end-to-end design of solutions for various international businesses. He has also been responsible for actively growing a culture of pragmatism, learning and skills development within his company, community, and industry.

Rishal has a passion for business mechanics and strategy, growing people and teams, design thinking, artificial intelligence, and philosophy. Rishal has founded various digital products to help people and businesses be more productive and focus on what's important. He has also spoken at dozens of conferences around the globe to make complex concepts more accessible and help people elevate themselves.

## Understanding that data is core to AI algorithms

Data is the input to the wonderful algorithms that perform feats that almost appear to be magic. With the incorrect choice of data, poorly represented data, or missing data, algorithms perform poorly, so the outcome is only as good as the data provided. The world is filled with data, and that data exists in forms we can't even sense. Data can represent values that are measured numerically, such as the current temperature in the Arctic, the number of fish in a pond, or your current age in days. All these examples involve capturing accurate numeric values based on facts. It's difficult to misinterpret this data. The temperature at a specific location at a specific point in time is absolutely true and is not subject to any bias. This type of data is known as *quantitative data*.

Data can also represent values of observations, such as the smell of a flower or one's level of agreement with a politician's policies. This type of data is known as *qualitative data* and is sometimes difficult to interpret because it's not an absolute truth, but a perception of someone's truth. Figure 1.1 illustrates some examples of the quantitative and qualitative data around us.

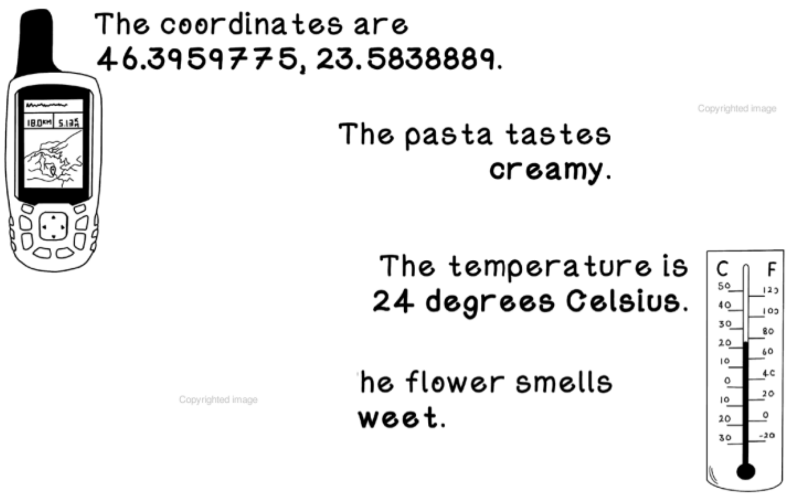



**Figure 1.1 Examples of data around us**

Data is raw facts about things, so recordings of it usually have no bias. In the real world, however, data is collected, recorded, and related by people based on a specific context with a specific understanding of how the data may be used. The act of constructing meaningful insights to answer questions based on data is creating *information*. Furthermore, the act of utilizing information with experiences and consciously applying it creates *knowledge*. This is partly what we try to simulate with AI algorithms.

Figure 1.2 shows how quantitative and qualitative data can be interpreted. Standardized instruments such as clocks, calculators, and scales are usually used to measure quantitative data, whereas our senses of smell, sound, taste, touch, and sight, as well as our opinionated thoughts, are usually used to create qualitative data.

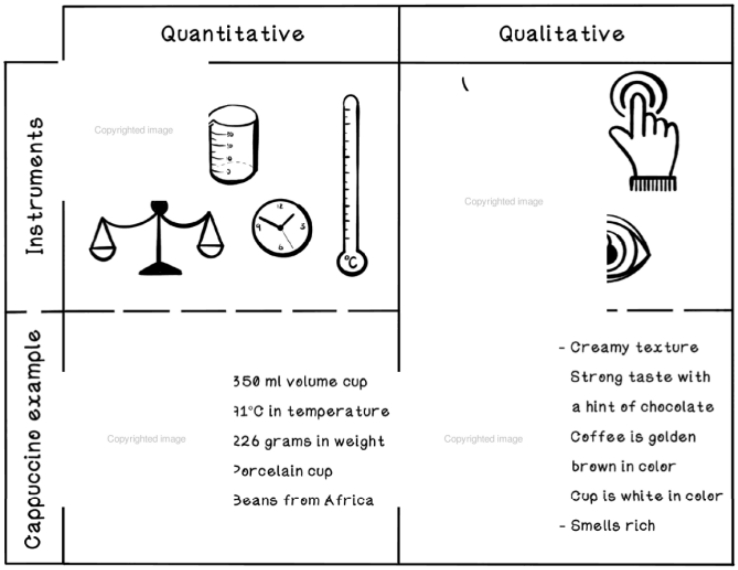



**Figure 1.2 Qualitative data versus quantitative data**

Data, information, and knowledge can be interpreted differently by different people, based on their level of understanding of that domain and their outlook on the world, and this fact has consequences for the quality of solutions—making the scientific aspect of creating technology hugely important. By following repeatable scientific processes to capture data, conduct experiments, and accurately report findings, we can ensure more accurate results and better solutions to problems when processing data with algorithms.

## Viewing algorithms as instructions in recipes

We now have a loose definition of AI and an understanding of the importance of data. Because we will be exploring several AI algorithms throughout this book, it is useful to understand exactly what an algorithm is. An *algorithm* is a set of instructions and rules provided as a specification to accomplish a specific goal. Algorithms typically accept inputs, and after several finite steps in which the algorithm progresses through varying states, an output is produced.

Even something as simple as reading a book can be represented as an algorithm. Here's an example of the steps involved in reading this book:

1. Find the book *Grokking Artificial Intelligence Algorithms.*
2. Open the book.

3. While unread pages remain,
   a. Read page.
   b. Turn to next page.
   c. Think about what you have learned.
4. Think about how you can apply your learnings in the real world.

An algorithm can be viewed as a recipe, as seen in figure 1.3. Given some ingredients and tools as inputs, and instructions for creating a specific dish, a meal is the output.

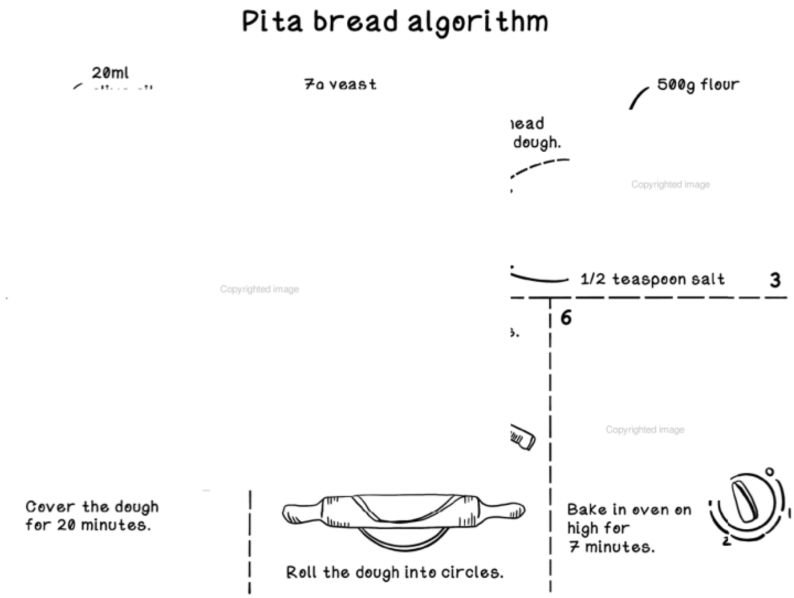



**Figure 1.3 An example showing that an algorithm is like a recipe**

Algorithms are used for many different solutions. For example, we can enable live video chat across the world through compression algorithms, and we can navigate cities through map applications that use real-time routing algorithms. Even a simple "Hello World" program has many algorithms at play to translate the human-readable programming language into machine code and execute the instructions on the hardware. You can find algorithms everywhere if you look closely enough.

To illustrate something more closely related to the algorithms in this book, figure 1.4 shows a number-guessing-game algorithm represented as a flow chart. The computer generates a random number in a given range, and the player attempts to guess that number. Notice that the algorithm has discrete steps that perform an action or determine a decision before moving to the next operation.

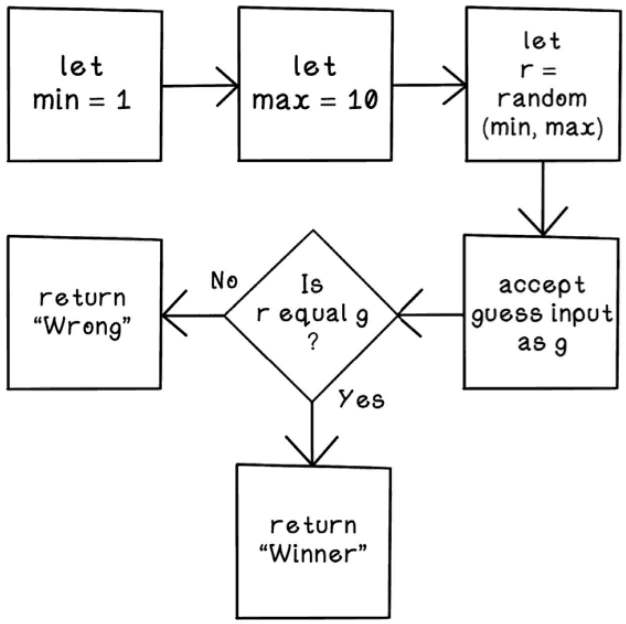



**Figure 1.4 A number-guessing-game algorithm flow chart**

Given our understanding of technology, data, intelligence, and algorithms: AI algorithms are sets of instructions that use data to create systems that exhibit intelligent behavior and solve hard problems.

## A brief history of artificial intelligence

A brief look back at the strides made in AI is useful for understanding that old techniques and new ideas can be harnessed to solve problems in innovative ways. AI is not a new idea. History is filled with myths of mechanical men and autonomous "thinking" machines. Looking back, we find that we're standing on the shoulders of giants. Perhaps we ourselves can contribute to the pool of knowledge in a small way.

Looking at past developments highlights the importance of understanding the fundamentals of AI; algorithms from decades ago are critical in many modern AI implementations. This book starts with fundamental algorithms that help build the intuition of problem-solving and gradually moves to more interesting and modern approaches.

Figure 1.5 isn't an exhaustive list of achievements in AI—it is simply a small set of examples. History is filled with many more breakthroughs!

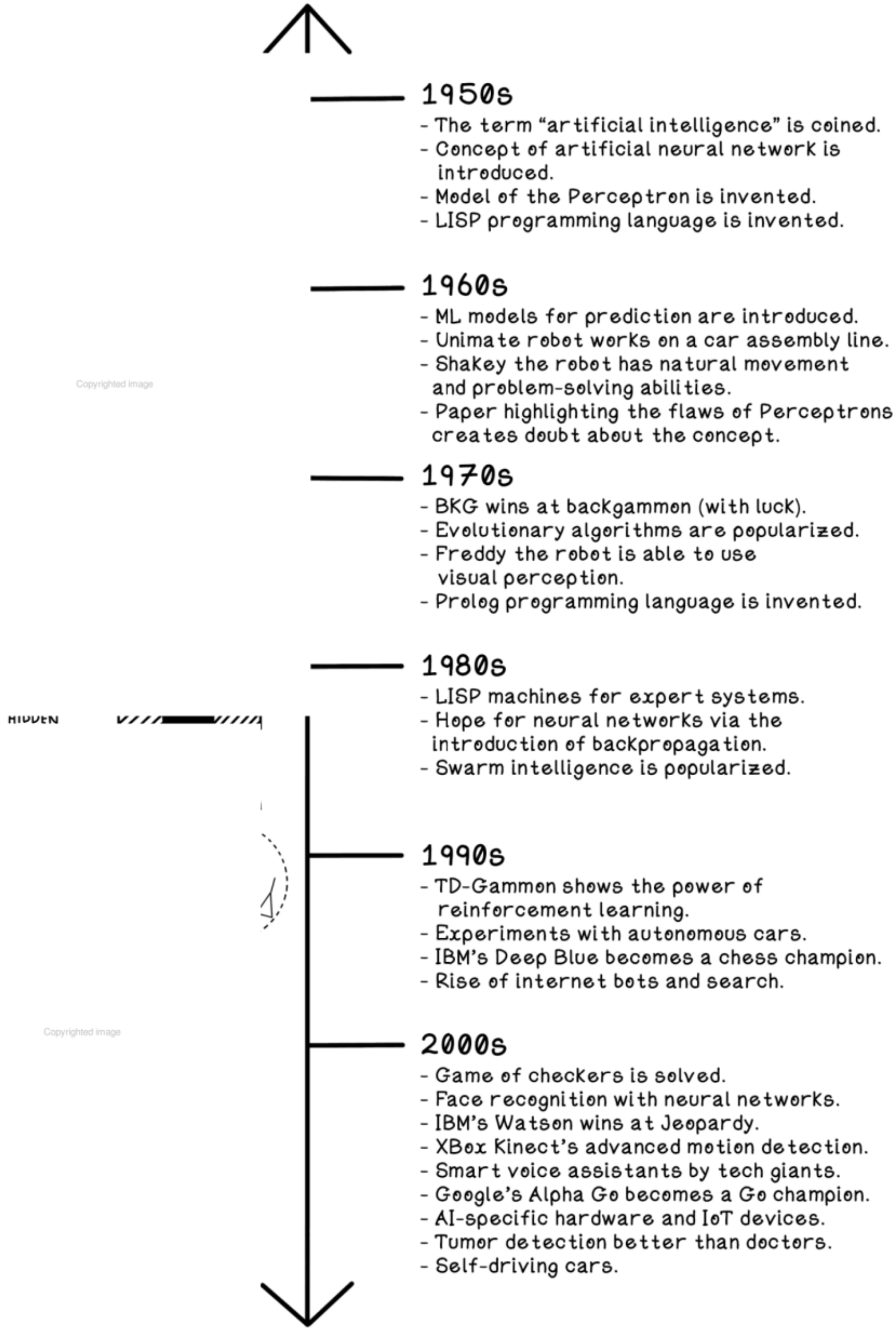



**Figure 1.5 The evolution of AI**

# Intuition of artificial intelligence concepts

AI is a hot topic, as are machine learning and deep learning. Trying to make sense of these different but similar concepts can be a daunting experience. Additionally, within the domain of AI, distinctions exist among different levels of intelligence.

In this section, we demystify some of these concepts. The section is also a road map of the topics covered throughout this book.

Let's dive into the different levels of AI, introduced with figure 1.6.



**Figure 1.6 Levels of AI**

## Narrow intelligence: Specific-purpose solutions

*Narrow intelligence* systems solve problems in a specific context or domain. These systems usually cannot solve a problem in one context and apply that same understanding in another. A system developed to understand customer interactions and spending behavior, for example, would not be capable of identifying cats in an image. Usually, for something to be effective in solving a problem, it needs to be quite specialized in the domain of the problem, which makes it difficult to adapt to other problems.

Different narrow intelligence systems can be combined in sensible ways to create something greater that seems to be more general in its intelligence. An example is a voice assistant. This system can understand natural language, which alone is a narrow problem, but through integration with other narrow intelligence systems, such as web searches and music recommenders, it can exhibit qualities of general intelligence.

## General intelligence: Humanlike solutions

*General intelligence* is humanlike intelligence. As humans, we are able to learn from various experiences and interactions in the world and apply that understanding from one problem to another. If you felt pain when touching something hot as a child, for example, you can extrapolate and know that other things that are hot may have a chance of hurting you. General intelligence in humans, however, is more than just reasoning something like "Hot things may be harmful." General intelligence encompasses memory, spatial reasoning through visual inputs, use of knowledge, and more. Achieving general intelligence in a machine seems to be an unlikely feat in the short term, but advancements in quantum computing, data processing, and AI algorithms could make it a reality in the future.

## Super intelligence: The great unknown

Some ideas of *super intelligence* appear in science-fiction movies set in postapocalyptic worlds, in which all machines are connected, are able to reason about things beyond our understanding, and dominate humans. There are many philosophical differences about whether humans could create something more intelligent than ourselves and, if we could, whether we'd even know. Super intelligence is the great unknown, and for a long time, any definitions will be speculation.

## Old AI and new AI

Sometimes, the notions of old AI and new AI are used. *Old AI* is often understood as being systems in which people encoded the rules that cause an algorithm to exhibit intelligent behavior—via in-depth knowledge of the problem or by trial and error. An example of old AI is a person manually creating a decision tree and the rules and options in the entire decision tree. *New AI* aims to create algorithms and models that learn from data and create their own rules that perform as accurately as, or better than, human-created rules. The difference is that the latter may find important patterns in the data that a person may never find or that would take a person much longer to find. Search algorithms are often seen as old AI, but a robust understanding of them is useful in learning more complex approaches. This book aims to introduce the most popular AI algorithms and gradually build on each concept. Figure 1.7 illustrates the relationship between some of the different concepts within artificial intelligence.

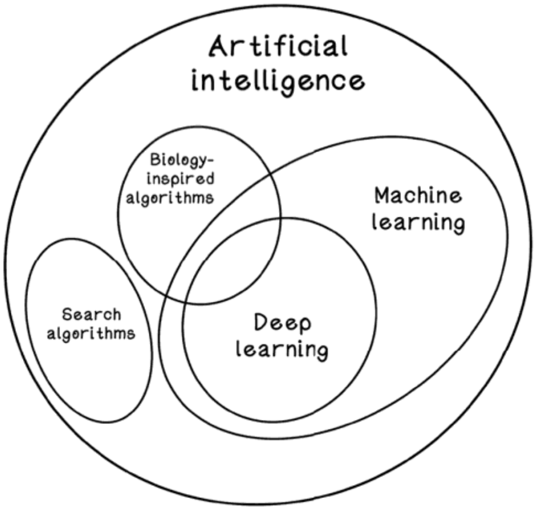



**Figure 1.7 Categorization of concepts within AI**

## Search algorithms

*Search algorithms* are useful for solving problems in which several actions are required to achieve a goal, such as finding a path through a maze or determining the best move to make in a game. Search algorithms evaluate future states and attempt to find the optimal path to the most valuable goal. Typically, we have too many possible solutions to brute-force each one. Even small search spaces could result in thousands of hours of computing to find the best solution. Search algorithms provide smart ways to evaluate the search space. Search algorithms are used in online search engines, map routing applications, and even game-playing agents.

## Biology-inspired algorithms

When we look at the world around us, we notice incredible things in various creatures, plants, and other living organisms. Examples include the cooperation of ants in gathering food, the flocking of birds when migrating, estimating how brains work, and the evolution of different organisms to produce stronger offspring. By observing and learning from various phenomena, we've gained knowledge of how these organic systems operate and of how simple rules can result in emergent intelligent behavior. Some of these phenomena have inspired algorithms that are useful in AI, such as evolutionary algorithms and swarm intelligence algorithms.

*Evolutionary algorithms* are inspired by the theory of evolution defined by Charles Darwin. The concept is that a population reproduces to create new individuals and that through this process, the mixture of genes and mutation produces individuals that perform better than their ancestors. *Swarm intelligence* is a group of seemingly "dumb" individuals exhibiting intelligent behavior. Ant-colony optimization and particle-swarm optimization are two popular algorithms that we will be exploring in this book.

## Machine learning algorithms

Machine learning takes a statistical approach to training models to learn from data. The umbrella of machine learning has a variety of algorithms that can be harnessed to improve understanding of relationships in data, to make decisions, and to make predictions based on that data.

There are three main approaches in machine learning:

- *Supervised learning* means training models with algorithms when the training data has known outcomes for a question being asked, such as determining the type of fruit if we have a set of data that includes the weight, color, texture, and fruit label for each example.
- *Unsupervised learning* uncovers hidden relationships and structures within the data that guide us in asking the dataset relevant questions. It may find patterns in properties of similar fruits and group them accordingly, which can inform the exact questions we want to ask the data. These core concepts and algorithms helps us create a foundation for exploring advanced algorithms in the future.
- *Reinforcement learning* is inspired by behavioral psychology. In short, it describes rewarding an individual if a useful action was performed and penalizing that individual if an unfavorable action was performed. To explore a human example, when a child achieves good results on their report card, they are usually rewarded, but poor performance sometimes results in punishment, reinforcing the behavior of achieving good results. Reinforcement learning is useful for exploring how computer programs or robots interact with dynamic environments. An example is a robot that is tasked to open doors; it is penalized when it doesn't open a door and rewarded when it does. Over time, after many attempts, the robot "learns" the sequence of actions required to open a door.

## Deep learning algorithms

*Deep learning*, which stems from machine learning, is a broader family of approaches and algorithms that are used to achieve narrow intelligence and strive toward general intelligence. Deep learning usually implies that the approach is attempting to solve a problem in a more general way like spatial reasoning, or it is being applied to problems that require more generalization such as computer vision and speech recognition. General problems are things humans are good at solving. For example, we can match visual patterns in almost any context. Deep learning also concerns itself with supervised learning, unsupervised learning, and reinforcement learning. Deep learning approaches usually employ many layers of artificial neural networks. By leveraging different layers of intelligent components, each layer solves specialized problems; together, the layers solve complex problems toward a greater goal. Identifying any object in an image, for example, is a general problem, but it can be broken into understanding color, recognizing shapes of objects, and identifying relationships among objects to achieve a goal.

# Uses for artificial intelligence algorithms

The uses for AI techniques are potentially endless. Where there are data and problems to solve, there are potential applications of AI. Given the ever-changing environment, the evolution of interactions among humans, and the changes in what people and industries demand, AI can be applied in innovative ways to solve real-world problems. This section describes the application of AI in various industries.

## Agriculture: Optimal plant growth

One of the most important industries that sustain human life is agriculture. We need to be able to grow quality crops for mass consumption economically. Many farmers grow crops on a commercial scale to enable us to purchase fruit and vegetables at stores conveniently. Crops grow differently based on the type of crop, the nutrients in the soil, the water content of the soil, the bacteria in the water, and the weather conditions in the area, among other things. The goal is to grow as much high-quality produce as possible within a season, because specific crops generally grow well only during specific seasons.

Farmers and other agriculture organizations have captured data about their farms and crops over the years. Using that data, we can leverage machines to find patterns and relationships among the variables in the crop-growing process and identify the factors that contribute most to successful growth. Furthermore, with modern digital sensors, we can record weather conditions, soil attributes, water conditions, and crop growth in real time. This data, combined with intelligent algorithms, can enable real-time recommendations and adjustments for optimal growth (figure 1.8).

## Logistics: Routing and optimization

The logistics industry is a huge market of different types of vehicles delivering different types of goods to different locations, with different demands and deadlines. Imagine the complexity in a large e-commerce site's delivery planning. Whether the deliverables are consumer goods, construction equipment, parts for machinery, or fuel, the system aims to be as optimal as possible to ensure that demand is met and costs are minimized.

You may have heard of the traveling-salesperson problem: a salesperson needs to visit several locations to complete their job, and the aim is to find the shortest distance to accomplish this task. Logistics problems are similar but usually immensely more complex due to the changing environment of the real world. Through AI, we can find optimal routes between locations in terms of time and distance. Furthermore, we can find the best routes based on traffic patterns, construction blockages, and even road types based on the vehicle being used. Additionally, we can compute the best way to pack each vehicle and what to pack in each vehicle in such a way that each delivery is optimized.

## Telecoms: Optimizing networks

The telecommunications industry has played a huge role in connecting the world. These companies lay expensive infrastructure of cables, towers, and satellites to create a network that many consumers and organizations can use to communicate via the internet or private networks. Operating this equipment is expensive, so optimization of a network allows for more connections, which allows more people to access high-speed connections. AI can be used to monitor behavior on a network and optimize routing. Additionally, these networks record requests and responses; this data can be used to optimize the networks based on known load from certain individuals, areas, and specific local networks. The network data can also be instrumental for understanding where people are and who they are, which is useful for city planning.

## Games: Creating AI agents

Since home and personal computers first became widely available, games have been a selling point for computer systems. Games were popular very early in the history of personal computers. If we think back, we may remember arcade machines, television consoles, and personal computers with gaming capabilities. The games of chess, backgammon, and others have been dominated by AI machines. If the complexity of a game is low enough, a computer can potentially find all possibilities and make a decision based on that knowledge faster than a human can. Recently, a computer was able to defeat human champions in the strategic game, Go. Go has simple rules for territory control but has huge complexity in terms of the decisions that need to be made for a winning scenario. A computer can't generate all possibilities for beating the best human players because the search space is so large; instead, it calls for a more-general algorithm that can "think" abstractly, strategize, and plan moves toward a goal. That algorithm has already been invented and has succeeded in defeating world champions. It has also been adapted to other applications, such as playing Atari games and modern multiplayer games. This system is called Alpha Go.

Several research organizations have developed AI systems that are capable of playing highly complex games better than human players and teams. The goal of this work is to create general approaches that can adapt to different contexts. At face value, these game-playing AI algorithms may seem unimportant, but the consequence of developing these systems is that the approach can be applied effectively in other important problem spaces. Figure 1.10 illustrates how a reinforcement learning algorithm can learn to play a classic video game like Mario.



**Figure 1.10 Using neural networks to learn how to play games**

### Art: Creating masterpieces

Unique, interesting artists have created beautiful paintings. Each artist has their own way of expressing the world around them. We also have amazing music compositions that are appreciated by the masses. In both cases, the quality of the art cannot be measured quantitatively; rather, it is measured qualitatively (by how much people enjoy the piece). The factors involved are difficult to understand and capture; the concept is driven by emotion.

Many research projects aim to build AI that generates art. The concept involves generalization. An algorithm would need to have a broad and general understanding of the subject to create something that fits those parameters. A Van Gogh AI, for example, would need to understand all of Van Gogh's work and extract the style and "feel" so that it can apply that data to other images. The same thinking can be applied to extracting hidden patterns in areas such as health care, cybersecurity, and finance.

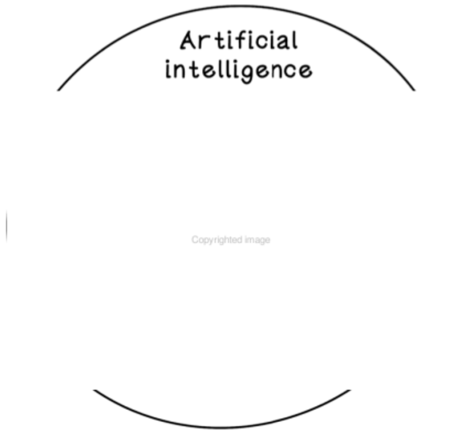
Now that we have abstract intuition about what AI is, the categorization of themes within it, the problems it aims to solve, and some use cases, we will be diving into one of the oldest and simplest forms of mimicking intelligence: search algorithms. Search algorithms provide a good grounding in some concepts that are employed by other, more sophisticated AI algorithms explored throughout this book.

## SUMMARY OF INTUITION OF ARTIFICIAL INTELLIGENCE

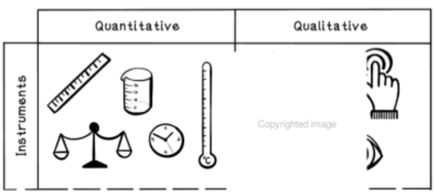
AI is difficult to define. There is no clear consensus.

Look at implementations as being AI-like things that exhibit intelligence.

Many disciplines are encompassed in AI.




AI implementations almost always have room for error. Be cautious about the consequences of this.




Quality and preparation of data is important.

AI has many uses and applications. Apply your mind!


Be responsible when developing technology.

# Search fundamentals 2



## This chapter covers

- The intuition of planning and searching
- Identifying problems suited to be solved using search algorithms
- Representing problem spaces in a way suitable to be processed by search algorithms
- Understanding and designing fundamental search algorithms to solve problems

## What are planning and searching?

When we think about what makes us intelligent, the ability to plan before carrying out actions is a prominent attribute. Before embarking on a trip to a different country, before starting a new project, before writing functions in code, planning happens. *Planning* happens at different levels of detail in different contexts to strive for the best possible outcome when carrying out the tasks involved in accomplishing goals (figure 2.1).

# Cost of computation: The reason for smart algorithms

In programming, functions consist of operations, and due to the way that traditional computers work, different functions use different amounts of processing time. The more computation required, the more expensive the function is. *Big O notation* is used to describe the complexity of a function or algorithm. Big O notation models the number of operations required as the input size increases. Here are some examples and associated complexities:

- *A single operation that prints* `Hello World`—This operation is a single operation, so the cost of computation is O(1).
- *A function that iterates over a list and prints each item in the list*—The number of operations is dependent on the number of items in the list. The cost is O(n).
- *A function that compares every item in a list with every item in another list*—This operation costs $O(n^2)$.

Figure 2.3 depicts different costs of algorithms. Algorithms that require operations to explore as the size of the input increases are the worst-performing; algorithms that require a more constant number of operations as the number of inputs increases are better.

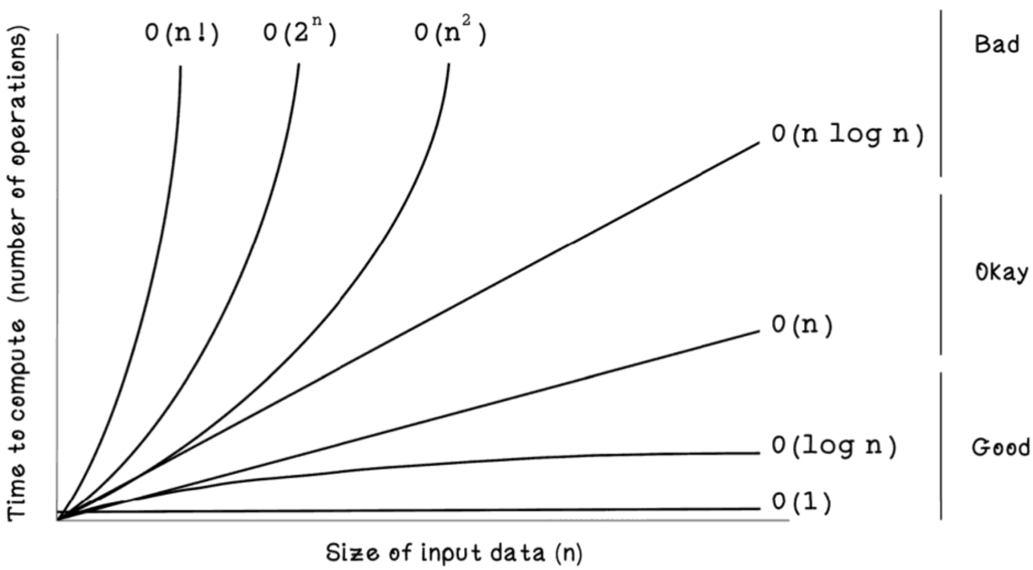



Figure 2.3 Big O complexity

Understanding that different algorithms have different computation costs is important because addressing this is the entire purpose of intelligent algorithms that solve problems well and quickly. Theoretically, we can solve almost any problem by brute-forcing every possible option until we find the best one, but in reality, the computation could take hours or even years, which makes it infeasible for real-world scenarios.

## Problems applicable to searching algorithms

Almost any problem that requires a series of decisions to be made can be solved with search algorithms. Depending on the problem and the size of the search space, different algorithms may be employed to help solve it. Depending on the search algorithm selected and the configuration used, the optimal solution or a best available solution may be found. In other words, a good solution will be found, but it might not necessarily be the best solution. When we speak about a "good solution" or "optimal solution," we are referring to the performance of the solution in addressing the problem at hand.

One scenario in which search algorithms are useful is being stuck in a maze and attempting to find the shortest path to a goal. Suppose that we're in a square maze consisting of an area of 10 blocks by 10 blocks (figure 2.4). There exists a goal that we want to reach and barriers that we cannot step into. The objective is to find a path to the goal while avoiding barriers with as few steps as possible by moving north, south, east, or west. In this example, the player cannot move diagonally.

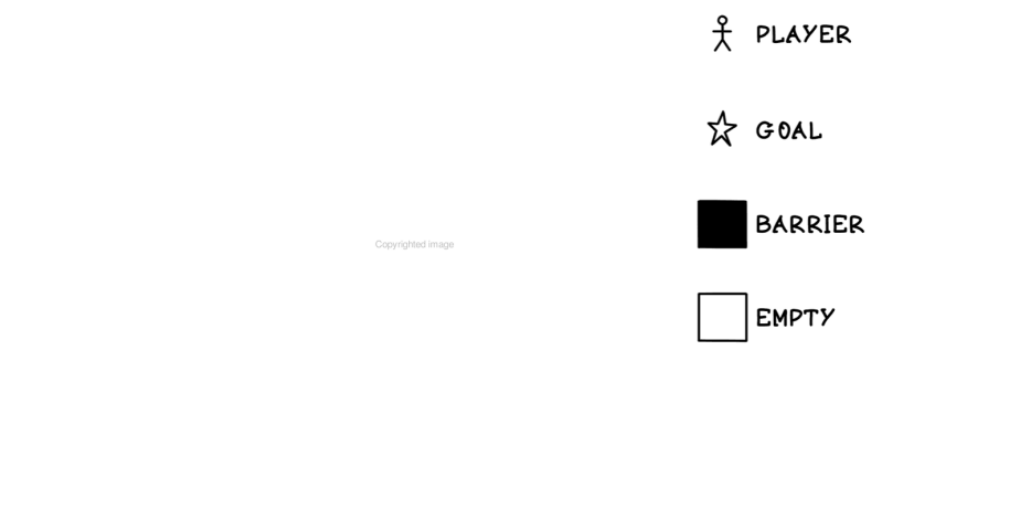



Figure 2.4 An example of the maze problem

How can we find the shortest path to the goal while avoiding barriers? By evaluating the problem as a human, we can try each possibility and count the moves. Using trial and error, we can find the paths that are the shortest, given that this maze is relatively small.

Using the example maze, figure 2.5 depicts some possible paths to reach the goal, although note that we don't reach the goal in option 1.

**Figure 2.5 Examples of possible paths to the maze problem**

By looking at the maze and counting blocks in different directions, we can find several solutions to the problem. Five attempts have been made to find four successful solutions out of an unknown number of solutions. It will take exhaustive effort to attempt to compute all possible solutions by hand:

- Attempt 1 is not a valid solution. It took 4 actions, and the goal was not found.
- Attempt 2 is a valid solution, taking 17 actions to find the goal.
- Attempt 3 is a valid solution, taking 23 actions to find the goal.
- Attempt 4 is a valid solution, taking 17 actions to find the goal.
- Attempt 5 is the best valid solution, taking 15 actions to find the goal. Although this attempt is the best one, it was found by chance.

If the maze were a lot larger, like the one in figure 2.6, it would take an immense amount of time to compute the best possible path manually. Search algorithms can help.



**Figure 2.6 A large example of the maze problem**

Our power as humans is to perceive a problem visually, understand it, and find solutions given the parameters. As humans, we understand and interpret data and information in an abstract way. A computer cannot yet understand generalized information in the natural form that we do. The problem space needs to be represented in a form that is applicable to computation and can be processed with search algorithms.

## Representing state: Creating a framework to represent problem spaces and solutions

When representing data and information in a way that a computer can understand, we need to encode it logically so that it can be understood objectively. Although the data will be encoded subjectively by the person who performs the task, there should be a concise, consistent way to represent it.

Let's clarify the difference between data and information. *Data* is raw facts about something, and *information* is an interpretation of those facts that provides insight about the data in the specific domain. Information requires context and processing of data to provide meaning. As an example, each individual distance traveled in the maze example is data, and the sum of the total distance traveled is information. Depending on the perspective, level of detail, and desired outcome, classifying something as data or information can be subjective to the context and person or team (figure 2.7).

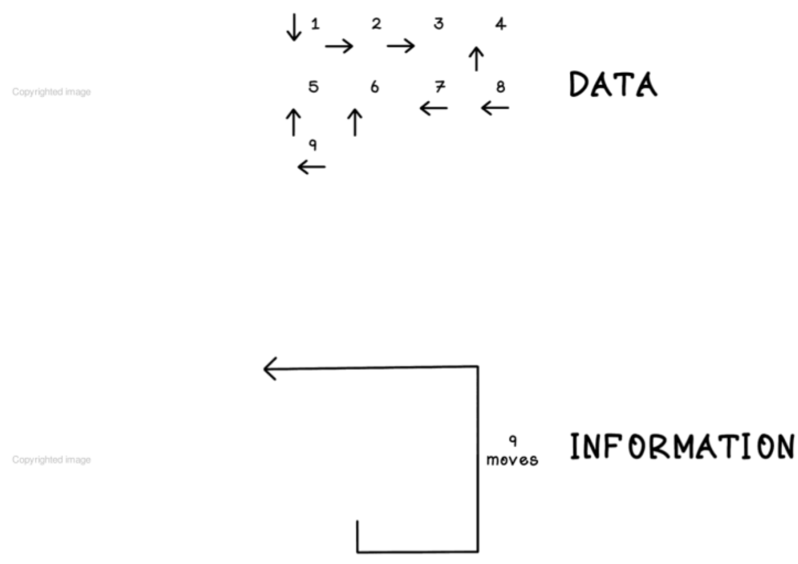



**Figure 2.7 Data versus information**

## Representing a graph as a concrete data structure

A graph can be represented in several ways for efficient processing by algorithms. At its core, a graph can be represented by an array of arrays that indicates relationships among nodes, as shown in figure 2.11. It is sometimes useful to have another array that simply lists all nodes in the graph so that the distinct nodes do not need to be inferred from the relationships.

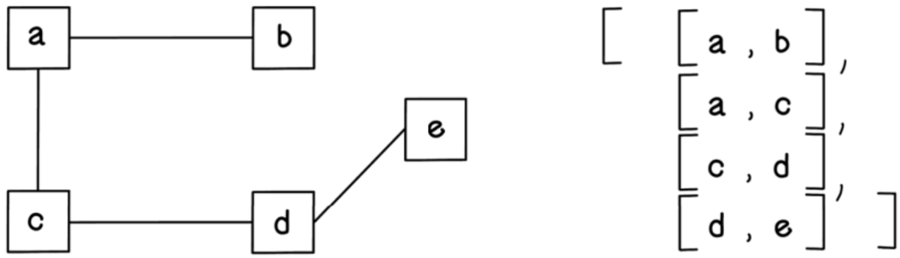



**Figure 2.11 Representing a graph as an array of arrays**

Other representations of graphs include an incidence matrix, an adjacency matrix, and an adjacency list. By looking at the names of these representations, you see that the adjacency of nodes in a graph is important. An *adjacent node* is a node that is connected directly to another node.

**EXERCISE: REPRESENT A GRAPH AS A MATRIX**

How would you represent the following graph using edge arrays?

**SOLUTION: REPRESENT A GRAPH AS A MATRIX**



```
[ [a, c],
  [a, f],
  [b, g],
  [b, f],
  [c, d],
  [d, g],
  [d, e],
  [e, h]  ]
```

Array of edges

| | a | b | c | d | e | f | g | h |
|---|---|---|---|---|---|---|---|---|
| a | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| b | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| c | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| d | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| e | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| f | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| g | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| h | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

Adjacency matrix

## Trees: The concrete structures used to represent search solutions

A *tree* is a popular data structure that simulates a hierarchy of values or objects. A *hierarchy* is an arrangement of things in which a single object is related to several other objects below it. A tree is a *connected acyclic graph*—every node has an edge to another node, and no cycles exist.

In a tree, the value or object represented at a specific point is called a *node*. Trees typically have a single root node with zero or more child nodes that could contain subtrees. Let's take a deep breath and jump into some terminology. When a node has connected nodes, the root node is called the *parent*. You can apply this thinking recursively. A child node may have its own child nodes, which may also contain subtrees. Each child node has a single parent node. A node without any children is a leaf node.

Trees also have a total height. The level of specific nodes is called a *depth*.

The terminology used to relate family members is heavily used in working with trees. Keep this analogy in mind, as it will help you connect the concepts in the tree data structure. Note that in figure 2.12, the height and depth are indexed from 0 from the root node.

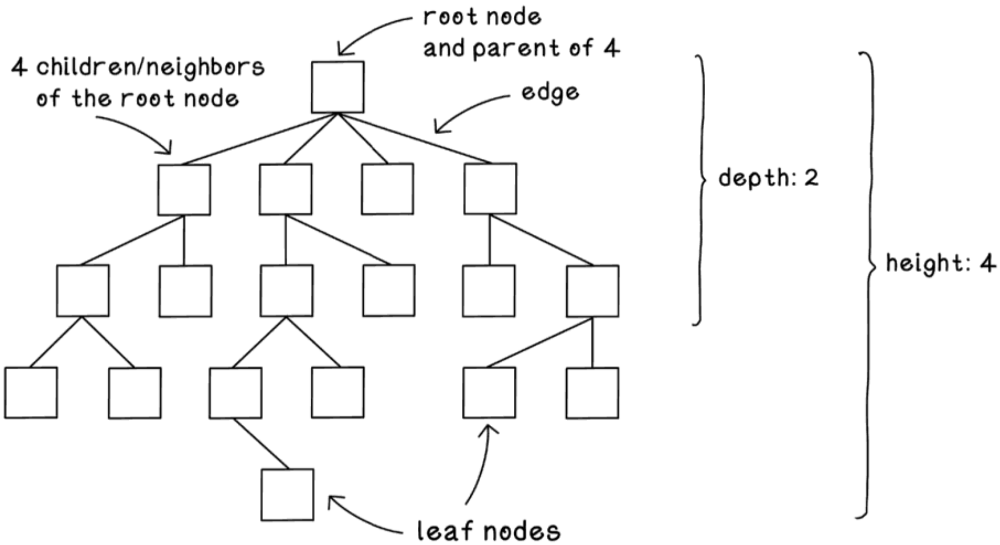



**Figure 2.12 The main attributes of a tree**

The topmost node in a tree is called the *root node*. A node directly connected to one or more other nodes is called a *parent node*. The nodes connected to a parent node are called *child nodes* **or** *neighbors*. Nodes connected to the same parent node are called *siblings*. A connection between two nodes is called an *edge*.

A *path* is a sequence of nodes and edges connecting nodes that are not directly connected. A node connected to another node by following a path away from the root node is called a *descendent*, and a node connected to another node by following a path toward the root node is called an *ancestor*. A node with no children is called a *leaf node*. The term *degree* is used to describe the number of children a node has; therefore, a leaf node has degree zero.

Figure 2.13 represents a path from the start point to the goal for the maze problem. This path contains nine nodes that represent different moves being made in the maze.

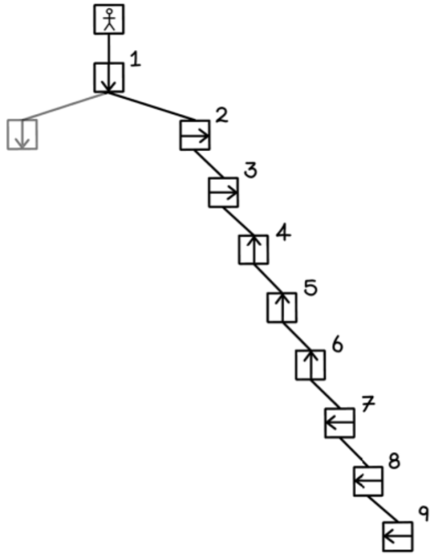



**Figure 2.13 A solution to the maze problem represented as a tree**

Trees are the fundamental data structure for search algorithms, which we will be diving into next. Sorting algorithms are also useful in solving certain problems and computing solutions more efficiently. If you're interested in learning more about sorting algorithms, take a look at *Grokking Algorithms* (Manning Publications).

# Uninformed search: Looking blindly for solutions

*Uninformed search* is also known as *unguided search*, *blind search*, or *brute-force search*. Uninformed search algorithms have no additional information about the domain of the problem apart from the representation of the problem, which is usually a tree.

Think about exploring things you want to learn. Some people might look at a wide breadth of different topics and learn the basics of each, whereas other people might choose one narrow topic and explore its subtopics in depth. This is what breadth-first search (BFS) and depth-first search (DFS) involve, respectively. *Depth-first search* explores a specific path from the start until it finds a goal at the utmost depth. *Breadth-first search* explores all options at a specific depth before moving to options deeper in the tree.

Consider the maze scenario (figure 2.14). In attempting to find an optimal path to the goal, assume the following simple constraint to prevent getting stuck in an endless loop