IN THIS CHAPTER

» Understanding deep learning

» Working with deep learning

» Developing deep learning applications

» Considering deep learning limitations

Chapter 1

Introducing Deep Learning

You have probably heard a lot about deep learning. The term appears all over the place and seems to apply to everything. In reality, deep learning is a subset of machine learning, which in turn is a subset of artificial intelligence (AI). The first goal of this chapter is to help you understand what deep learning is really all about and how it applies to the world today. You may be surprised to learn that deep learning isn't the only game in town; other methods of analyzing data exist. In fact, deep learning meets a specific set of needs when it comes to data analysis, so you might be using other methods and not even know it.

Deep learning is just a subset of AI, but it's an important subset. You see deep learning techniques used for a number of tasks, but not every task. In fact, some people associate deep learning with tasks that it can't perform. The next step in discovering deep learning is to understand what it can and can't do for you.

As part of working with deep learning in this book, you write applications that rely on deep learning to process data and then produce a desired output. Of course, you need to know a little about the programming environment before you can do much. Even though Chapter 3 discusses how to install and configure Python, the language used to demonstrate deep learning in this book, you first need to know a little more about the options available to you.

The chapter closes with a discussion of why deep learning shouldn't be the only data processing technique in your toolkit. Yes, deep learning can perform amazing tasks when used appropriately, but it can also cause serious problems when applied to problems that it doesn't support well. Sometimes you need to look to other technologies to perform a given task, or figure out which technologies to use with deep learning to provide a more efficient and elegant solution to specific problems.

Defining What Deep Learning Means

An understanding of deep learning begins with a precise definition of terms. Otherwise, you have a hard time separating the media hype from the realities of what deep learning can actually provide. Deep learning is part of both AI and machine learning, as shown in Figure 1-1. To understand deep learning, you must begin at the outside — that is, you start with AI, and then work your way through machine learning, and then finally define deep learning. The following sections help you through this process.

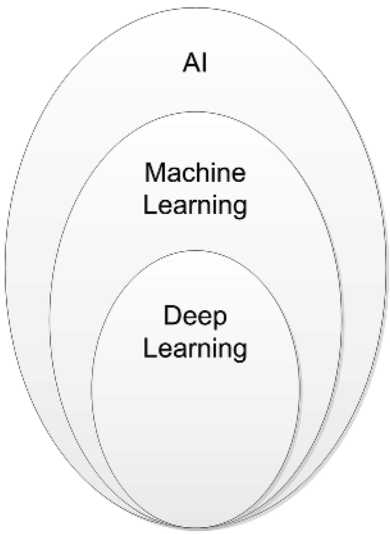

FIGURE 1-1: Deep learning is a subset of machine learning which is a subset of AI.

Starting from Artificial Intelligence

Saying that AI is an artificial intelligence doesn't really tell you anything meaningful, which is why so many discussions and disagreements arise over this term. Yes, you can argue that what occurs is artificial, not having come from a natural source. However, the intelligence part is, at best, ambiguous. People define intelligence in many different ways. However, you can say that intelligence involves certain mental exercises composed of the following activities:

- **Learning:** Having the ability to obtain and process new information.
- **Reasoning:** Being able to manipulate information in various ways.
- **Understanding:** Considering the result of information manipulation.
- **Grasping truths:** Determining the validity of the manipulated information.
- **Seeing relationships:** Divining how validated data interacts with other data.
- **Considering meanings:** Applying truths to particular situations in a manner consistent with their relationship.
- **Separating fact from belief:** Determining whether the data is adequately supported by provable sources that can be demonstrated to be consistently valid.

The list could easily get quite long, but even this list is relatively prone to interpretation by anyone who accepts it as viable. As you can see from the list, however, intelligence often follows a process that a computer system can mimic as part of a simulation:

1. Set a goal based on needs or wants.
2. Assess the value of any currently known information in support of the goal.
3. Gather additional information that could support the goal.
4. Manipulate the data such that it achieves a form consistent with existing information.
5. Define the relationships and truth values between existing and new information.
6. Determine whether the goal is achieved.
7. Modify the goal in light of the new data and its effect on the probability of success.
8. Repeat Steps 2 through 7 as needed until the goal is achieved (found true) or the possibilities for achieving it are exhausted (found false).

REMEMBER

Even though you can create algorithms and provide access to data in support of this process within a computer, a computer's capability to achieve intelligence is severely limited. For example, a computer is incapable of understanding anything because it relies on machine processes to manipulate data using pure math in a strictly mechanical fashion. Likewise, computers can't easily separate truth from mistruth. In fact, no computer can fully implement any of the mental activities described in the list that describes intelligence.

When thinking about AI, you must consider the goals of the people who develop an AI. The goal is to mimic human intelligence, not replicate it. A computer doesn't truly think, but it gives the appearance of thinking. However, a computer actually provides this appearance only in the logical/mathematical form of intelligence. A computer is moderately successful in mimicking visual-spatial and bodily-kinesthetic intelligence. A computer has a low, passable capability in interpersonal and linguistic intelligence. Unlike humans, however, a computer has no way to mimic intrapersonal or creative intelligence.

Considering the role of AI

As described in the previous section, the first concept that's important to understand is that AI doesn't really have anything to do with human intelligence. Yes, some AI is modeled to simulate human intelligence, but that's what it is: a simulation. When thinking about AI, notice that an interplay exists between goal seeking, data processing used to achieve that goal, and data acquisition used to better understand the goal. AI relies on algorithms to achieve a result that may or may not have anything to do with human goals or methods of achieving those goals. With this in mind, you can categorize AI in four ways:

- **Acting humanly:** When a computer acts like a human, it best reflects the Turing test, in which the computer succeeds when differentiation between the computer and a human isn't possible (see `http://www.turing.org.uk/scrapbook/test.html` for details). This category also reflects what the media would have you believe that AI is all about. You see it employed for technologies such as natural language processing, knowledge representation, automated reasoning, and machine learning (all four of which must be present to pass the test).

  The original Turing Test didn't include any physical contact. The newer, Total Turing Test does include physical contact in the form of perceptual ability interrogation, which means that the computer must also employ both computer vision and robotics to succeed. Modern techniques include the idea of achieving the goal rather than mimicking humans completely. For example, the Wright brothers didn't succeed in creating an airplane by precisely copying the flight of birds; rather, the birds provided ideas that led to aerodynamics, which in turn eventually led to human flight. The goal is to fly. Both birds and humans achieve this goal, but they use different approaches.

- **Thinking humanly:** When a computer thinks as a human, it performs tasks that require intelligence (as contrasted with rote procedures) from a human to succeed, such as driving a car. To determine whether a program thinks like a human, you must have some method of determining how humans think, which the cognitive modeling approach defines. This model relies on three techniques:

- **Introspection:** Detecting and documenting the techniques used to achieve goals by monitoring one's own thought processes.
- **Psychological testing:** Observing a person's behavior and adding it to a database of similar behaviors from other persons given a similar set of circumstances, goals, resources, and environmental conditions (among other things).
- **Brain imaging:** Monitoring brain activity directly through various mechanical means, such as Computerized Axial Tomography (CAT), Positron Emission Tomography (PET), Magnetic Resonance Imaging (MRI), and Magnetoencephalography (MEG).

After creating a model, you can write a program that simulates the model. Given the amount of variability among human thought processes and the difficulty of accurately representing these thought processes as part of a program, the results are experimental at best. This category of thinking humanly is often used in psychology and other fields in which modeling the human thought process to create realistic simulations is essential.

- **Thinking rationally:** Studying how humans think using some standard enables the creation of guidelines that describe typical human behaviors. A person is considered rational when following these behaviors within certain levels of deviation. A computer that thinks rationally relies on the recorded behaviors to create a guide as to how to interact with an environment based on the data at hand. The goal of this approach is to solve problems logically, when possible. In many cases, this approach would enable the creation of a baseline technique for solving a problem, which would then be modified to actually solve the problem. In other words, the solving of a problem in principle is often different from solving it in practice, but you still need a starting point.
- **Acting rationally:** Studying how humans act in given situations under specific constraints enables you to determine which techniques are both efficient and effective. A computer that acts rationally relies on the recorded actions to interact with an environment based on conditions, environmental factors, and existing data. As with rational thought, rational acts depend on a solution in principle, which may not prove useful in practice. However, rational acts do provide a baseline upon which a computer can begin negotiating the successful completion of a goal.

HUMAN VERSUS RATIONAL PROCESSES

Human processes differ from rational processes in their outcome. A process is *rational* if it always does the right thing based on the current information, given an ideal performance measure. In short, rational processes go by the book and assume that "the book" is actually correct. Human processes involve instinct, intuition, and other variables that don't necessarily reflect the book and may not even consider the existing data. As an example, the rational way to drive a car is to always follow the laws. However, traffic isn't rational. If you follow the laws precisely, you end up stuck somewhere because other drivers aren't following the laws precisely. To be successful, a self-driving car must therefore act humanly, rather than rationally.

You find AI used in a great many applications today. The only problem is that the technology works so well that you don't even know it exists. In fact, you might be surprised to find that many devices in your home already make use of this technology. The uses for AI number in the millions — all safely out of sight even when they're quite dramatic in nature. Here are just a few of the ways in which you might see AI used:

- **Fraud detection:** You get a call from your credit card company asking whether you made a particular purchase. The credit card company isn't being nosy; it's simply alerting you to the fact that someone else could be making a purchase using your card. The AI embedded within the credit card company's code detected an unfamiliar spending pattern and alerted someone to it.
- **Resource scheduling:** Many organizations need to schedule the use of resources efficiently. For example, a hospital may have to determine where to put a patient based on the patient's needs, availability of skilled experts, and the amount of time the doctor expects the patient to be in the hospital.
- **Complex analysis:** Humans often need help with complex analysis because there are literally too many factors to consider. For example, the same set of symptoms could indicate more than one problem. A doctor or other expert might need help making a diagnosis in a timely manner to save a patient's life.
- **Automation:** Any form of automation can benefit from the addition of AI to handle unexpected changes or events. A problem with some types of automation today is that an unexpected event, such as an object in the wrong place, can actually cause the automation to stop. Adding AI to the automation can allow the automation to handle unexpected events and continue as though nothing happened.

- **Customer service:** The customer service line you call today may not even have a human behind it. The automation is good enough to follow scripts and use various resources to handle the vast majority of your questions. With good voice inflection (provided by AI as well), you may not even be able to tell that you're talking with a computer.
- **Safety systems:** Many of the safety systems found in machines of various sorts today rely on AI to take over the vehicle in a time of crisis. For example, many automatic braking systems rely on AI to stop the car based on all the inputs that a vehicle can provide, such as the direction of a skid.
- **Machine efficiency:** AI can help control a machine in such a manner as to obtain maximum efficiency. The AI controls the use of resources so that the system doesn't overshoot speed or other goals. Every ounce of power is used precisely as needed to provide the desired services.

Focusing on machine learning

Machine learning is one of a number of subsets of AI and the only one this book discusses. In machine learning, the goal is to create a simulation of human learning so that an application can adapt to uncertain or unexpected conditions. To perform this task, machine learning relies on algorithms to analyze huge datasets.

REMEMBER

Currently, machine learning can't provide the sort of AI that the movies present (a machine can't intuitively learn as a human can); it can only simulate specific kinds of learning, and only in a narrow range at that. Even the best algorithms can't think, feel, present any form of self-awareness, or exercise free will. Characteristics that are basic to humans are frustratingly difficult for machines to grasp because of these limits in perception. Machines aren't self-aware.

What machine learning can do is perform predictive analytics far faster than any human can. As a result, machine learning can help humans work more efficiently. The current state of AI, then, is one of performing analysis, but humans must still consider the implications of that analysis: making the required moral and ethical decisions. The essence of the matter is that machine learning provides just the learning part of AI, and that part is nowhere near ready to create an AI of the sort you see in films.

REMEMBER

The main point of confusion between learning and intelligence is that people assume that simply because a machine gets better at its job (it can learn), it's also aware (has intelligence). Nothing supports this view of machine learning. The same phenomenon occurs when people assume that a computer is purposely causing problems for them. The computer can't assign emotions and therefore acts only upon the input provided and the instruction contained within an application to process that input. A true AI will eventually occur when computers can finally emulate the clever combination used by nature:

- **Genetics:** Slow learning from one generation to the next
- **Teaching:** Fast learning from organized sources
- **Exploration:** Spontaneous learning through media and interactions with others

To keep machine learning concepts in line with what the machine can actually do, you need to consider specific machine learning uses. It's useful to view uses of machine learning outside the normal realm of what many consider the domain of AI. Here are a few uses for machine learning that you might not associate with an AI:

- **Access control:** In many cases, access control is a yes-or-no proposition. An employee smartcard grants access to a resource in much the same way as people have used keys for centuries. Some locks do offer the capability to set times and dates that access is allowed, but such coarse-grained control doesn't really answer every need. By using machine learning, you can determine whether an employee should gain access to a resource based on role and need. For example, an employee can gain access to a training room when the training reflects an employee role.
- **Animal protection:** The ocean might seem large enough to allow animals and ships to cohabitate without problem. Unfortunately, many animals get hit by ships each year. A machine learning algorithm could allow ships to avoid animals by learning the sounds and characteristics of both the animal and the ship. (The ship would rely on underwater listening gear to track the animals through their sounds, which you can actually hear a long distance from the ship.)
- **Predicting wait times:** Most people don't like waiting when they have no idea of how long the wait will be. Machine learning allows an application to determine waiting times based on staffing levels, staffing load, complexity of the problems the staff is trying to solve, availability of resources, and so on.

Moving from machine learning to deep learning

Deep learning is a subset of machine learning, as previously mentioned. In both cases, algorithms appear to learn by analyzing huge amounts of data (however, learning can occur even with tiny datasets in some cases). However, deep learning varies in the depth of its analysis and the kind of automation it provides. You can summarize the differences between the two like this:

- **A completely different paradigm:** Machine learning is a set of many different techniques that enable a computer to learn from data and to use what it learns to provide an answer, often in the form of a prediction. Machine learning relies on different paradigms such as using statistical analysis, finding analogies in data, using logic, and working with symbols. Contrast the myriad techniques used by machine learning with the single technique used by deep learning, which mimics human brain functionality. It processes data using computing units, called *neurons,* arranged into ordered sections, called *layers.* The technique at the foundation of deep learning is the *neural network.*
- **Flexible architectures:** Machine learning solutions offer many knobs (adjustments) called *hyperparameters* that you tune to optimize algorithm learning from data. Deep learning solutions use hyperparameters, too, but they also use multiple user-configured layers (the user specifies number and type). In fact, depending on the resulting neural network, the number of layers can be quite large and form unique neural networks capable of specialized learning: Some can learn to recognize images, while others can detect and parse voice commands. The point is that the term *deep* is appropriate; it refers to the large number of layers potentially used for analysis. The architecture consists of the ensemble of different neurons and their arrangement in layers in a deep learning solution.
- **Autonomous feature definition:** Machine learning solutions require human intervention to succeed. To process data correctly, analysts and scientist use a lot of their own knowledge to develop working algorithms. For instance, in a machine learning solution that determines the value of a house by relying on data containing the wall measures of different rooms, the machine learning algorithm won't be able to calculate the surface of the house unless the analyst specifies how to calculate it beforehand. Creating the right information for a machine learning algorithm is called feature creation, which is a time-consuming activity. Deep learning doesn't require humans to perform any feature-creation activity because, thanks to its many layers, it defines its own best features. That's also why deep learning outperforms machine learning in otherwise very difficult tasks such as recognizing voice and images, understanding text, or beating a human champion at the Go game (the digital form of the board game in which you capture your opponent's territory).

TECHNICAL STUFF

You need to understand a number of issues with regard to deep learning solutions, the most important of which is that the computer still doesn't understand anything and isn't aware of the solution it has provided. It simply provides a form of feedback loop and automation conjoined to produce desirable outputs in less time than a human could manually produce precisely the same result by manipulating a machine learning solution.

The second issue is that some benighted people have insisted that the deep learning layers are hidden and not accessible to analysis. This isn't the case. Anything a computer can build is ultimately traceable by a human. In fact, the General Data Protection Regulation (GDPR) (`https://eugdpr.org/`) requires that humans perform such analysis (see the article at `https://www.pcmag.com/commentary/361258/how-gdpr-will-impact-the-ai-industry` for details). The requirement to perform this analysis is controversial, but current law says that someone must do it.

The third issue is that self-adjustment goes only so far. Deep learning doesn't always ensure a reliable or correct result. In fact, deep learning solutions can go horribly wrong (see the article at `https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist` for details). Even when the application code doesn't go wrong, the devices used to support the deep learning can (see the article at `https://www.pcmag.com/commentary/361918/learning-from-alexas-mistakes?source=SectionArticles` for details). Even so, with these problems in mind, you can see deep learning used for a number of extremely popular applications, as described at `https://medium.com/@vratulmittal/top-15-deep-learning-applications-that-will-rule-the-world-in-2018-and-beyond-7c6130c43b01`.

Using Deep Learning in the Real World

Make no mistake: People do use deep learning in the real world to perform a broad range of tasks. For example, many automobiles today use a voice interface. The voice interface can perform basic tasks, even right from the outset. However, the more you talk to it, the better the voice interface performs. The interface learns as you talk to it — not only the manner in which you say things, but also your personal preferences. The following sections give you a little information on how deep learning works in the real world.

Understanding the concept of learning

When humans learn, they rely on more than just data. Humans have intuition, along with an uncanny grasp of what will and what won't work. Part of this inborn knowledge is instinct, which is passed from generation to generation through DNA. The way humans interact with input is also different from what a computer will do. When dealing with a computer, learning is a matter of building a database consisting of a neural network that has weights and biases built into it to ensure proper data processing. The neural network then processes data, but not in a manner that's even remotely the same as what a human will do.

Performing deep learning tasks

Humans and computers are best at different tasks. Humans are best at reasoning, thinking through ethical solutions, and being emotional. A computer is meant to process data — lots of data — really fast. You commonly use deep learning to solve problems that require looking for patterns in huge amounts of data — problems whose solution is nonintuitive and not immediately noticeable. The article at `http://www.yaronhadad.com/deep-learning-most-amazing-applications/` tells you about 30 different ways in which people are currently using deep learning to perform tasks. In just about every case, you can sum up the problem and its solution as processing huge amounts of data quickly, looking for patterns, and then relying on those patterns to discover something new or to create a particular kind of output.

Employing deep learning in applications

Deep learning can be a stand-alone solution, as illustrated in this book, but it's often used as part of a much larger solution and mixed with other technologies. For example, mixing deep learning with expert systems is not uncommon. The article at `https://www.sciencedirect.com/science/article/pii/0167923694900213` describes this mixture to some degree. However, real applications are more than just numbers generated from some nebulous source. When working in the real world, you must also consider various kinds of data sources and understand how those data sources work. A camera may require a different sort of deep learning solution to obtain information from it, while a thermometer or proximity detector may output simple numbers (or analog data that requires some sort of processing to use). Real-world solutions are messy, so you need to be prepared with more than one solution to problems in your toolkit.

Considering the Deep Learning Programming Environment

You may automatically assume that you must jump through a horrid set of hoops and learn esoteric programming skills to delve into deep learning. It's true that you gain flexibility by writing applications using one of the programming languages that work well for deep learning needs. However, Deep Learning Studio (see the article at `https://towardsdatascience.com/is-deep-learning-without-programming-possible-be1312df9b4a` for details) and other products like it are enabling people to create deep learning solutions without programming. Essentially, such solutions involve describing what you want as output by defining a model graphically. These kinds of solutions work well for straightforward problems that others have already had to solve, but they lack the flexibility to do something completely different — a task that requires something more than simple analysis.

Deep learning solutions in the cloud, such as that provided by Amazon Web Services (AWS) (`https://aws.amazon.com/deep-learning/`), can give you additional flexibility. These environments also tend to make the development environment simpler by providing as much or little support as you want. In fact, AWS provides support for various kinds of serverless computing (`https://aws.amazon.com/serverless/`) in which you don't worry about any sort of infrastructure. However, these solutions can become quite expensive. Even though they give you greater flexibility than using a premade solution, they still aren't as flexible as using an actual development environment.

You have other nonprogramming solutions to consider as well. For example, if you want power and flexibility, but don't want to program to get it, you could rely on a product such as MATLAB (`https://www.mathworks.com/help/deeplearning/ug/deep-learning-in-matlab.html`), which provide a deep learning toolkit. MATLAB and certain other environments do focus more on the algorithms you want to use, but to gain full functionality from them, you need to write scripts as a minimum, which means that you're dipping your toe into programming to some extent. A problem with these environments is that they can also be lacking in the power department, so some solutions may take longer than you expect.

REMEMBER

At some point, no matter how many other solutions you try, serious deep learning problems will require programming. When reviewing the choices online, you often see AI, machine learning, and deep learning all lumped together. However, just as the three technologies work at different levels, so do the programming languages that you require. A good deep learning solution will require the use of multiprocessing, preferably using a Graphics Processing Unit (GPU) with lots of cores. Your language of choice must also support the GPU through a compatible library or package. So, just choosing a language usually isn't enough; you need to investigate further to ensure that the language will actually meet your needs. With this caution in mind, here are the top languages (in order of popularity, as of this writing) for deep learning use (as defined at `https://www.datasciencecentral.com/profiles/blogs/which-programming-language-is-considered-to-be-best-for-machine`):

- Python
- R
- MATLAB (the scripting language, not the product)
- Octave

The only problem with this list is that other developers have other opinions. Python and R normally appear at the top of everyone's lists, but after that you can find all sorts of opinions. The article at `https://www.geeksforgeeks.org/top-5-best-programming-languages-for-artificial-intelligence-field/` gives you some alternative ideas. When choosing a language, you usually have to consider these issues:

- **Learning curve:** Your experiences have a lot to say about what you find easiest to learn. Python is probably the best choice for someone who has programmed for a number of years, but R might be the better choice for someone who has already experienced functional programming. MATLAB or Octave might work best for a math professional.
- **Speed:** Any sort of deep learning solution will require a lot of processing power. Many people say that because R is a statistical language, it offers more in the way of statistical support and usually provides a faster result. Actually, Python's support for great parallel programming probably offsets this advantage when you have the required hardware.
- **Community support:** Many forms of community support exist, but the two that are most important for deep learning are help in defining a solution and access to a wealth of premade programming aids. Of the four, Octave probably provides the least in the way of community support; Python provides the most.
- **Cost:** How much a language costs depends on the kind of solution you choose and where you run it. For example, MATLAB is a proprietary product that requires purchase, so you have something invested immediately when using MATLAB. However, even though the other languages are free at the outset, you can find hidden costs, such as running your code in the cloud to gain access to GPU support.
- **DNN Frameworks support:** A framework can make working with your language significantly easier. However, you have to have a framework that works well with all other parts of your solution. The two most popular frameworks are TensorFlow and PyTorch. Oddly enough, Python is the only language that supports both, so it offers you the greatest flexibility. You use Caffe with MATLAB and TensorFlow with R.
- **Production ready:** A language has to support the kind of output needed for your project. In this regard, Python shines because it's a general-purpose language. You can create any sort of application needed with it. However, the more specific environments provided by the other languages can be incredibly helpful with some projects, so you need to consider all of them.

Overcoming Deep Learning Hype

Previous parts of this chapter discuss some issues with the perception of deep learning, such as some people's belief that it appears everywhere and does everything. The problem with deep learning is that it has been a victim of its own media campaign. Deep learning solves specific sorts of problems. The following sections help you avoid the hype associated with deep learning.

Discovering the start-up ecosystem

Using a deep learning solution is a lot different from creating a deep learning solution of your own. The infographic at `https://www.analyticsvidhya.com/blog/2018/08/infographic-complete-deep-learning-path/` gives you some ideas on how to get started with Python (a process this book simplifies for you). The educational requirements alone can take a while to fulfill. However, after you have worked through a few projects on your own, you begin to realize that the hype surrounding deep learning extends all the way to the start of setup. Deep learning isn't a mature technology, so trying to use it is akin to building a village on the moon or deep diving the Marianas Trench. You're going to encounter issues, and the technology will constantly change on you.

Some of the methods used to create deep learning solutions need work, too. The concept of a computer actually learning anything is false, as is the idea that computers have any form of sentience at all. The reason that Microsoft, Amazon, and other vendors have problems with deep learning is that even their engineers have unrealistic expectations. Deep learning comes down to math and pattern matching — really fancy math and pattern matching, to be sure, but the idea that it's anything else is simply wrong.

Knowing when not to use deep learning

Deep learning is only one way to perform analysis, and it's not always the best way. For example, even though expert systems are considered old technology, you can't really create a self-driving car without one for the reasons described at `https://aitrends.com/ai-insider/expert-systems-ai-self-driving-cars-crucial-innovative-techniques/`. A deep learning solution turns out to be way too slow for this particular need. Your car will likely contain a deep learning solution, but you're more likely to use it as part of the voice interface.

AI in general and deep learning in particular can make the headlines when the technology fails to live up to expectations. For example, the article at `https://www.techrepublic.com/article/top-10-ai-failures-of-2016/` provides a list of AI failures, some of which relied on deep learning as well. It's a mistake to think that deep learning can somehow make ethical decisions or that it will choose the right course of action based on feelings (which no machine has). Anthropomorphizing the use of deep learning will always be a mistake. Some tasks simply require a human.

Speed and the capability to think like a human are the top issues for deep learning, but there are many more. For example, you can't use deep learning if you don't have sufficient data to train it. In fact, the article at `https://www.sas.com/en_us/insights/articles/big-data/5-machine-learning-mistakes.html` offers a list of five common mistakes that people make when getting into machine learning and deep learning environments. If you don't have the right resources, deep learning will never work.