Chapter 3

What is Computational Neuroscience?

3.1 Vocabulary

- Algorithm
- Bottom-up processing
- Computational neuroscience
- Computational theory
- Emergent phenomena
- Hardware and implementation
- Hebbian learning
- Reconstruction
- Reductionism
- Reductionism
- Representation
- Top-down processing
- Turing machine

3.2 Introduction

Look at this picture of a desk on the page in front of you. As you look at it, your brain is somehow able to turn the raw sense data coming from your eyes into a judgment about the identity of the object in the image. If we want to build a machine that can demonstrate a similar capacity to judge — then we need to be able to model and understand the mechanisms that are at work when we see the desk. We are now left with at least two questions: *(1) How might we create a computer program that performs the same cognitive tasks as us humans?* And (2) *What can such as program tell us about the mechanisms at work in our own human brains?* While these questions may not be exhaustive of the concerns of computational neuroscience, they should at least give you a taste of some of the issues the discipline grapples with on a regular basis.
For more information please explore.

3.3 What is computational neuroscience?

Computational neuroscience is an interdisciplinary field that applies the principles of mathematics, philosophy, and computer science to study the inner workings of the brain. Computer models are critical to computational neuroscience, because they allow experiments to be conducted in a highly controlled and replicable fashion. In this context, a "model" is a simplified and simulated version of a system that tries to guess how the actual (simulated) system would behave in the real world.

For example, suppose a computational neuroscientist wants to understand how the human brain begins to make sense of sounds. A computer model could be constructed for this purpose, because many disparate aspects of the hearing parts of the brain have been measured. Such measurements would make constructing a useful computer model possible, because they would constrain the design model. In other words, our researcher could design a model where the simulated features match the measurements of the corresponding real features of the brain. This model could be useful, because our researcher has access to all the features of the computer model—including those that could not be easily and ethically measured in the actual human brain. This utility would be borne out in plausible inferences about currently unmeasured properties and behaviors of the brain.

While it is true that the inferences we've just discussed can not be made with complete certainty, they can be instrumental in guiding future research as new technology (and sources of funding) become available. Even if a given model ends up not holding up to future data, the model could still prove useful for developing artificial intelligences. With the potential of computer models of the brain in mind, it may be tempting to think you could build a model that truly "understands" in the same way that a person does. The question of whether or not such a model is possible is a matter of much debate, so suffice it to say we will only briefly survey the issues here.

On the side arguing that a computer model could never truly understand something (e.g. the Chinese language) the way a human does is the philosopher John Searle. Searle makes use of his famous "Chinese Room Argument" to suggest that merely following a set of rules to produce a desired result from a given input is not enough to count as true understanding. For example, Searle would argue that using a big book of rules for writing Chinese answers that respond to Chinese questions is not the same thing as having a natural conversation in Chinese.[1]

This may seem intuitively true, but many of Searle's opponents[2] argue that Searle's alleged argument is only intuitively true. That is to say that Searle is merely provoking intuitions rather than supplying premises or facts that lead to his desired conclusion.

The motivations for this debate could be explained in terms of the level of organization at which Searle and his opponents appear to be thinking.

While Searle published his paper on the "Chinese Room Argument" in 1980, the conversation of whether computers would be able to fully understand humans had been ongoing for many decades prior. In 1936, Alan Turing, created the theory of the Turing test to find when there was an equivalence between artificial intelligence and humans. Turing found the limitations of computation by understanding the limitations of humanity, and thus he created the **Turing Machine.** This machine paved way for the turing test, that tested whether a human could determine whether they were interacting with another machine or another human. Hence, the question of how to tell when artificial intelligence will be comparable to human intelligence is an ongoing problem today.

[1] See https://plato.stanford.edu/entries/chinese-room/ for more on this thought experiment.
[2] E.g. Dennett, Thagard, and Pinker to name a few.

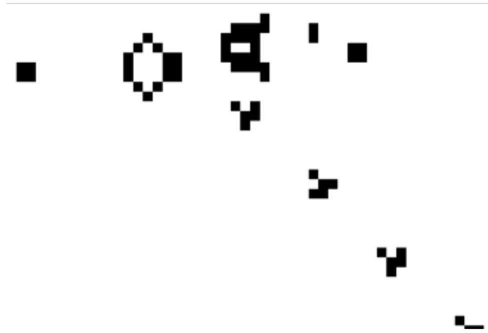

Figure 3.1: Example of Conway's Game of Life

3.4 Levels of organization

In 1982, David Marr, introduced a new approach to analysis. He believed that there were three levels in the model of the brain. The first level is the **computational theory**, , which is a description of the information going into the system, and the corresponding output desired from the system. An example of this is addition. The input is two numbers, and the desired output is the sum of those numbers. The second level consists of the representational scheme and the algorithm. The **representational scheme** is the description of the functional elements that are used in the computation, while the **algorithm** is the set of operations that are performed with or by those elements in order to carry out the transformation specified by the computational theory. One example is a cookbook recipe; this will define a step by step process (an algorithm) for how to produce a product given a set of clearly defined ingredients (a representational scheme). The third level is **hardware implementation**, which refers to the physical machinery that realizes the algorithm.

The goal of computational neuroscience is to be able to replicate the functions of the brain, such as the one you just performed, in a non-organic setting. One of the ways to do this is through computer programming software, such as the application Python.

Consider the following basic code:

```
In [1]: 2+2

Out[1]: 4
```

Here is an example of an addition problem coded in Python. We can see the input, the algorithm which is transforming the input, and then the correct output of 4. Given the properties of coding software such as Python, which takes input, and runs it through a set algorithm. Is this similar to the way we humans do it? If so, how? If not, why not?

3.5 Applications of computational neuroscience

As we delve deeper into Computational Neuroscience, we will find that the field has a variety of potential applications when it comes to understanding how cognition happens. Computational Neuroscience, perhaps most importantly, allows us to create models of our cognitive processes, such that they are able to capture the basis of complex phenomena in a simple way. The brain is an extremely complex organ, and while we may not always understand all of its architecture and functionality, by using methods of Computational Neuroscience we are able to abstract certain notions to the extent that they become comprehensible. In doing so, we develop the ability to understand interactions between neurons in the brain and begin seeing the nature of certain causal relationships. Furthermore, we can begin to predict how complex systems in the brain will behave when presented with particular stimuli. This may all seem rather abstract, so let's think of an example we often take for granted: Vision. How is it that we are capable of recognizing a variety of highly specific things and distinguishing them from one another? What constitutes recognition and the neural processes behind it? How do we decide what deserves recognition and what doesn't? Why is it so difficult to replicate these seemingly innate processes in a robot? All these questions can be addressed using Computational Neuroscience. We must, however, keep one thing in mind. When creating models, the correct level of simplicity is difficult if not impossible to discern. So, it is important to engage in criticism and scrutiny when developing the simplest and most efficient model one can think of.

You may be saying to yourself, "Ok, well all this is great in theory, but where do we begin?". We start by trying to establish **Emergent Phenomena.** Emergent Phenomena allow us to reframe composite systems such that we are able to understand their underlying mechanisms in simpler terms. To clarify, an Emergent Phenomenon can highlight both the mechanics and the nature of a particular system. For instance, a normal car cannot function properly unless it has four wheels. However, the car also derives part of its "car-ness" from the fact that it has four wheels. There are two primary schools of thought in this domain: **Reductionism** and **Reconstructionism.** Reductionism is the idea that in order to understand a given complex system, we must be able to reduce it to its simplest form, while Reconstructionism claims that we go in the opposite direction and reconstruct the system such that we are able to capture its complexity. After all, when we create models of the brain, it is not sufficient to explain their architecture. We must also show how the architecture gives rise to certain relationships and interactions. To do this, we can design our models using one of the following approaches: **Top-down processing** or **Bottom-up**

Figure 3.2: One example of emergent phenomena are the flight patterns of geese. One goose alone flies as it wishes but a collection of geese come together to form a v-shaped pattern that affects the overall movement of the geese.

processing. Top-down processing makes us design our models with a certain purpose, or goal in mind. Bottom-up processing leads us to establish a base of information or data before creating the model, and then creating the model and its purpose off of what we have collected.

3.6 The future of computational neuroscience

Figure 3.3: This image can be approached in two different ways. For bottom-up processing, we can see the letters first and then figure out the words. For top-down processing, we can see the words and decide what each letter is.

Given all this information, what is the future of Computational Neuroscience? Interest in the field is increasing steadily and everyday the range of its possible applications grows. In 2013, the Obama administration began the BRAIN initiative, a program designed to facilitate the development of innovative technologies that allow for a well-rounded and versatile understanding of brain function. In

2005, a Swiss team of scientists began another initiative called the Blue Brain project whereby they reconstructed the mammalian brain using simulations in order to generate a comprehension of the basic underlying principles of brain function and architecture. Computational Neuroscience becomes more relevant everyday and allows us to tackle difficult issues like the complexities of **Hebbian Learning** and Neural Networks. The implications of the field are perhaps unparalleled by any other fields of scientific inquiry, in that they may hold the answers to the creation of true Artificial Intelligence and the understanding of Consciousness and perception in the human brain.

3.7 Summary

Every person has a brain, but that doesn't mean we understand what the brain is, or what it does. The brain is a complex organism with many facets to understand, and to fully comprehend; one must understand the mechanics and the reasoning behind each component. Whether studying in a top-down process, or a bottom-up process, one must keep in mind Marr's three levels of investigation: computational theory, representation and algorithm, and hardware and implementation. Computers and brains are different organisms, but by studying them alongside one another, a deeper understanding can be elicited. Turing introduced the world to the concept of using humans as a test of how to understand artificial intelligence, but we can now use artificial intelligence to understand the brain as well.

3.8 Exercises:

1. Try out Conway's Game of Life at https://bitstorm.org/gameoflife/ to explore more into the implications of the Turing Machine and computer simulations.

2. What are at least two methods you could use to add two two integers such as 5 and 5? Now consider how you solved these two problems. Which of Marr's levels was changing?

3. Explain the difference between the top-down and bottom-up philosophy.

4. Which better explains and represents cognitive phenomena: the top-down approach, or the bottom-up approach?

5. Characterize the concept of emergent phenomena. Is human sentience that kind of phenomenon?

6. Does artificial intelligence have to have emotions to be sentient?

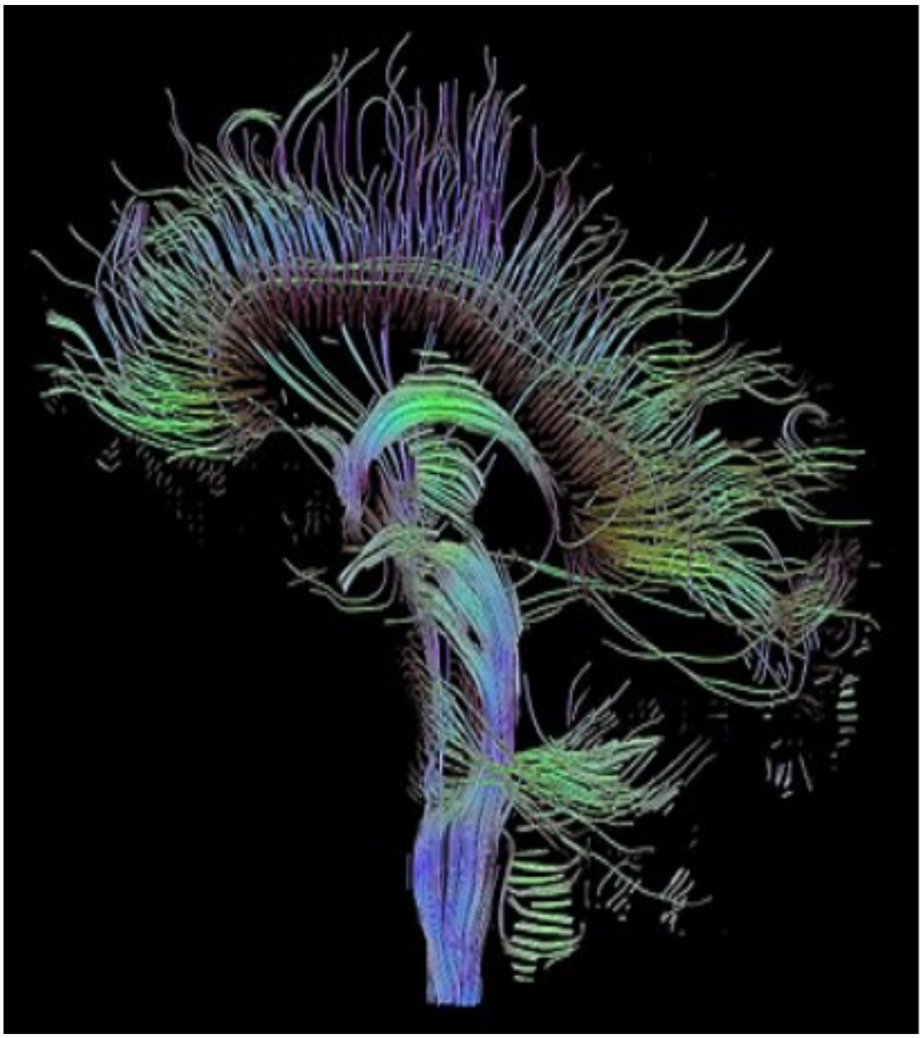

Figure 3.4: Applying Hebbian theory to computational neuroscience allows us to envision the connections between various neurons in the brain, displayed here is what is called a 'connectome'.