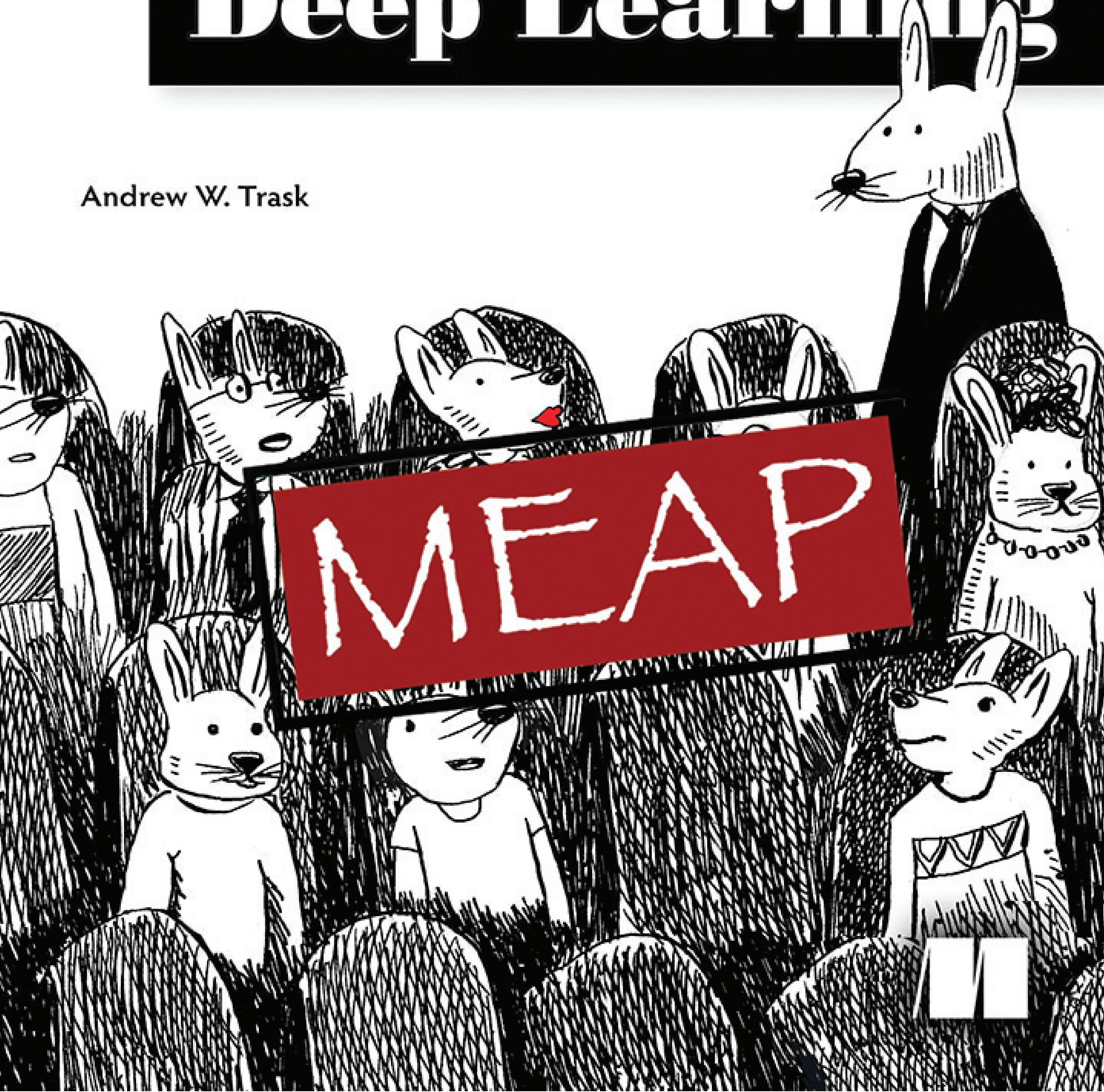
grokking

Deep Learning

Andrew W. Trask



MEAP

MEAP Edition
Manning Early Access Program

Grokking Deep Learning

Version 4

For more information on this and other Manning titles go to
www.manning.com

welcome

Thank you so much for purchasing *Grokking Deep Learning*. This book will teach you the fundamentals of Deep Learning from an intuitive perspective, so that you can understand how machines learn using Deep Learning. This book is not focused on learning a framework such as Torch, TensorFlow, or Keras. Instead, it is focused on teaching you the Deep Learning methods behind well known frameworks. Everything will be built from scratch using only Python and numpy (a matrix library). In this way, you will understand every detail that goes into training a neural network, not just how to use a code library. You should consider this book a pre-requisite to mastering one of the major frameworks.

There are many other resources for learning Deep Learning. I'm glad that you came to this one, as I have intentionally written it with what I believe is the lowest barrier to entry possible. No knowledge of Linear Algebra, Calculus, Convex Optimization, or even Machine Learning is assumed. Everything from these subjects that is necessary to understand Deep Learning will be explained as we go. If you have passed high school mathematics and hacked around in python, you're ready for this book, and when you complete this book, you will be ready to master a major deep learning framework.

Finally, as this is the MEAP, if there is any point in these first few chapters that something does not make sense, it is my hope that you would tweet your questions to me @iamtrask. I would be happy to help, and more importantly, I want to know if any section of the book is not fulfilling my personal commitment to the lowest barrier to entry possible so that I can adjust it for the final published work. Please, don't hesitate to reach out if you have questions.

These first three chapters will be walking you from a general introduction to Deep Learning all the way through to building your first working neural network. In these chapters, you will get a firm grasp on the philosophy behind how machines can learn the world you present to them. It's an exciting thing to see happen, and perhaps even more exciting, you will understand every nook and cranny of what makes this learning possible.

It is an honor to have your time and attention.
—Andrew Trask

brief contents

PART 1: NEURAL NETWORK BASICS

PART 2: IMAGE DEEP LEARNING

PART 3: LANGUAGE DEEP LEARNING (TEXT AND AUDIO)

PART 4: NEURAL NETWORKS THAT PLAY GAMES

Introducing Deep Learning

Why you should learn it

1

IN THIS CHAPTER

Why you should learn deep learning

Why you should read this book

What you need to get started

Welcome to Grokking Deep Learning

You're about to learn some of the most valuable skills of the century!

I'm very excited that you're here! You should be too! Deep Learning represents an exciting intersection of Machine Learning and Artificial Intelligence and a very significant impact on society and industry. The methods discussed in this book are changing the world all around you. From optimizing the engine of your car to deciding which content you view on social media, it's everywhere. It's powerful. And quite frankly, it's fun!

Why should you learn Deep Learning?

It is a powerful tool for the *incremental automation of intelligence.*

From the beginning of time, humans have been building better and better tools to understand and control the environment around us. Deep Learning is today's chapter in this story of innovation. Perhaps what makes this chapter so compelling is that this field is more of a *mental* innovation than a *mechanical one*. Much like its sister fields in Machine Learning, Deep Learning seeks to *automate intelligence* bit by bit, and in the past few years it has achieved enourmous success and progress in this endeavor, exceeding previous records in Computer Vision, Speech Recognition, Machine Translation, and many other tasks. This is particularly extraordinary given that Deep Learning seems to use *largely the same brain-inspired algorithm* (Neural Networks) for achieving these accomplishments across a vast number of fields. This has lead to tremendous excitement that perhaps we have in fact discovered more than just a great tool, but a window into our own minds as well.

Deep Learning has the potential for significant automation of *skilled labor.*

There is a substantial amount of hype around the potential impacts of Deep Learning if the current trend of progress is extrapolated at varying speeds. While many of these predictions are over-zealous, there is one that I think merits your consideration, job displacement. I think that this claim stands out from the rest for no other reason than if Deep Learning's innovations stopped *today*, there would already be an incredible impact on skilled labor around the globe. Call center operators, taxi drivers, and low-level business analysts are compelling examples where Deep Learning can provide a low-cost alternative. Fortunately, the economy doesn't turn on a dime, but in many ways we are already past the point of concern with the current power of the technology. It is my hope that you (and people you know) will be enabled by this book to transition from perhaps one of the industries facing disruption into an industry ripe with growth and prosperity, Deep Learning.

It's fun and incredibly creative. You will discover much about what it is to be human by trying to simulate intelligence and creativity.

Personally, I got into Deep Learning because it's fascinating. It's an amazing intersection between man and machine. Unpacking exactly what it means to think, to reason, and to create is enlightening, engaging, and for me it's quite inspiring. Consider having a dataset filled with every painting ever painted, and then using that to teach a machine how to paint like Monet. Insanely, it's possible, and it's mind-bogglingly cool to see how it works.

Why you should read this book!

Uniquely low barrier to entry

The reason you should read this book is the same reason I'm writing it. I don't know of another resource (book, course, large blog series) that teaches Deep Learning **without assuming advanced knowledge of mathematics** (i.e. college degree in a mathy field). Don't get me wrong, there are really good reasons for teaching it using math. Math is, after all, a language. It is certainly more **efficient** to teach Deep Learning using this language, but I don't think it's absolutely necessary to assume advanced knowledge of math in order to become a skilled, knowledgable practitioner who has a firm understanding of the "how" behind Deep Learning. So, why should you learn Deep Learning using this book? I'm going to assume you have a High School level background in math (and that it's rusty), and *explain everything else you need to know as we go along.* Remember multiplication? Remember x-y graphs (the square with lines on it)? Awesome! You'll be fine.

To help you understand what's *inside* a framework (Torch, TensorFlow, etc.)

There are two major groups of Deep Learning educational material (books, courses, etc.). One group is focused around how to use popular frameworks and code libraries such as Torch, Tensorflow, Keras, and others. The other group is focused around teaching Deep Learning itself, otherwise known as the *science under the hood* of these major frameworks. Ultimately, learning about *both* is important. It's like if you want to be a NASCAR driver, you need to learn BOTH about the particular model of car you're driving (the framework), AND about driving itself (the science/skill). However, just learning about a framework is like learning about the pros and cons of a Generation-6 Chevrolet SS before you know what a stick shift is. This book is about teaching you what *Deep Learning* is so that you can then learn a framework.

All math related material will be backed by intuitive *analogies*.

Whenever I encounter a math formula in the wild, I take a two-step approach. The first is to translate its methods into an intuitive *analogy* to the real world. I almost never just take a formula at face value. I break it into *parts*, each with a story of its own. That will be the approach of this book as well. Anytime we encounter a math concept, I'll offer an alternative *analogy* for what the formula is actually doing.

> "Everything should be made as simple as possible, but no simpler"
>
> \- Albert Einstein

Everything after the introduction chapters is "project" based.

If there is one thing I hate when learning something new, it is when I have to question whether or not what I'm learning is useful/relevant. If someone is teaching me everything there is to know about a hammer without actually taking my hand and helping me drive in a nail, then they're not really teaching me how to use a hammer. I know that there are going to be dots that weren't connected, and if I was thrown out into the real world with a hammer, a box of nails, and a bunch of 2x4s, I'm going to have to do some guesswork.

This book is about giving you the wood, nails, and a hammer *before* telling you about what they do. Each lesson is about picking up the tools and building stuff with them, explaining how stuff works along the way. In this way, you don't leave with a list of facts about the various deep learning tools we'll work with, you leave with the ability to use them to solve problems. Furthermore, you will understand the most important part, when and why each tool is appropriate for each problem you want to solve. It is with this knowledge that you will be empowered to pursue a career in research and/or industry.

What you need to get started

Install Jupyter Notebook and the Numpy python library

My absolute favorite place to work is a Jupyter Notebook. One of the most important parts of learning deep learning (for me), is the abiliy to stop a network while it's training and tear apart absolutely every piece and see what it looks like. This is something that jupyter notebook is incredibly useful for. As for numpy, perhaps the most compelling case for why this book leaves nothing out is that we'll only be using a single matrix library. In this way, you will understand **how** everything works, not just how to call a framework. This book teaches Deep Learning from absolute scratch.... soup to nuts. Installation instructions for these two tools can be found at (http://jupyter.org/) for Jupyter and (http://numpy.org) for numpy.

Pass High School Mathematics

There are some mathematical assumptions that are simply out of depth for this book, but the goal of this book is to teach Deep Learning assuming you understand basic algebra.

Find a personal problem you are interested in

This might seem like an optional "need" to get started. I guess it could be, but seriously, I highly, highly recommend finding one. Everyone I know who has become successful at this stuff had some sort of problem they were trying to solve. Learning this stuff was just a "dependency" to solving some other interesting task. For me, it was using Twitter to predict the stock market. It's just something that I thought was really fascinating. It's what drove me to sit down and read the next chapter and build the next prototype. And as it turns out, this field is **so new**... and is changing **so fast**... that if you spend the next couple years chasing one project with these tools, you'll find yourself being one of the leading experts in that *particular problem* faster than you might think. For me, chasing this idea took me from barely knowing anything about programming to a research grant at a hedge fund applying what I learned in around 18 months! Having a problem you're fascinated with that involves using one dataset to predict another is the key catalyst! Go find one!

You'll probably need some Python knowledge

Python is my teaching library of choice, but I'll provide a few others online.

Python is an amazingly intuitive language. I think it just might be the most widely adopted and intuitively readable language yet constructed. Furthermore, the Python community has a passion for simplicity that can't be beat. For these reasons, I want to stick with python for all of the examples. On this book's Github, I'll provide all of the examples in a variety of other languages as well, but for the in-page explanations, we're going to use python.

How much coding experience should you have? At least the basics...

Scan through the Python Codecademy course (https://www.codecademy.com/learn/python). If you can read through the table of contents and feel comfortable with the terms mentioned, you're all set! If not, then just take the course and come back when you're done! It's designed to be a beginner course and it's very well crafted.

Conclusion and Primer for Chapter 2

So, if you've got your Jupyter Notebook in-hand and feel comfortable with the basics of Python, you're ready for the next chapter! As a heads up, Chapter 2 is the last chapter that will be mostly dialogue based (without building something). It's just designed to give you an awareness of the high level vocabulary, concepts, and fields in Artificial Intelligence, Machine Learning, and most importantly... Deep Learning.