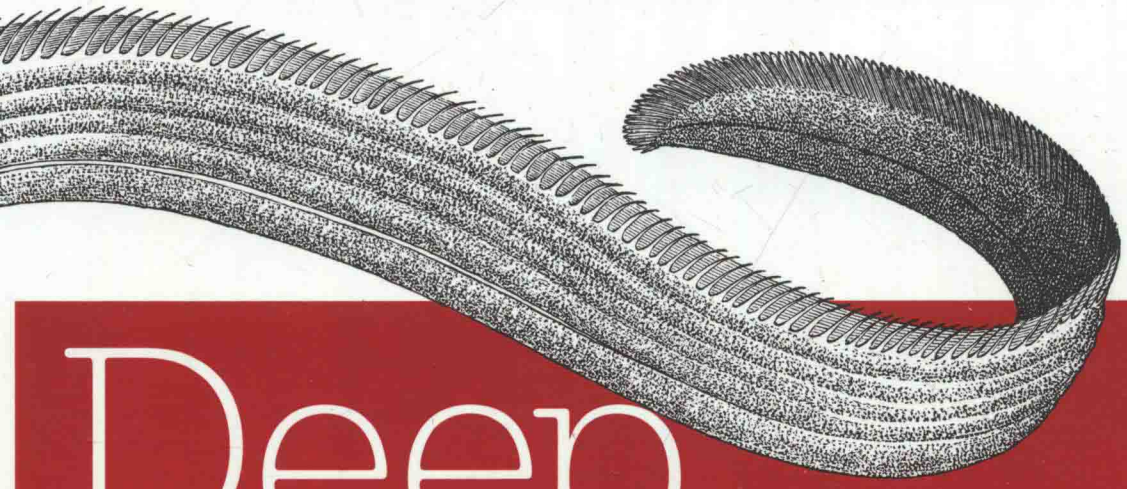
O'REILLY®



Deep Learning

深度学习（影印版）

東南大學出版社

Josh Patterson, Adam Gibson 著

深度学习（影印版）

Deep Learning

Josh Patterson, Adam Gibson 著

Beijing • Boston • Farnham • Sebastopol • Tokyo O'REILLY®

Inc.授权东南大学出版社出版

南京 东南大学出版社

图书在版编目(CIP)数据

深度学习:英文/(美)乔希·帕特森(Josh Patterson),(美)亚当·吉普森(Adam Gibson)著. —影印本. —南京:东南大学出版社,2018.2

书名原文:Deep Learning

ISBN 978-7-5641-7516-0

Ⅰ.①深… Ⅱ.①乔… ②亚… Ⅲ.①机器学习—英文 Ⅳ.①TP181

中国版本图书馆 CIP 数据核字(2017)第 296220 号

图字:10-2017-428 号

© 2017 by O'Reilly Media, Inc.

Reprint of the English Edition, jointly published by O'Reilly Media, Inc. and Southeast University Press, 2018. Authorized reprint of the original English edition, 2017 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly Media, Inc.出版 2017。

英文影印版由东南大学出版社出版 2018。此影印版的出版和销售得到出版权和销售权的所有者 —— O'Reilly Media, Inc.的许可。

版权所有,未得书面许可,本书的任何部分和全部不得以任何形式重制。

深度学习(影印版)

出版发行:东南大学出版社

地　　址:南京四牌楼 2 号　　邮编:210096

出 版 人:江建中

网　　址:http://www.seupress.com

电子邮件:press@seupress.com

印　　刷:常州市武进第三印刷有限公司

开　　本:787 毫米×980 毫米　　16 开本

印　　张:33.25

字　　数:651 千字

版　　次:2018 年 2 月第 1 版

印　　次:2018 年 2 月第 1 次印刷

书　　号:ISBN 978-7-5641-7516-0

定　　价:99.00 元

本社图书若有印装质量问题,请直接与营销部联系。电话(传真):025-83791830

For my sons Ethan, Griffin, and Dane: Go forth, be persistent, be bold.

—J. Patterson

Preface

What's in This Book?

The first four chapters of this book are focused on enough theory and fundamentals to give you, the practitioner, a working foundation for the rest of the book. The last five chapters then work from these concepts to lead you through a series of practical paths in deep learning using DL4J:

- Building deep networks
- Advanced tuning techniques
- Vectorization for different data types
- Running deep learning workflows on Spark

DL4J as Shorthand for Deeplearning4j

We use the names DL4J and Deeplearning4j interchangeably in this book. Both terms refer to the suite of tools in the Deeplearning4j library.

We designed the book in this manner because we felt there was a need for a book covering "enough theory" while being practical enough to build production-class deep learning workflows. We feel that this hybrid approach to the book's coverage fits this space well.

Chapter 1 is a review of machine learning concepts in general as well as deep learning in particular, to bring any reader up to speed on the basics needed to understand the rest of the book. We added this chapter because many beginners can use a refresher or primer on these concepts and we wanted to make the project accessible to the largest audience possible.

Chapter 2 builds on the concepts from Chapter 1 and gives you the foundations of neural networks. It is largely a chapter in neural network theory but we aim to

此为试读,需要完整PDF请访问: www.ertongbook.com

present the information in an accessible way. Chapter 3 further builds on the first two chapters by bringing you up to speed on how deep networks evolved from the fundamentals of neural networks. Chapter 4 then introduces the four major architectures of deep networks and provides you with the foundation for the rest of the book.

In Chapter 5, we take you through a number of Java code examples using the techniques from the first half of the book. Chapters 6 and 7 examine the fundamentals of tuning general neural networks and then how to tune specific architectures of deep networks. These chapters are platform-agnostic and will be applicable to the practitioner of any deep learning library. Chapter 8 is a review of the techniques of vectorization and the basics on how to use DataVec (DL4J's ETL and vectorization workflow tool). Chapter 9 concludes the main body of the book with a review on how to use DL4J natively on Spark and Hadoop and illustrates three real examples that you can run on your own Spark clusters.

The book has many Appendix chapters for topics that were relevant yet didn't fit directly in the main chapters. Topics include:

- Artificial Intelligence
- Using Maven with DL4J projects
- Working with GPUs
- Using the ND4J API
- and more

Who Is "The Practitioner"?

Today, the term "data science" has no clean definition and often is used in many different ways. The world of data science and artificial intelligence (AI) is as broad and hazy as any terms in computer science today. This is largely because the world of machine learning has become entangled in nearly all disciplines.

This widespread entanglement has historical parallels to when the World Wide Web (90s) wove HTML into every discipline and brought many new people into the land of technology. In the same way, all types—engineers, statisticians, analysts, artists—are entering the machine learning fray every day. With this book, our goal is to democratize deep learning (and machine learning) and bring it to the broadest audience possible.

If you find the topic interesting and are reading this preface—*you are the practitioner, and this book is for you.*

Who Should Read This Book?

As opposed to starting out with toy examples and building around those, we chose to start the book with a series of fundamentals to take you on a full journey through deep learning.

We feel that too many books leave out core topics that the enterprise practitioner often needs for a quick review. Based on our machine learning experiences in the field, we decided to lead-off with the materials that entry-level practitioners often need to brush up on to better support their deep learning projects.

You might want to skip Chapters 1 and 2 and get right to the deep learning fundamentals. However, we expect that you will appreciate having the material up front so that you can have a smooth glide path into the more difficult topics in deep learning that build on these principles. In the following sections, we suggest some reading strategies for different backgrounds.

The Enterprise Machine Learning Practitioner

We split this category into two subgroups:

- Practicing data scientist
- Java engineer

The practicing data scientist

This group typically builds models already and is fluent in the realm of data science. If this is you, you can probably skip Chapter 1 and you'll want to lightly skim Chapter 2. We suggest moving on to Chapter 3 because you'll probably be ready to jump into the fundamentals of deep networks.

The Java engineer

Java engineers are typically tasked with integrating machine learning code with production systems. If this is you, starting with Chapter 1 will be interesting for you because it will give you a better understanding of the vernacular of data science. Appendix E should also be of keen interest to you because integration code for model scoring will typically touch ND4J's API directly.

The Enterprise Executive

Some of our reviewers were executives of large Fortune 500 companies and appreciated the content from the perspective of getting a better grasp on what is happening in deep learning. One executive commented that it had "been a minute" since college, and Chapter 1 was a nice review of concepts. If you're an executive, we suggest that

you begin with a quick skim of Chapter 1 to reacclimate yourself to some terminology. You might want to skip the chapters that are heavy on APIs and examples, however.

The Academic

If you're an academic, you likely will want to skip Chapters 1 and 2 because graduate school will have already covered these topics. The chapters on tuning neural networks in general and then architecture-specific tuning will be of keen interest to you because this information is based on research and transcends any specific deep learning implementation. The coverage of ND4J will also be of interest to you if you prefer to do high-performance linear algebra on the Java Virtual Machine (JVM).

Conventions Used in This Book

The following typographical conventions are used in this book:

Italic
: Indicates new terms, URLs, email addresses, filenames, and file extensions.

`Constant width`
: Used for program listings, as well as within paragraphs to refer to program elements such as variable or function names, databases, data types, environment variables, statements, and keywords. Also used for module and package names, and to show commands or other text that should be typed literally by the user and the output of commands.

`Constant width italic`
: Shows text that should be replaced with user-supplied values or by values determined by context.

This element signifies a tip or suggestion.

This element signifies a general note.

This element signifies a warning or caution.

Using Code Examples

Supplemental material (virtual machine, data, scripts, and custom command-line tools, etc.) is available for download at *https://github.com/deeplearning4j/oreilly-book-dl4j-examples*.

This book is here to help you get your job done. In general, if example code is offered with this book, you may use it in your programs and documentation. You do not need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Selling or distributing a CD-ROM of examples from O'Reilly books does require permission. Answering a question by citing this book and quoting example code does not require permission. Incorporating a significant amount of example code from this book into your product's documentation does require permission.

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: "*Deep Learning: A Practitioner's Approach* by Josh Patterson and Adam Gibson (O'Reilly). Copyright 2017 Josh Patterson and Adam Gibson, 978-1-4919-1425-0."

If you feel your use of code examples falls outside fair use or the permission given above, feel free to contact us at *permissions@oreilly.com*.

Administrative Notes

In Java code examples, we often omit the import statements. You can see the full import listings in the actual code repository. The API information for DL4J, ND4J, DataVec, and more is available on this website:

http://deeplearning4j.org/documentation

You can find all code examples at:

https://github.com/deeplearning4j/oreilly-book-dl4j-examples

For more resources on the DL4J family of tools, check out this website:

http://deeplearning4j.org

O'Reilly Safari

Safari (formerly Safari Books Online) is a membership-based training and reference platform for enterprise, government, educators, and individuals.

Members have access to thousands of books, training videos, Learning Paths, interactive tutorials, and curated playlists from over 250 publishers, including O'Reilly Media, Harvard Business Review, Prentice Hall Professional, Addison-Wesley Professional, Microsoft Press, Sams, Que, Peachpit Press, Adobe, Focal Press, Cisco Press, John Wiley & Sons, Syngress, Morgan Kaufmann, IBM Redbooks, Packt, Adobe Press, FT Press, Apress, Manning, New Riders, McGraw-Hill, Jones & Bartlett, and Course Technology, among others.

For more information, please visit *http://oreilly.com/safari*.

How to Contact Us

Please address comments and questions concerning this book to the publisher:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (in the United States or Canada)
707-829-0515 (international or local)
707-829-0104 (fax)

To comment or ask technical questions about this book, send email to *bookquestions@oreilly.com*. If you find any errors or glaring omissions, if you find anything confusing, or if you have any ideas for improving the book, please email Josh Patterson at *jpatterson@floe.tv*.

We have a web page for this book, where we list errata, examples, and any additional information. You can access this page at *http://bit.ly/deep_learning_oreilly*.

For more information about our books, courses, conferences, and news, see our website at *http://www.oreilly.com*.

Find us on Facebook: *http://facebook.com/oreilly*

Follow us on Twitter: *http://twitter.com/oreillymedia*

Watch us on YouTube: *http://www.youtube.com/oreillymedia*

Follow Josh Patterson on Twitter: @jpatanooga (*https://twitter.com/jpatanooga*)

Follow Adam Gibson on Twitter: @agibsonccc (*https://twitter.com/agibsonccc*)

Acknowledgments

Josh

I relied on many folks far smarter than myself to help shape the ideas and review the content in this book. No project the size of DL4J operates in a vacuum, and I relied on many of the community experts and engineers at Skymind to construct many of the ideas and guidelines in this book.

Little did I know that hacking on what became DL4J with Adam (after a chance meeting at MLConf) would end up as a book. To be fair, while I was there from the beginning on DL4J, Adam did far more commits than I ever did. So, to Adam I am considerably grateful in the commitment to the project, commitment to the idea of deep learning on the JVM, and to staying the course during some considerably uncertain early days. And, yes, you were right: ND4J was a good idea.

Writing can be a long, lonely path and I'd like to specifically thank Alex Black for his considerable efforts, not only in reviewing the book, but also for contributing content in the appendixes. Alex's encyclopedia-like knowledge of neural network published literature was key in crafting many of the small details of this book and making sure that all the big and little things were correct. Chapters 6 and 7 just wouldn't be half of what they became without Alex Black.

Susan Eraly was key in helping construct the loss function section and contributed appendix material, as well (many of the equations in this book owe a debt of correctness to Susan), along with many detailed review notes. Melanie Warrick was key in reviewing early drafts of the book, providing feedback, and providing notes for the inner workings of Convolutional Neural Networks (CNNs).

David Kale was a frequent ad hoc reviewer and kept me on my toes about many key network details and paper references. Dave was always there to provide the academic's view on how much rigor we needed to provide while understanding what kind of audience we were after.

James Long was a critical ear for my rants on what should or should not be in the book, and was able to lend a practical viewpoint from a practicing statistician's point of view. Many times there was not a clear correct answer regarding how to communicate a complex topic, and James was my sounding board for arguing the case from multiple sides. Whereas David Kale and Alex Black would frequently remind me of the need for mathematical rigor, James would often play the rational devil's advocate in just how much of it we needed before we "drown the reader in math."

此为试读,需要完整PDF请访问: www.ertongbook.com

Vyacheslav "Raver" Kokorin added quality insight to the development of the Natural Language Processing (NLP) and Word2Vec examples.

I'd like to make note of the support we received from our CEO at Skymind, Chris Nicholson. Chris supported this book at every turn and in no small part helped us with the needed time and resources to make this happen.

I would like to thank the people who contributed appendix chapters: Alex Black (Backprop, DataVec), Vyacheslav "Raver" Kokorin (GPUs), Susan Eraly (GPUs), and Ruben Fiszel (Reinforcement Learning). Other reviewers of the book at various stages include Grant Ingersol, Dean Wampler, Robert Chong, Ted Malaska, Ryan Geno, Lars George, Suneel Marthi, Francois Garillot, and Don Brown. Any errors that you might discover in this book should be squarely placed on my doorstep.

I'd like to thank our esteemed editor, Tim McGovern, for the feedback, notes, and just overall patience with a project that spanned years and grew by three chapters. I felt like he gave us the space to get this right, and we appreciate it.

Following are some other folks I'd like to recognize who had an impact on my career leading up to this book: my parents (Lewis and Connie), Dr. Andy Novobiliski (grad school), Dr. Mina Sartipi (thesis advisor), Dr. Billy Harris (graduate algorithms), Dr. Joe Dumas (grad school), Ritchie Carroll (creator of the openPDC), Paul Trachian, Christophe Bisciglia and Mike Olson (for recruiting me to Cloudera), Malcom Ramey (for my first real programming job), The University of Tennessee at Chattanooga, and Lupi's Pizza (for feeding me through grad school).

Last, and especially not least, I'd like to thank my wife Leslie and my sons Ethan, Griffin, and Dane for their patience while I worked late, often, and sometimes on vacation.

Adam

I would like to thank my team at Skymind for all the work they piled on in assisting with review of the book and content as we continued to iterate on the book. I would especially like to thank Chris who tolerated my crazy idea of writing a book while attempting to do a startup.

DL4J started in 2013 with a chance meeting with Josh at MLConf and it has grown in to quite the project now used all over the world. DL4J has taken me all over the world and has really opened my world up to tons of new experiences.

Firstly, I would like to thank my coauthor Josh Patterson who did the lion's share of the book and deserves much of the credit. He put in nights and weekends to get the book out the door while I continued working on the codebase and continuing to adapt the content to new features through the years.

Echoing Josh, many of our team mates and contributors who joined early on such as Alex, Melanie, Vyacheslav "Raver" Kokorin, and later on folks like Dave helping us as an extra pair of eyes on the math due diligence.

Tim McGovern has been a great ear for some of my crazy ideas on content for O'Reilly and was also amazing in letting me name the book.

Table of Contents

Preface xiii

1. A Review of Machine Learning 1
The Learning Machines 1
How Can Machines Learn? 2
Biological Inspiration 4
What Is Deep Learning? 6
Going Down the Rabbit Hole 7
Framing the Questions 8
The Math Behind Machine Learning: Linear Algebra 8
Scalars 9
Vectors 9
Matrices 10
Tensors 10
Hyperplanes 10
Relevant Mathematical Operations 11
Converting Data Into Vectors 11
Solving Systems of Equations 13
The Math Behind Machine Learning: Statistics 15
Probability 16
Conditional Probabilities 18
Posterior Probability 19
Distributions 19
Samples Versus Population 22
Resampling Methods 22
Selection Bias 22
Likelihood 23
How Does Machine Learning Work? 23

Regression 23
Classification 25
Clustering 26
Underfitting and Overfitting 26
Optimization 27
Convex Optimization 29
Gradient Descent 30
Stochastic Gradient Descent 32
Quasi-Newton Optimization Methods 33
Generative Versus Discriminative Models 33
Logistic Regression 34
The Logistic Function 35
Understanding Logistic Regression Output 35
Evaluating Models 36
The Confusion Matrix 36
Building an Understanding of Machine Learning 40

2. Foundations of Neural Networks and Deep Learning..... 41
Neural Networks 41
The Biological Neuron 43
The Perceptron 45
Multilayer Feed-Forward Networks 50
Training Neural Networks 56
Backpropagation Learning 57
Activation Functions 65
Linear 66
Sigmoid 66
Tanh 67
Hard Tanh 68
Softmax 68
Rectified Linear 69
Loss Functions 71
Loss Function Notation 71
Loss Functions for Regression 72
Loss Functions for Classification 75
Loss Functions for Reconstruction 77
Hyperparameters 78
Learning Rate 78
Regularization 79
Momentum 79
Sparsity 80

3. Fundamentals of Deep Networks..... 81
Defining Deep Learning 81
What Is Deep Learning? 81
Organization of This Chapter 91
Common Architectural Principles of Deep Networks 92
Parameters 92
Layers 93
Activation Functions 93
Loss Functions 95
Optimization Algorithms 96
Hyperparameters 100
Summary 105
Building Blocks of Deep Networks 105
RBMs 106
Autoencoders 112
Variational Autoencoders 114

4. Major Architectures of Deep Networks..... 117
Unsupervised Pretrained Networks 118
Deep Belief Networks 118
Generative Adversarial Networks 121
Convolutional Neural Networks (CNNs) 125
Biological Inspiration 126
Intuition 126
CNN Architecture Overview 128
Input Layers 130
Convolutional Layers 130
Pooling Layers 140
Fully Connected Layers 140
Other Applications of CNNs 141
CNNs of Note 141
Summary 142
Recurrent Neural Networks 143
Modeling the Time Dimension 143
3D Volumetric Input 146
Why Not Markov Models? 148
General Recurrent Neural Network Architecture 149
LSTM Networks 150
Domain-Specific Applications and Blended Networks 159
Recursive Neural Networks 160
Network Architecture 160
Varieties of Recursive Neural Networks 161

Applications of Recursive Neural Networks 161
Summary and Discussion 162
Will Deep Learning Make Other Algorithms Obsolete? 162
Different Problems Have Different Best Methods 162
When Do I Need Deep Learning? 163

5. Building Deep Networks. 165
Matching Deep Networks to the Right Problem 165
Columnar Data and Multilayer Perceptrons 166
Images and Convolutional Neural Networks 166
Time-series Sequences and Recurrent Neural Networks 167
Using Hybrid Networks 169
The DL4J Suite of Tools 169
Vectorization and DataVec 170
Runtimes and ND4J 170
Basic Concepts of the DL4J API 172
Loading and Saving Models 172
Getting Input for the Model 173
Setting Up Model Architecture 173
Training and Evaluation 174
Modeling CSV Data with Multilayer Perceptron Networks 175
Setting Up Input Data 178
Determining Network Architecture 178
Training the Model 181
Evaluating the Model 181
Modeling Handwritten Images Using CNNs 182
Java Code Listing for the LeNet CNN 183
Loading and Vectorizing the Input Images 185
Network Architecture for LeNet in DL4J 186
Training the CNN 190
Modeling Sequence Data by Using Recurrent Neural Networks 191
Generating Shakespeare via LSTMs 191
Classifying Sensor Time-series Sequences Using LSTMs 200
Using Autoencoders for Anomaly Detection 207
Java Code Listing for Autoencoder Example 207
Setting Up Input Data 211
Autoencoder Network Architecture and Training 211
Evaluating the Model 213
Using Variational Autoencoders to Reconstruct MNIST Digits 214
Code Listing to Reconstruct MNIST Digits 214
Examining the VAE Model 217
Applications of Deep Learning in Natural Language Processing 221

此为试读,需要完整PDF请访问: www.ertongbook.com