contents

preface xvii
acknowledgments xix
about this book xx
about the author xxiii
about the cover illustration xxiv

1 *What is deep learning?* 1

1.1 Artificial intelligence, machine learning, and deep learning 2

Artificial intelligence 2 ▪ *Machine learning* 3 ▪ *Learning rules and representations from data* 4 ▪ *The "deep" in "deep learning"* 7 ▪ *Understanding how deep learning works, in three figures* 8 ▪ *What deep learning has achieved so far* 10 *Don't believe the short-term hype* 11 ▪ *The promise of AI* 12

1.2 Before deep learning: A brief history of machine learning 13

Probabilistic modeling 13 ▪ *Early neural networks* 14 *Kernel methods* 14 ▪ *Decision trees, random forests, and gradient boosting machines* 15 ▪ *Back to neural networks* 16 *What makes deep learning different* 17 ▪ *The modern machine learning landscape* 18

1.3 Why deep learning? Why now? 20

Hardware 20 ▪ Data 21 ▪ Algorithms 22 ▪ A new wave of investment 23 ▪ The democratization of deep learning 24 Will it last? 24

2 The mathematical building blocks of neural networks 26

2.1 A first look at a neural network 27

2.2 Data representations for neural networks 31

Scalars (rank-0 tensors) 31 ▪ Vectors (rank-1 tensors) 31 Matrices (rank-2 tensors) 32 ▪ Rank-3 and higher-rank tensors 32 ▪ Key attributes 32 ▪ Manipulating tensors in NumPy 34 ▪ The notion of data batches 35 ▪ Real-world examples of data tensors 35 ▪ Vector data 35 ▪ Timeseries data or sequence data 36 ▪ Image data 37 ▪ Video data 37

2.3 The gears of neural networks: Tensor operations 38

Element-wise operations 38 ▪ Broadcasting 40 ▪ Tensor product 41 Tensor reshaping 43 ▪ Geometric interpretation of tensor operations 44 A geometric interpretation of deep learning 47

2.4 The engine of neural networks: Gradient-based optimization 48

What's a derivative? 49 ▪ Derivative of a tensor operation: The gradient 51 ▪ Stochastic gradient descent 52 ▪ Chaining derivatives: The Backpropagation algorithm 55

2.5 Looking back at our first example 61

Reimplementing our first example from scratch in TensorFlow 63 Running one training step 64 ▪ The full training loop 65 Evaluating the model 66

3 Introduction to Keras and TensorFlow 68

3.1 What's TensorFlow? 69

3.2 What's Keras? 69

3.3 Keras and TensorFlow: A brief history 71

3.4 Setting up a deep learning workspace 71

Jupyter notebooks: The preferred way to run deep learning experiments 72 ▪ Using Colaboratory 73

3.5 First steps with TensorFlow 75

Constant tensors and variables 76 ▪ Tensor operations: Doing math in TensorFlow 78 ▪ A second look at the GradientTape API 78 ▪ An end-to-end example: A linear classifier in pure TensorFlow 79

3.6 Anatomy of a neural network: Understanding core Keras APIs 84

Layers: The building blocks of deep learning 84 ▪ From layers to models 87 ▪ The "compile" step: Configuring the learning process 88 ▪ Picking a loss function 90 ▪ Understanding the fit() method 91 ▪ Monitoring loss and metrics on validation data 91 ▪ Inference: Using a model after training 93

4 *Getting started with neural networks: Classification and regression 95*

4.1 Classifying movie reviews: A binary classification example 97

The IMDB dataset 97 ▪ Preparing the data 98 ▪ Building your model 99 ▪ Validating your approach 102 ▪ Using a trained model to generate predictions on new data 105 ▪ Further experiments 105 ▪ Wrapping up 106

4.2 Classifying newswires: A multiclass classification example 106

The Reuters dataset 106 ▪ Preparing the data 107 ▪ Building your model 108 ▪ Validating your approach 109 ▪ Generating predictions on new data 111 ▪ A different way to handle the labels and the loss 112 ▪ The importance of having sufficiently large intermediate layers 112 ▪ Further experiments 113 Wrapping up 113

4.3 Predicting house prices: A regression example 113

The Boston housing price dataset 114 ▪ Preparing the data 114 Building your model 115 ▪ Validating your approach using K-fold validation 115 ▪ Generating predictions on new data 119 ▪ Wrapping up 119

5 *Fundamentals of machine learning 121*

5.1 Generalization: The goal of machine learning 121

Underfitting and overfitting 122 ▪ The nature of generalization in deep learning 127

5.2 Evaluating machine learning models 133

Training, validation, and test sets 133 ▪ Beating a common-sense baseline 136 ▪ Things to keep in mind about model evaluation 137

5.3 Improving model fit 138

Tuning key gradient descent parameters 138 ▪ Leveraging better architecture priors 139 ▪ Increasing model capacity 140

5.4 Improving generalization 142

Dataset curation 142 ▪ Feature engineering 143 ▪ Using early stopping 144 ▪ Regularizing your model 145

6 The universal workflow of machine learning 153

6.1 Define the task 155

Frame the problem 155 ▪ Collect a dataset 156 ▪ Understand your data 160 ▪ Choose a measure of success 160

6.2 Develop a model 161

Prepare the data 161 ▪ Choose an evaluation protocol 162 Beat a baseline 163 ▪ Scale up: Develop a model that overfits 164 ▪ Regularize and tune your model 165

6.3 Deploy the model 165

Explain your work to stakeholders and set expectations 165 Ship an inference model 166 ▪ Monitor your model in the wild 169 ▪ Maintain your model 170

7 Working with Keras: A deep dive 172

7.1 A spectrum of workflows 173

7.2 Different ways to build Keras models 173

The Sequential model 174 ▪ The Functional API 176 Subclassing the Model class 182 ▪ Mixing and matching different components 184 ▪ Remember: Use the right tool for the job 185

7.3 Using built-in training and evaluation loops 185

Writing your own metrics 186 ▪ Using callbacks 187 Writing your own callbacks 189 ▪ Monitoring and visualization with TensorBoard 190

7.4 Writing your own training and evaluation loops 192

Training versus inference 194 ▪ Low-level usage of metrics 195 A complete training and evaluation loop 195 ▪ Make it fast with tf.function 197 ▪ Leveraging fit() with a custom training loop 198

8 Introduction to deep learning for computer vision 201

8.1 Introduction to convnets 202

The convolution operation 204 ▪ The max-pooling operation 209

8.2 Training a convnet from scratch on a small dataset 211

The relevance of deep learning for small-data problems 212 Downloading the data 212 ▪ Building the model 215 Data preprocessing 217 ▪ Using data augmentation 221

8.3 Leveraging a pretrained model 224

Feature extraction with a pretrained model 225 ▪ Fine-tuning a pretrained model 234

9 *Advanced deep learning for computer vision 238*

9.1 Three essential computer vision tasks 238

9.2 An image segmentation example 240

9.3 Modern convnet architecture patterns 248

Modularity, hierarchy, and reuse 249 ▪ Residual connections 251 Batch normalization 255 ▪ Depthwise separable convolutions 257 Putting it together: A mini Xception-like model 259

9.4 Interpreting what convnets learn 261

Visualizing intermediate activations 262 ▪ Visualizing convnet filters 268 ▪ Visualizing heatmaps of class activation 273

10 *Deep learning for timeseries 280*

10.1 Different kinds of timeseries tasks 280

10.2 A temperature-forecasting example 281

Preparing the data 285 ▪ A common-sense, non-machine learning baseline 288 ▪ Let's try a basic machine learning model 289 Let's try a 1D convolutional model 290 ▪ A first recurrent baseline 292

10.3 Understanding recurrent neural networks 293

A recurrent layer in Keras 296

10.4 Advanced use of recurrent neural networks 300

Using recurrent dropout to fight overfitting 300 ▪ Stacking recurrent layers 303 ▪ Using bidirectional RNNs 304 Going even further 307

11 *Deep learning for text 309*

11.1 Natural language processing: The bird's eye view 309

11.2 Preparing text data 311

Text standardization 312 ▪ Text splitting (tokenization) 313 Vocabulary indexing 314 ▪ Using the TextVectorization layer 316

11.3 Two approaches for representing groups of words: Sets and sequences 319

Preparing the IMDB movie reviews data 320 ▪ Processing words as a set: The bag-of-words approach 322 ▪ Processing words as a sequence: The sequence model approach 327

11.4 The Transformer architecture 336

Understanding self-attention 337 ▪ Multi-head attention 341 The Transformer encoder 342 ▪ When to use sequence models over bag-of-words models 349

11.5 Beyond text classification: Sequence-to-sequence learning 350

A machine translation example 351 ▪ Sequence-to-sequence learning with RNNs 354 ▪ Sequence-to-sequence learning with Transformer 358

12 *Generative deep learning 364*

12.1 Text generation 366

A brief history of generative deep learning for sequence generation 366 ▪ How do you generate sequence data? 367 The importance of the sampling strategy 368 ▪ Implementing text generation with Keras 369 ▪ A text-generation callback with variable-temperature sampling 372 ▪ Wrapping up 376

12.2 DeepDream 376

Implementing DeepDream in Keras 377 ▪ Wrapping up 383

12.3 Neural style transfer 383

The content loss 384 ▪ The style loss 384 ▪ Neural style transfer in Keras 385 ▪ Wrapping up 391

12.4 Generating images with variational autoencoders 391

Sampling from latent spaces of images 391 ▪ Concept vectors for image editing 393 ▪ Variational autoencoders 393 Implementing a VAE with Keras 396 ▪ Wrapping up 401

12.5 Introduction to generative adversarial networks 401

A schematic GAN implementation 402 ▪ A bag of tricks 403 ▪ Getting our hands on the CelebA dataset 404 The discriminator 405 ▪ The generator 407 ▪ The adversarial network 408 ▪ Wrapping up 410

13 *Best practices for the real world 412*

13.1 Getting the most out of your models 413

Hyperparameter optimization 413 ▪ Model ensembling 420

13.2 Scaling-up model training 421

Speeding up training on GPU with mixed precision 422 Multi-GPU training 425 ▪ TPU training 428

14 *Conclusions 431*

14.1 Key concepts in review 432

Various approaches to AI 432 ▪ What makes deep learning special within the field of machine learning 432 ▪ How to think about deep learning 433 ▪ Key enabling technologies 434 ▪ The universal machine learning workflow 435 ▪ Key network architectures 436 ▪ The space of possibilities 440

14.2 The limitations of deep learning 442

The risk of anthropomorphizing machine learning models 443 Automatons vs. intelligent agents 445 ▪ Local generalization vs. extreme generalization 446 ▪ The purpose of intelligence 448 Climbing the spectrum of generalization 449

14.3 Setting the course toward greater generality in AI 450

On the importance of setting the right objective: The shortcut rule 450 ▪ A new target 452

14.4 Implementing intelligence: The missing ingredients 454

Intelligence as sensitivity to abstract analogies 454 ▪ The two poles of abstraction 455 ▪ The missing half of the picture 458

14.5 The future of deep learning 459

Models as programs 460 ▪ Blending together deep learning and program synthesis 461 ▪ Lifelong learning and modular subroutine reuse 463 ▪ The long-term vision 465

14.6 Staying up to date in a fast-moving field 466

Practice on real-world problems using Kaggle 466 ▪ Read about the latest developments on arXiv 466 ▪ Explore the Keras ecosystem 467

14.7 Final words 467

index 469