

Contents

33

31

34

37

41

44

48

53

64

65

67

69

78

78

79

81

82

135

Foreword by Dr. Anima Anandkumar xxi

Foreword by Dr. Craig Clawson xxiii

Preface xxv

Acknowledgments li

About the Author liii

1 THE ROSENBLATT PERCEPTRON 1

Example of a Two-Input Perceptron 4

The Perceptron Learning Algorithm 7

Limitations of the Perceptron 15

Combining Multiple Perceptrons 17

Implementing Perceptrons with Linear Algebra 20

 Vector Notation 21

 Dot Product 23

 Extending the Vector to a 2D Matrix 24

 Matrix-Vector Multiplication 25

 Matrix-Matrix Multiplication 26

 Summary of Vector and Matrix Operations Used for Perceptrons 28

 Dot Product as a Matrix Multiplication 29

 Extending to Multidimensional Tensors 29

Different Activation Functions to Avoid Vanishing Gradient in Hidden Layers	136
Variations on Gradient Descent to Improve Learning	141
Experiment: Tweaking Network and Learning Parameters	143
Hyperparameter Tuning and Cross-Validation	146
Using a Validation Set to Avoid Overfitting	148
Cross-Validation to Improve Use of Training Data	149
Concluding Remarks on the Path Toward Deep Learning	150
6 FULLY CONNECTED NETWORKS APPLIED TO REGRESSION	153
Output Units	154
Logistic Unit for Binary Classification	155
Softmax Unit for Multiclass Classification	156
Linear Unit for Regression	159
The Boston Housing Dataset	160
Programming Example: Predicting House Prices with a DNN	161
Improving Generalization with Regularization	166
Experiment: Deeper and Regularized Models for House Price Prediction	169
Concluding Remarks on Output Units and Regression Problems	170
7 CONVOLUTIONAL NEURAL NETWORKS APPLIED TO IMAGE CLASSIFICATION	171
The CIFAR-10 Dataset	173
Characteristics and Building Blocks for Convolutional Layers	175
Combining Feature Maps into a Convolutional Layer	180
Combining Convolutional and Fully Connected Layers into a Network	181
Effects of Sparse Connections and Weight Sharing	185

Programming Example: Image Classification with a Convolutional Network . . .	190
Concluding Remarks on Convolutional Networks	201
8 DEEPER CNNs AND PRETRAINED MODELS	205
VGGNet	206
GoogLeNet	210
ResNet	215
Programming Example: Use a Pretrained ResNet Implementation	223
Transfer Learning	226
Backpropagation for CNN and Pooling	228
Data Augmentation as a Regularization Technique	229
Mistakes Made by CNNs	231
Reducing Parameters with Depthwise Separable Convolutions	232
Striking the Right Network Design Balance with EfficientNet	234
Concluding Remarks on Deeper CNNs	235
9 PREDICTING TIME SEQUENCES WITH RECURRENT NEURAL NETWORKS	237
Limitations of Feedforward Networks	241
Recurrent Neural Networks	242
Mathematical Representation of a Recurrent Layer	243
Combining Layers into an RNN	245
Alternative View of RNN and Unrolling in Time	246
Backpropagation Through Time	248
Programming Example: Forecasting Book Sales	250
Standardize Data and Create Training Examples	256
Creating a Simple RNN	258

Comparison with a Network Without Recurrence	262
Extending the Example to Multiple Input Variables	263
Dataset Considerations for RNNs	264
Concluding Remarks on RNNs	265
10 LONG SHORT-TERM MEMORY	267
Keeping Gradients Healthy	267
Introduction to LSTM	272
LSTM Activation Functions	277
Creating a Network of LSTM Cells	278
Alternative View of LSTM	280
Related Topics: Highway Networks and Skip Connections	282
Concluding Remarks on LSTM	282
11 TEXT AUTOCOMPLETION WITH LSTM AND BEAM SEARCH	285
Encoding Text	285
Longer-Term Prediction and Autoregressive Models	287
Beam Search	289
Programming Example: Using LSTM for Text Autocompletion	291
Bidirectional RNNs	298
Different Combinations of Input and Output Sequences	300
Concluding Remarks on Text Autocompletion with LSTM	302
12 NEURAL LANGUAGE MODELS AND WORD EMBEDDINGS	303
Introduction to Language Models and Their Use Cases	304
Examples of Different Language Models	307

n-Gram Model	307
Skip-Gram Model	309
Neural Language Model	309
Benefit of Word Embeddings and Insight into How They Work	313
Word Embeddings Created by Neural Language Models	315
Programming Example: Neural Language Model and Resulting Embeddings	319
King – Man + Woman = Queen	329
King – Man + Woman ! = Queen	331
Language Models, Word Embeddings, and Human Biases	332
Related Topic: Sentiment Analysis of Text	334
Bag-of-Words and Bag-of-N-Grams	334
Similarity Metrics	338
Combining BoW and DL	340
Concluding Remarks on Language Models and Word Embeddings	342
13 WORD EMBEDDINGS FROM word2vec AND GloVe	343
Using word2vec to Create Word Embeddings Without a Language Model	344
Reducing Computational Complexity Compared to a Language Model	344
Continuous Bag-of-Words Model	346
Continuous Skip-Gram Model	348
Optimized Continuous Skip-Gram Model to Further Reduce Computational Complexity	349
Additional Thoughts on word2vec	352
word2vec in Matrix Form	353
Wrapping Up word2vec	354
Concluding Remarks	502

Programming Example: Exploring Properties of GloVe Embeddings	356
Concluding Remarks on word2vec and GloVe	361
14 SEQUENCE-TO-SEQUENCE NETWORKS AND NATURAL LANGUAGE TRANSLATION	363
<hr/>	
Encoder-Decoder Model for Sequence-to-Sequence Learning	366
Introduction to the Keras Functional API	368
Programming Example: Neural Machine Translation	371
Experimental Results	387
Properties of the Intermediate Representation	389
Concluding Remarks on Language Translation	391
15 ATTENTION AND THE TRANSFORMER	393
<hr/>	
Rationale Behind Attention	394
Attention in Sequence-to-Sequence Networks	395
Computing the Alignment Vector	400
Mathematical Notation and Variations on the Alignment Vector	402
Attention in a Deeper Network	404
Additional Considerations	405
Alternatives to Recurrent Networks	406
Self-Attention	407
Multi-head Attention	410
The Transformer	411
Concluding Remarks on the Transformer	415

16 ONE-TO-MANY NETWORK FOR IMAGE CAPTIONING **417**

Extending the Image Captioning Network with Attention	420
Programming Example: Attention-Based Image Captioning	421
Concluding Remarks on Image Captioning	443

17 MEDLEY OF ADDITIONAL TOPICS **447**

Autoencoders	448
Use Cases for Autoencoders	449
Other Aspects of Autoencoders	451
Programming Example: Autoencoder for Outlier Detection	452
Multimodal Learning	459
Taxonomy of Multimodal Learning	459
Programming Example: Classification with Multimodal Input Data	465
Multitask Learning	469
Why to Implement Multitask Learning	470
How to Implement Multitask Learning	471
Other Aspects and Variations on the Basic Implementation	472
Programming Example: Multiclass Classification and Question Answering with a Single Network	473
Process for Tuning a Network	477
When to Collect More Training Data	481
Neural Architecture Search	482
Key Components of Neural Architecture Search	482
Programming Example: Searching for an Architecture for CIFAR-10 Classification	488
Implications of Neural Architecture Search	501
Concluding Remarks	502

18	SUMMARY AND NEXT STEPS	503
	Things You Should Know by Now	503
	Ethical AI and Data Ethics	505
	Problems to Look Out For	506
	Checklist of Questions	512
	Things You Do Not Yet Know	512
	Reinforcement Learning	513
	Variational Autoencoders and Generative Adversarial Networks	513
	Neural Style Transfer	515
	Recommender Systems	515
	Models for Spoken Language	516
	Next Steps	516
A	LINEAR REGRESSION AND LINEAR CLASSIFIERS	519
	Linear Regression as a Machine Learning Algorithm	519
	Univariate Linear Regression	520
	Multivariate Linear Regression	521
	Modeling Curvature with a Linear Function	522
	Computing Linear Regression Coefficients	523
	Classification with Logistic Regression	525
	Classifying XOR with a Linear Classifier	528
	Classification with Support Vector Machines	531
	Evaluation Metrics for a Binary Classifier	533

B OBJECT DETECTION AND SEGMENTATION **539**

Object Detection	540
R-CNN	542
Fast R-CNN	544
Faster R-CNN	546
Semantic Segmentation	549
Upsampling Techniques	550
Deconvolution Network	557
U-Net	558
Instance Segmentation with Mask R-CNN	559

C WORD EMBEDDINGS BEYOND word2vec AND GloVe **563**

Wordpieces	564
FastText	566
Character-Based Method	567
ELMo	572
Related Work	575

D GPT, BERT, AND RoBERTa **577**

GPT	578
BERT	582
Masked Language Model Task	582
Next-Sentence Prediction Task	583
BERT Input and Output Representations	584
Applying BERT to NLP Tasks	586
RoBERTa	586

Historical Work Leading Up to GPT and BERT	588
Other Models Based on the Transformer	590
E NEWTON-RAPHSON VERSUS GRADIENT DESCENT	593
Newton-Raphson Root-Finding Method	594
Newton-Raphson Applied to Optimization Problems	595
Relationship Between Newton-Raphson and Gradient Descent	597
F MATRIX IMPLEMENTATION OF DIGIT CLASSIFICATION NETWORK	599
Single Matrix	599
Mini-Batch Implementation	602
G RELATING CONVOLUTIONAL LAYERS TO MATHEMATICAL CONVOLUTION	607
H GATED RECURRENT UNITS	613
Alternative GRU Implementation	616
Network Based on the GRU	616
I SETTING UP A DEVELOPMENT ENVIRONMENT	621
Python	622
Programming Environment	623
Jupyter Notebook	623
Using an Integrated Development Environment	624
Programming Examples	624
Supporting Spreadsheet	625

Datasets	625
MNIST	625
Bookstore Sales Data from US Census Bureau	626
Frankenstein from Project Gutenberg	627
GloVe Word Embeddings	627
Anki Bilingual Sentence Pairs	627
COCO	627
Installing a DL Framework	628
System Installation	628
Virtual Environment Installation	629
GPU Acceleration	629
Docker Container	630
Using a Cloud Service	630
TensorFlow Specific Considerations	630
Key Differences Between PyTorch and TensorFlow	631
Need to Write Our Own Fit/Training Function	631
Explicit Moves of Data Between NumPy and PyTorch	633
Explicit Transfer of Data Between CPU and GPU	633
Explicitly Distinguishing Between Training and Inference	634
Sequential versus Functional API	634
Lack of Compile Function	635
Recurrent Layers and State Handling	635
Cross-Entropy Loss	635
View/Reshape	636

J CHEAT SHEETS

637

CONTENTS

647 *Works Cited* 647

667 *Index* 667

693 *Bookstore Sales Data from US Census Bureau* 693

694 *From Word Embeddings* 694

695 *AI-Bi-Directional Sentence Pairs* 695

696 *1000* 696

698 *Building a DL Framework* 698

698 *System Installation* 698

699 *Virtual Environment Installation* 699

699 *GPU Acceleration* 699

630 *Docker Container* 630

630 *Using a Cloud Service* 630

630 *Some Few Specific Considerations* 630

631 *How to Connect Between Python and TensorFlow* 631

631 *How to Write Our Own F1 Training Function* 631

633 *How to Move Data Between NumPy and Python* 633

633 *Explicit Transfer of Data Between CPU and GPU* 633

634 *Explicitly Distinguishing Between Training and Inference* 634

634 *Sequential versus Functional API* 634

635 *Use of Global Function* 635

635 *Placement Layers and State Handling* 635

635 *Cross-Entropy Loss* 635

636 *Layer Reshape* 636

637 *APPENDICES* 637