

# Inhaltsverzeichnis

<b>Teil I</b>	<b>Deep Learning vorgestellt</b>	<b>1</b>
<b>1</b>	<b>Biologisches und maschinelles Sehen</b>	<b>3</b>
1.1	Das biologische Sehen . . . . .	3
1.2	Maschinelles Sehen . . . . .	10
1.2.1	Das Neocognitron . . . . .	10
1.2.2	LeNet-5 . . . . .	11
1.2.3	Der traditionelle Machine-Learning-Ansatz . . . . .	14
1.2.4	ImageNet und die ILSVRC . . . . .	15
1.2.5	AlexNet . . . . .	17
1.3	TensorFlow Playground . . . . .	20
1.4	Quick, Draw! . . . . .	22
1.5	Zusammenfassung . . . . .	23
<b>2</b>	<b>Menschen- und Maschinensprache</b>	<b>25</b>
2.1	Deep Learning für Natural Language Processing . . . . .	26
2.1.1	Deep-Learning-Netze lernen Repräsentationen automatisch . . . . .	26
2.1.2	Natural Language Processing . . . . .	28
2.1.3	Eine kurze Geschichte des Deep Learning für NLP . . . . .	30
2.2	Repräsentationen von Sprache im Computer . . . . .	31
2.2.1	1-aus-n-Repräsentationen von Wörtern . . . . .	31
2.2.2	Wortvektoren . . . . .	32
2.2.3	Wortvektor-Arithmetik . . . . .	36
2.2.4	word2viz . . . . .	37
2.2.5	Lokalistische versus verteilte Repräsentationen . . . . .	39
2.3	Elemente der natürlichen menschlichen Sprache . . . . .	41
2.4	Google Duplex . . . . .	44
2.5	Zusammenfassung . . . . .	46

<b>3</b>	<b>Maschinenkunst</b>	<b>47</b>
3.1	Eine feuchtfröhliche Nacht .....	47
3.2	Berechnungen auf nachgemachten menschlichen Gesichtern .....	50
3.3	Stiltransfer: Fotos in einen Monet verwandeln (und umgekehrt) ....	53
3.4	Machen Sie Ihre eigenen Skizzen fotorealistisch .....	54
3.5	Fotorealistische Bilder aus Text erzeugen .....	55
3.6	Bildverarbeitung mittels Deep Learning .....	56
3.7	Zusammenfassung .....	58
<b>4</b>	<b>Spielende Maschinen</b>	<b>61</b>
4.1	Deep Learning, KI und andere Monster .....	61
4.1.1	Künstliche Intelligenz .....	61
4.1.2	Machine Learning .....	63
4.1.3	Representation Learning .....	63
4.1.4	Künstliche neuronale Netze .....	63
4.1.5	Deep Learning .....	64
4.1.6	Maschinelles Sehen .....	65
4.1.7	Natural Language Processing .....	66
4.2	Drei Arten von Machine-Learning-Problemen .....	66
4.2.1	Supervised Learning .....	67
4.2.2	Unsupervised Learning .....	67
4.2.3	Reinforcement Learning .....	68
4.3	Deep Reinforcement Learning .....	70
4.4	Videospiele .....	72
4.5	Brettspiele .....	73
4.5.1	AlphaGo .....	74
4.5.2	AlphaGo Zero .....	78
4.5.3	AlphaZero .....	81
4.6	Manipulation von Objekten .....	83
4.7	Populäre Umgebungen für das Deep-Reinforcement-Learning .....	85
4.7.1	OpenAI Gym .....	85
4.7.2	DeepMind Lab .....	86
4.7.3	UnityML-Agents .....	88
4.8	Drei Arten von KI .....	89
4.8.1	Artificial Narrow Intelligence .....	89
4.8.2	Artificial General Intelligence .....	89
4.8.3	Artificial Super Intelligence .....	89
4.8.4	Zusammenfassung .....	90

---

<b>Teil II</b>	<b>Die nötige Theorie</b>	<b>91</b>
<b>5</b>	<b>Der (Code-)Karren vor dem (Theorie-)Pferd</b>	<b>93</b>
5.1	Voraussetzungen	93
5.2	Installation	94
5.3	Ein flaches Netzwerk in Keras	94
5.3.1	Der MNIST-Datensatz handgeschriebener Ziffern	95
5.3.2	Ein schematisches Diagramm des Netzwerks	96
5.3.3	Die Daten laden	99
5.3.4	Die Daten umformatieren	101
5.3.5	Die Architektur eines neuronalen Netzes entwerfen	102
5.3.6	Trainieren eines Deep-Learning-Modells	103
5.4	Zusammenfassung	104
<b>6</b>	<b>Künstliche Neuronen, die Hotdogs erkennen</b>	<b>105</b>
6.1	Das Einmaleins der biologischen Neuroanatomie	105
6.2	Das Perzeptron	106
6.2.1	Der Hotdog/Nicht-Hotdog-Detektor	107
6.2.2	Die wichtigste Gleichung in diesem Buch	111
6.3	Moderne Neuronen und Aktivierungsfunktionen	112
6.3.1	Das Sigmoid-Neuron	113
6.3.2	Das Tanh-Neuron	115
6.3.3	ReLU: Rectified Linear Units	116
6.4	Ein Neuron auswählen	118
6.5	Zusammenfassung	119
	Schlüsselkonzepte	119
<b>7</b>	<b>Künstliche neuronale Netze</b>	<b>121</b>
7.1	Die Eingabeschicht	121
7.2	Vollständig verbundene Schichten	122
7.3	Ein vollständig verbundenes Netzwerk zum Erkennen von Hotdogs	124
7.3.1	Forwardpropagation durch die erste verborgene Schicht	125
7.3.2	Forwardpropagation durch nachfolgende Schichten	126
7.4	Die Softmax-Schicht eines Netzwerks zum Klassifizieren von Fastfood	129
7.5	Zurück zu unserem flachen Netzwerk	132
7.6	Zusammenfassung	134
	Schlüsselkonzepte	135

<b>8</b>	<b>Deep Networks trainieren</b>	<b>137</b>
8.1	Kostenfunktionen	137
8.1.1	Quadratische Kosten	138
8.1.2	Gesättigte Neuronen	139
8.1.3	Kreuzentropie-Kosten	140
8.2	Optimierung: Lernen, um die Kosten zu minimieren	143
8.2.1	Der Gradientenabstieg	143
8.2.2	Die Lernrate	146
8.2.3	Batch-Größe und stochastischer Gradientenabstieg	148
8.2.4	Dem lokalen Minimum entkommen	152
8.3	Backpropagation	155
8.4	Die Anzahl der verborgenen Schichten und der Neuronen anpassen	156
8.5	Ein mittleres Netz in Keras	158
8.6	Zusammenfassung	161
	Schlüsselkonzepte	162
<b>9</b>	<b>Deep Networks verbessern</b>	<b>163</b>
9.1	Die Initialisierung der Gewichte	163
9.1.1	Xavier-Glorot-Verteilungen	168
9.2	Instabile Gradienten	171
9.2.1	Verschwindende Gradienten	171
9.2.2	Explodierende Gradienten	172
9.2.3	Batch-Normalisierung	172
9.3	Modellgeneralisierung (Überanpassung vermeiden)	174
9.3.1	L1- und L2-Regularisierung	176
9.3.2	Dropout	177
9.3.3	Datenaugmentation	180
9.4	Intelligente Optimierer	181
9.4.1	Momentum	181
9.4.2	Nesterov-Momentum	182
9.4.3	AdaGrad	182
9.4.4	AdaDelta und RMSProp	183
9.4.5	Adam	183
9.5	Ein tiefes neuronales Netz in Keras	184
9.6	Regression	186
9.7	TensorBoard	189
9.8	Zusammenfassung	192
	Schlüsselkonzepte	193

---

<b>Teil III</b>	<b>Interaktive Anwendungen des Deep Learning</b>	<b>195</b>
<b>10</b>	<b>Maschinelles Sehen</b>	<b>197</b>
10.1	Convolutional Neural Networks	197
10.1.1	Die zweidimensionale Struktur der visuellen Bilddarstellung	198
10.1.2	Berechnungskomplexität	198
10.1.3	Konvolutionsschichten	199
10.1.4	Mehrere Filter	202
10.1.5	Ein Beispiel für Konvolutionsschichten	203
10.2	Hyperparameter von Konvolutionsfiltern	208
10.2.1	Kernel-Größe	208
10.2.2	Schrittlänge	209
10.2.3	Padding	209
10.3	Pooling-Schichten	210
10.4	LeNet-5 in Keras	212
10.5	AlexNet und VGGNet in Keras	218
10.6	Residualnetzwerke	221
10.6.1	Schwindende Gradienten: Das Grauen der tiefen CNN	221
10.6.2	Residualverbindungen	222
10.6.3	ResNet	225
10.7	Anwendungen des maschinellen Sehens	225
10.7.1	Objekterkennung	226
10.7.2	Bildsegmentierung	230
10.7.3	Transfer-Lernen	233
10.7.4	Capsule Networks	237
10.8	Zusammenfassung	238
	Schlüsselkonzepte	239
<b>11</b>	<b>Natural Language Processing</b>	<b>241</b>
11.1	Natürliche Sprachdaten vorverarbeiten	241
11.1.1	Tokenisierung	244
11.1.2	Alle Zeichen in Kleinbuchstaben umwandeln	247
11.1.3	Stoppwörter und Interpunktionszeichen entfernen	247
11.1.4	Stemming	248
11.1.5	N-Gramme verarbeiten	249
11.1.6	Vorverarbeitung des kompletten Textkorpus	251

11.2	Worteinbettungen mit word2vec erzeugen	254
11.2.1	Die prinzipielle Theorie hinter word2vec	254
11.2.2	Wortvektoren evaluieren	257
11.2.3	word2vec ausführen	258
11.2.4	Wortvektoren plotten	263
11.3	Der Bereich unter der ROC-Kurve	268
11.3.1	Die Wahrheitsmatrix	269
11.3.2	Die ROC-AUC-Metrik berechnen	270
11.4	Klassifikation natürlicher Sprache mit vertrauten Netzwerken	274
11.4.1	Die IMDb-Filmkritiken laden	274
11.4.2	Die IMDb-Daten untersuchen	278
11.4.3	Die Länge der Filmkritiken standardisieren	281
11.4.4	Vollständig verbundenes Netzwerk	281
11.4.5	Convolutional Networks	288
11.5	Netzwerke für die Verarbeitung sequenzieller Daten	293
11.5.1	Recurrent Neural Networks	294
11.5.2	Ein RNN in Keras implementieren	296
11.5.3	Long Short-Term Memory Units	299
11.5.4	Bidirektionale LSTMs	303
11.5.5	Gestapelte rekurrente Modelle	303
11.5.6	Seq2seq und Attention	305
11.5.7	Transfer-Lernen in NLP	307
11.6	Nichtsequenzielle Architekturen: Die funktionale API in Keras	308
11.7	Zusammenfassung	312
	Schlüsselkonzepte	313
<b>12</b>	<b>Generative Adversarial Networks</b>	<b>315</b>
12.1	Die grundlegende GAN-Theorie	315
12.2	Der Quick, Draw!-Datensatz	319
12.3	Das Diskriminator-Netzwerk	323
12.4	Das Generator-Netzwerk	326
12.5	Das Adversarial-Netzwerk	329
12.6	Das GAN-Training	331
12.7	Zusammenfassung	337
	Schlüsselkonzepte	338

---

<b>13</b>	<b>Deep Reinforcement Learning</b>	<b>341</b>
13.1	Die grundlegende Theorie des Reinforcement Learning . . . . .	341
13.1.1	Das Cart-Pole-Spiel . . . . .	342
13.1.2	Markow-Entscheidungsprozesse . . . . .	344
13.1.3	Die optimale Strategie . . . . .	347
13.2	Die grundlegende Theorie von Deep-Q-Learning-Netzwerken . . . .	349
13.2.1	Value-Funktionen . . . . .	350
13.2.2	Q-Value-Funktionen . . . . .	350
13.2.3	Einen optimalen Q-Value schätzen . . . . .	351
13.3	Einen DQN-Agenten definieren . . . . .	353
13.3.1	Initialisierungsparameter . . . . .	355
13.3.2	Das neuronale-Netze-Modell des Agenten bauen . . . . .	358
13.3.3	Sich an das Spiel erinnern . . . . .	359
13.3.4	Training über Memory Replay . . . . .	359
13.3.5	Eine Aktion auswählen . . . . .	361
13.3.6	Speichern und Laden der Modellparameter . . . . .	362
13.4	Mit einer OpenAI-Gym-Umgebung interagieren . . . . .	362
13.4.1	Hyperparameter-Optimierung mit SLM Lab . . . . .	366
13.5	Agenten jenseits von DQN . . . . .	369
13.5.1	Policy-Gradienten und der REINFORCE-Algorithmus . . .	370
13.5.2	Der Actor-Critic-Algorithmus . . . . .	371
13.6	Zusammenfassung . . . . .	372
	Schlüsselkonzepte . . . . .	373
<b>Teil IV</b>	<b>KI und Sie</b>	<b>375</b>
<b>14</b>	<b>Mit Ihren eigenen Deep-Learning-Projekten beginnen</b>	<b>377</b>
14.1	Ideen für Deep-Learning-Projekte . . . . .	377
14.1.1	Machine Vision und GANs . . . . .	377
14.1.2	Natural Language Processing . . . . .	380
14.1.3	Deep Reinforcement Learning . . . . .	381
14.1.4	Ein vorhandenes Machine-Learning-Projekt überführen . . . . .	381
14.2	Ressourcen für weitere Projekte . . . . .	382
14.2.1	Gesellschaftlich nützliche Projekte . . . . .	383
14.3	Der Modellierungsprozess einschließlich der Anpassung der Hyperparameter . . . . .	384
14.3.1	Automatisierung der Hyperparameter-Suche . . . . .	387

14.4	Deep-Learning-Bibliotheken .....	387
14.4.1	Keras und TensorFlow .....	388
14.4.2	PyTorch .....	390
14.4.3	MXNet, CNTK, Caffe und so weiter .....	391
14.5	Software 2.0 .....	391
14.6	Die kommende Artificial General Intelligence .....	394
14.7	Zusammenfassung .....	397

---

<b>Anhang</b>	<b>399</b>
---------------	------------

---

<b>A</b>	<b>Die formale Notation neuronaler Netze</b>	<b>401</b>
<b>B</b>	<b>Backpropagation</b>	<b>403</b>
<b>C</b>	<b>PyTorch</b>	<b>407</b>
C.1	PyTorch-Eigenschaften .....	407
C.1.1	Das Autograd System .....	407
C.1.2	Das Define-by-Run-Framework .....	407
C.1.3	PyTorch im Vergleich mit TensorFlow .....	408
C.2	PyTorch in der Praxis .....	409
C.2.1	Die PyTorch-Installation .....	409
C.2.2	Die grundlegenden Bausteine in PyTorch .....	410
C.2.3	Ein tiefes neuronales Netz in PyTorch bauen .....	411
<b>D</b>	<b>Bildnachweise</b>	<b>415</b>
	<b>Abbildungsverzeichnis</b>	<b>417</b>
	<b>Tabellenverzeichnis</b>	<b>429</b>
	<b>Beispielverzeichnis</b>	<b>431</b>
	<b>Index</b>	<b>435</b>