

## Auf einen Blick

1	Einführung .....	15
2	Machine Learning und Deep Learning .....	33
3	Neuronale Netze .....	67
4	Python und Machine-Learning-Bibliotheken .....	107
5	TensorFlow: Installation und Grundlagen .....	139
6	Keras .....	185
7	Netze und Metriken visualisieren .....	227
8	TensorFlow.js .....	285
9	Praxisbeispiele .....	361
10	Ausblick .....	463
11	Fazit .....	489

# Inhalt

<b>1</b>	<b>Einführung</b>	15
<b>1.1</b>	<b>Über dieses Buch</b>	15
1.1.1	Voraussetzungen	16
1.1.2	Aufbau des Buches	16
1.1.3	GitHub-Repository zum Buch	17
<b>1.2</b>	<b>Ein Einblick in Deep Learning</b>	17
<b>1.3</b>	<b>Deep Learning im Alltag und in der Zukunft</b>	19
1.3.1	Sprachverarbeitung	20
1.3.2	Automatische Sprachübersetzungen	21
1.3.3	Visual Computing	22
1.3.4	Autonomes Fahren	24
1.3.5	Deep Learning in den sozialen Netzwerken	25
1.3.6	Deep Learning in der Medizin	25
1.3.7	Deep Learning in der Industrie 4.0	27
1.3.8	Deep Learning in der Landwirtschaft	27
1.3.9	Deep Learning im Rechtswesen	28
1.3.10	Deep Learning in der Finanzwelt	29
1.3.11	Deep Learning in der Energiewirtschaft	31
<b>2</b>	<b>Machine Learning und Deep Learning</b>	33
<b>2.1</b>	<b>Einführung</b>	33
<b>2.2</b>	<b>Lernansätze beim Machine Learning</b>	38
2.2.1	Supervised Learning	39
2.2.2	Unsupervised Learning	41
2.2.3	Semi-supervised Learning	42
2.2.4	Reinforcement Learning	42
2.2.5	Aktives Lernen	44
<b>2.3</b>	<b>Deep-Learning-Frameworks</b>	44
<b>2.4</b>	<b>Datenbeschaffung</b>	46
2.4.1	Vorgefertigte Datasets	46
2.4.2	Eigene Datasets	47

<b>2.5 Datasets</b>	48
2.5.1 Kaggle	48
2.5.2 Google Dataset Search	49
2.5.3 Chars74K	50
2.5.4 ImageNET	50
2.5.5 ImageClef	53
2.5.6 VGG	54
2.5.7 YFCC100M	56
2.5.8 YouTube-8M	56
2.5.9 MS-Celeb-1M	57
2.5.10 CelebA	58
2.5.11 VoxCeleb2	59
2.5.12 Microsoft COCO	59
2.5.13 MNIST und Fashion-MNIST	60
2.5.14 UCI-Datasets	62
2.5.15 Uber-Datasets	62
2.5.16 CLEVR-Dataset	63
2.5.17 Weitere Datasets	64
2.5.18 Checkliste zu Datasets	64
<b>2.6 Zusammenfassung</b>	65
<b>3 Neuronale Netze</b>	67
<hr/>	
<b>3.1 Aufbau und Prinzip</b>	67
3.1.1 Das künstliche Neuron	69
3.1.2 Wie funktioniert ein künstliches Neuron?	69
3.1.3 Mehrschichtige neuronale Netze	71
<b>3.2 Lernprozess neuronaler Netze</b>	73
3.2.1 Lernvorgang	73
3.2.2 Backpropagation und Fehlerfunktionen	76
3.2.3 Gewichtsanzpassung und Training	77
3.2.4 Gradientenverfahren	79
<b>3.3 Datenaufbereitung</b>	81
<b>3.4 Ein einfaches neuronales Netz</b>	82
<b>3.5 Netzarchitekturen</b>	91
3.5.1 Convolutional Neural Networks	91

3.5.2 Rekurrente neuronale Netze	96
3.5.3 Generative Adversarial Networks	97
<b>3.6 Bekannte Netze</b>	98
3.6.1 LeNet	98
3.6.2 AlexNet	98
3.6.3 VGGNet	99
3.6.4 YOLO	100
3.6.5 PoseNet	100
<b>3.7 Fallstricke beim Deep Learning</b>	101
<b>3.8 Zusammenfassung</b>	106
<b>4 Python und Machine-Learning-Bibliotheken</b>	107
<hr/>	
<b>4.1 Installation von Python 3.7 mit Anaconda</b>	108
4.1.1 Installation unter Windows	109
4.1.2 Installation unter macOS	109
4.1.3 Installation unter Ubuntu Linux	109
4.1.4 Anlegen von Umgebungen	110
4.1.5 Organisation der Arbeitsumgebung zum Buch	111
<b>4.2 Alternative Installationen von Python 3.7</b>	113
4.2.1 Unter Microsoft Windows	113
4.2.2 Unter macOS	114
<b>4.3 Programmierumgebungen</b>	116
4.3.1 Microsoft Visual Studio Code	116
4.3.2 PyCharm CE	118
4.3.3 Spyder	120
<b>4.4 Jupyter Notebook</b>	121
<b>4.5 Python-Bibliotheken für das Machine Learning</b>	126
4.5.1 NumPy	126
4.5.2 Pandas	127
4.5.3 SciPy	127
4.5.4 Scikit-learn	128
<b>4.6 Nützliche Routinen mit NumPy und Scikit-learn für ML</b>	129
4.6.1 Ein Dataset aus einer CSV- oder JSON-Datei laden	129
4.6.2 Ein Dataset aus dem Web laden	130
4.6.3 Ein in Scikit-learn inkludiertes Dataset laden	130

4.6.4	Daten explorieren und visualisieren .....	130
4.6.5	Normalisierung von Daten .....	131
4.6.6	Reproduzierbarkeit von Ergebnissen .....	132
<b>4.7</b>	<b>Ein zweites Machine-Learning-Beispiel .....</b>	<b>133</b>
<b>4.8</b>	<b>Zusammenfassung .....</b>	<b>137</b>
<b>5</b>	<b>TensorFlow: Installation und Grundlagen .....</b>	<b>139</b>
<b>5.1</b>	<b>Einführung .....</b>	<b>139</b>
<b>5.2</b>	<b>Installation .....</b>	<b>143</b>
5.2.1	Installation unter Windows mit alleiniger CPU-Unterstützung .....	144
5.2.2	Installation unter macOS .....	144
5.2.3	Installation unter Linux .....	145
5.2.4	TensorFlow mit GPU-Unterstützung unter Windows .....	145
5.2.5	TensorFlow mit GPU-Unterstützung unter Linux .....	150
5.2.6	TensorFlow mit GPU-Unterstützung unter macOS .....	151
5.2.7	Überprüfung der TensorFlow-Installation .....	152
<b>5.3</b>	<b>Google Colab: TensorFlow ohne Installation benutzen .....</b>	<b>153</b>
<b>5.4</b>	<b>Tensoren .....</b>	<b>154</b>
5.4.1	Variablen .....	158
5.4.2	Operationen .....	159
<b>5.5</b>	<b>Graphen .....</b>	<b>160</b>
5.5.1	Konzept .....	161
5.5.2	Eager Execution .....	163
5.5.3	AutoGraph und der @tf.function-Dekorator .....	163
5.5.4	Graphen visualisieren .....	166
<b>5.6</b>	<b>Benutzung der CPU und GPU .....</b>	<b>171</b>
<b>5.7</b>	<b>Erstes Beispiel: Eine lineare Regression .....</b>	<b>174</b>
5.7.1	Schritt 1: Datengenerierung und Visualisierung .....	175
5.7.2	Schritt 2: Modell erstellen .....	176
5.7.3	Schritt 3: Modell trainieren und visualisieren .....	177
<b>5.8</b>	<b>Von TensorFlow 1.x zu TensorFlow 2 .....</b>	<b>180</b>
5.8.1	Was hat sich verändert? .....	180
5.8.2	Automatische Konvertierung mit tf_upgrade_v2 .....	182
<b>5.9</b>	<b>Zusammenfassung .....</b>	<b>183</b>

<b>6</b>	<b>Keras .....</b>	<b>185</b>
<b>6.1</b>	<b>Von Keras zu tf.keras .....</b>	<b>185</b>
6.1.1	K wie Keras oder Konfusion .....	186
6.1.2	Die Zukunft von Keras .....	188
<b>6.2</b>	<b>Erster Kontakt .....</b>	<b>189</b>
6.2.1	Sequential Model .....	189
6.2.2	Functional API .....	190
6.2.3	Keras-Layers .....	191
<b>6.3</b>	<b>Modelle trainieren .....</b>	<b>191</b>
<b>6.4</b>	<b>Modelle evaluieren .....</b>	<b>193</b>
<b>6.5</b>	<b>Modelle laden und exportieren .....</b>	<b>194</b>
6.5.1	Speichern des Modells als h5-Datei .....	194
6.5.2	Speichern als SavedModel-Format .....	195
6.5.3	Separates Speichern der Struktur und Parameter des Modells .....	196
<b>6.6</b>	<b>Keras Applications .....</b>	<b>197</b>
<b>6.7</b>	<b>Keras Callbacks .....</b>	<b>198</b>
<b>6.8</b>	<b>Projekt 1: Iris-Klassifikation mit Keras .....</b>	<b>200</b>
6.8.1	Dataset laden .....	200
6.8.2	Modell erstellen .....	202
6.8.3	Modell trainieren .....	202
6.8.4	Modell evaluieren .....	202
6.8.5	Modell benutzen .....	203
<b>6.9</b>	<b>Projekt 2: CNNs mit Fashion-MNIST .....</b>	<b>204</b>
6.9.1	Schritt 1: Laden der Daten .....	205
6.9.2	Schritt 2: Modell erstellen .....	209
6.9.3	Schritt 3: Modell trainieren .....	211
6.9.4	Schritt 4: Modell evaluieren .....	212
6.9.5	Schritt 5: Modell exportieren und benutzen .....	213
<b>6.10</b>	<b>Projekt 3: Ein einfaches CNN mit dem CIFAR-10-Dataset .....</b>	<b>213</b>
6.10.1	CIFAR-10-Dataset laden .....	214
6.10.2	Modell erstellen und trainieren .....	215
6.10.3	Modell als h5-Datei exportieren .....	218
6.10.4	Modell als SavedModel-Format exportieren .....	218
6.10.5	Modell benutzen .....	219
<b>6.11</b>	<b>Projekt 4: Aktienkursvorhersage mit RNNs und LSTMs .....</b>	<b>220</b>
6.11.1	Vorbereitung der Daten .....	221

6.11.2	Modell erstellen und trainieren .....	224
6.11.3	Modell testen .....	224
<b>6.12</b>	<b>Zusammenfassung .....</b>	<b>226</b>
<b>7</b>	<b>Netze und Metriken visualisieren .....</b>	<b>227</b>
<b>7.1</b>	<b>TensorBoard .....</b>	<b>228</b>
7.1.1	Graphen visualisieren .....	229
7.1.2	Metriken und Skalare visualisieren .....	232
7.1.3	Histogram und Distributions Dashboard .....	235
7.1.4	Text-Dashboard .....	237
7.1.5	Images-Dashboard .....	240
7.1.6	Integration von TensorBoard in Jupyter Notebook .....	242
7.1.7	Weitere TensorBoard-Dashboards .....	243
<b>7.2</b>	<b>TensorBoard.dev .....</b>	<b>246</b>
<b>7.3</b>	<b>Debugging mit TensorBoard (nur TF 1.x) .....</b>	<b>248</b>
7.3.1	Debbing eines einfachen Modells .....	248
7.3.2	Debugging eines CNNs .....	253
<b>7.4</b>	<b>Der TensorBoard-Debugger mit Keras (nur TF 1.x) .....</b>	<b>257</b>
<b>7.5</b>	<b>Visualisierung mit Keras .....</b>	<b>259</b>
7.5.1	Die Struktur eines Modells mit plot_model() visualisieren .....	259
7.5.2	Aktivierungen visualisieren .....	260
7.5.3	tf-explain .....	264
7.5.4	Keras-Metriken mit Bokeh darstellen .....	265
<b>7.6</b>	<b>Visualisierung von CNNs mit Quiver (nur mit Keras.io) .....</b>	<b>267</b>
7.6.1	Installation .....	268
7.6.2	Anpassungen für Keras.io und TensorFlow 2 .....	268
7.6.3	Start .....	269
<b>7.7</b>	<b>Interaktive Visualisierung mit Keras-Callbacks, Node.js und HTML5 selbst implementieren .....</b>	<b>271</b>
7.7.1	Projektarchitektur .....	271
7.7.2	Benutzung von Keras-Callbacks .....	272
7.7.3	Server .....	274
7.7.4	Aufbau der Visualisierung .....	277
7.7.5	Start der Visualisierung .....	279
<b>7.8</b>	<b>Weitere Visualisierungsmöglichkeiten .....</b>	<b>281</b>
7.8.1	Netron .....	282

7.8.2	Net2Vis .....	282
7.8.3	ConX .....	283
<b>8</b>	<b>TensorFlow.js .....</b>	<b>285</b>
<b>8.1</b>	<b>Anwendungsfälle .....</b>	<b>285</b>
<b>8.2</b>	<b>Installation von BrowserSync .....</b>	<b>288</b>
<b>8.3</b>	<b>Installation von TensorFlow.js .....</b>	<b>290</b>
<b>8.4</b>	<b>Konzepte .....</b>	<b>293</b>
8.4.1	Keras vs. TensorFlow.js .....	293
8.4.2	Variablen .....	296
8.4.3	Tensoren .....	297
8.4.4	Operationen .....	301
8.4.5	Modelle und Schichten .....	302
8.4.6	Bilddaten mit tf.fromPixels() einlesen .....	303
<b>8.5</b>	<b>Ihr erstes Modell mit TensorFlow.js: Eine quadratische Regression .....</b>	<b>304</b>
8.5.1	Schritt 1: index.html-Datei anlegen .....	305
8.5.2	Schritt 2: Daten anlegen .....	309
8.5.3	Schritt 3: Modell erstellen .....	309
8.5.4	Schritt 4: Modell trainieren .....	310
8.5.5	Trainingsmetriken mit tfjs-vis visualisieren .....	312
<b>8.6</b>	<b>Laden und Speichern von Modellen .....</b>	<b>318</b>
8.6.1	Modelle mit model.save() speichern .....	318
8.6.2	Modell laden .....	324
8.6.3	Keras-Modelle exportieren und in TensorFlow.js importieren .....	326
<b>8.7</b>	<b>PoseNet-Modell mit TensorFlow.js .....</b>	<b>327</b>
8.7.1	Schritt 1: Implementierung der Benutzerschnittstelle .....	329
8.7.2	Schritt 2: Laden des PoseNet-Modells .....	334
8.7.3	Schritt 3: Single- und Multi-Pose-Erkennung .....	334
8.7.4	Schritt 4: Ergebnisse von PoseNet visualisieren .....	336
<b>8.8</b>	<b>Eine intelligente Smart-Home-Komponente mit TensorFlow.js und Node.js .....</b>	<b>343</b>
8.8.1	Schritt 1: Das TensorFlow.js-Projekt für Node.js vorbereiten .....	343
8.8.2	Schritt 2: Analyse und Vorbereitung der Eingabedaten .....	344
8.8.3	Schritt 3: Modell trainieren .....	349
8.8.4	Schritt 4: Modell speichern .....	351
8.8.5	Schritt 5: Modell testen .....	351

8.8.6	Schritt 6: Modell als Webservice benutzen .....	353
8.8.7	Erweiterungsmöglichkeiten .....	355
<b>8.9</b>	<b>TensorFlow.js noch einfacher: ml5.js .....</b>	<b>356</b>
<b>9</b>	<b>Praxisbeispiele .....</b>	<b>361</b>
<b>9.1</b>	<b>Projekt 1: Verkehrszeichenerkennung mit Keras .....</b>	<b>363</b>
9.1.1	Schritt 1: Datasets .....	364
9.1.2	Schritt 2: Datenanalyse und Vorbereitung der Klassifikation .....	366
9.1.3	Schritt 3: Modell trainieren .....	368
9.1.4	Schritt 4: Modell testen .....	374
9.1.5	Erweiterungsmöglichkeiten .....	378
<b>9.2</b>	<b>Projekt 2: Intelligente Spurerkennung mit Keras und OpenCV .....</b>	<b>379</b>
9.2.1	Schritt 1: Methodik zur Datenerstellung .....	380
9.2.2	Schritt 2: Benutzung der Modelle .....	385
<b>9.3</b>	<b>Projekt 3: Erkennung der Umgebung mit YOLO und TensorFlow.js bzw. ml5.js .....</b>	<b>392</b>
9.3.1	Das YOLO-Modell .....	392
9.3.2	Schritt 1: Vorbereitung des Projekts .....	395
9.3.3	Schritt 2: Aufbereitung der Objekterkennung .....	396
9.3.4	Schritt 3: Benutzung des YOLO-Modells .....	401
9.3.5	Schritt 4: Erweiterungsmöglichkeiten .....	402
<b>9.4</b>	<b>Projekt 4: Haus oder Katze? Vorgefertigte Modelle mit Keras benutzen – VGG-19 .....</b>	<b>403</b>
9.4.1	Schritt 1: Vorbereitung .....	404
9.4.2	Schritt 2: Instanziierung des VGG-19-Modells .....	405
9.4.3	Schritt 3: Benutzung des Modells .....	405
9.4.4	Schritt 4: Benutzung weiterer Modelle und Tests .....	406
<b>9.5</b>	<b>Projekt 5: Buchstaben- und Ziffernerkennung mit dem Chars74K-Dataset und Datenaugmentierung .....</b>	<b>410</b>
9.5.1	Schritt 1: Datenanalyse .....	410
9.5.2	Schritt 2: Datenaugmentierung mit OpenCV .....	411
9.5.3	Schritt 3: Zwei Modelle erstellen und trainieren .....	412
9.5.4	Schritt 4: Modell evaluieren .....	414
9.5.5	Schritt 5: Datenaugmentierung mit ImageDataGenerator .....	416
<b>9.6</b>	<b>Projekt 6: Stimmungsanalyse mit Keras .....</b>	<b>418</b>
9.6.1	Schritt 1: Eingabedaten und Datenstruktur .....	419

9.6.2	Schritt 2: Modell(e) erstellen .....	422
9.6.3	Schritt 3: Modell trainieren .....	423
9.6.4	Schritt 4: Modell benutzen .....	424
<b>9.7</b>	<b>Projekt 7: Sentiment-Analyse mit TensorFlow.js .....</b>	<b>425</b>
9.7.1	Schritt 1: Die Benutzerschnittstelle erstellen und das Modell laden .....	426
9.7.2	Schritt 2: Modell benutzen .....	429
<b>9.8</b>	<b>Projekt 8: Benutzung von TensorFlow Hub .....</b>	<b>429</b>
9.8.1	iMet Collection Attribute Classifier .....	430
9.8.2	Stiltransfer .....	433
<b>9.9</b>	<b>Projekt 9: Hyperparameter-Tuning mit TensorBoard .....</b>	<b>437</b>
9.9.1	Table View .....	439
9.9.2	Parallel Coordinates View .....	440
9.9.3	Scatter Plot Matrix View .....	441
<b>9.10</b>	<b>Projekt 10: CNN mit Fashion-MNIST und TensorFlow-Estimators (nur TF 1.x) .....</b>	<b>442</b>
9.10.1	Schritt 1: Modell mit tf.estimator erstellen .....	443
9.10.2	Schritt 2: Modell anlegen .....	444
9.10.3	Schritt 3: Modell trainieren und evaluieren .....	448
9.10.4	Schritt 4: Vorhersage mit predict() .....	450
9.10.5	Schritt 5: Benutzung des Modells .....	451
9.10.6	Bonus: Alternativmodell mit einem DNNClassifier .....	453
9.10.7	Erweiterung .....	455
<b>9.11</b>	<b>Allgemeine Tipps und Tricks .....</b>	<b>455</b>
9.11.1	Datenanalyse .....	455
9.11.2	Das Problem und die Aufgaben eines Modells korrekt definieren .....	456
9.11.3	Wie viele verdeckte Schichten und Neuronen brauche ich? .....	457
9.11.4	Die Falle des Overfittings bzw. Underfittings vermeiden .....	458
9.11.5	Datenaugmentierung und synthetische Daten .....	459
9.11.6	Normalisierung der Daten .....	459
9.11.7	Eingabedimensionen von Tensoren .....	460
9.11.8	Hilfe, es geht nicht! .....	461
<b>10</b>	<b>Ausblick .....</b>	<b>463</b>
<b>10.1</b>	<b>Deep Learning in der Cloud .....</b>	<b>463</b>
10.1.1	Google Cloud Machine Learning .....	463
10.1.2	Microsoft Azure .....	465

10.1.3	Amazon Deep Learning-AMIs .....	466
10.1.4	IBM Watson Services .....	468
10.1.5	Sony Neural Network Console .....	469
10.1.6	Adobe Sensei .....	469
10.1.7	Cloud oder nicht Cloud? .....	470
<b>10.2</b>	<b>Bildgenerierung mit Deep Learning</b> .....	<b>471</b>
10.2.1	Google Deep Dream .....	471
10.2.2	pix2pix .....	472
10.2.3	Microsoft Sketch2Code .....	474
10.2.4	Stiltransfer mit GANs .....	474
<b>10.3</b>	<b>Musik mit Deep Learning</b> .....	<b>476</b>
10.3.1	FlowMachine .....	476
10.3.2	Magenta.js .....	476
<b>10.4</b>	<b>Videogenerierung mit Deep Learning</b> .....	<b>478</b>
<b>10.5</b>	<b>Deep Learning einfacher gemacht</b> .....	<b>480</b>
10.5.1	Automatisiertes Machine Learning .....	481
10.5.2	Uber Ludwig .....	483
10.5.3	Teachable Machine .....	485
10.5.4	RunwayML .....	487
<b>11</b>	<b>Fazit</b> .....	<b>489</b>
	<hr/>	
	Index .....	492