

Vorwort	IX
1 Werkzeuge und Techniken	1
1.1 Arten neuronaler Netze	1
1.2 Datenbeschaffung	12
1.3 Vorverarbeitung von Daten	19
2 Fehlerbehebung	27
2.1 Probleme bemerken	27
2.2 Laufzeitfehler beheben	28
2.3 Zwischenergebnisse überprüfen	31
2.4 Wählen der richtigen Aktivierungsfunktion (für die letzte Schicht)	32
2.5 Regularisierung und Drop-out	33
2.6 Netzwerkstruktur, Batch-Größe und Lernrate	35
3 Die Ähnlichkeit von Texten mithilfe von Worteinbettungen berechnen	37
3.1 Wortähnlichkeiten mithilfe vortrainierter Worteinbettungen finden	38
3.2 Word2vec-Mathematik	40
3.3 Worteinbettungen visualisieren	42
3.4 Objektklassen in Einbettungen finden	44
3.5 Semantische Abstände innerhalb einer Klasse berechnen	47
3.6 Länderdaten auf einer Landkarte visualisieren	49
4 Ein Empfehlungssystem anhand ausgehender Wikipedia-Links erstellen	51
4.1 Sammeln der Daten	51
4.2 Trainieren von Filmeinbettungen	55
4.3 Ein Filmempfehlungssystem erstellen	58
4.4 Vorhersagen einfacher Filmmerkmale	59

5	Text im Stil eines Beispieltexts generieren	63
5.1	Den Text von gemeinfreien Büchern beschaffen	63
5.2	Texte im Stil von Shakespeare generieren	64
5.3	Code mit RNNs erzeugen	68
5.4	Steuerung der Temperatur des Outputs	70
5.5	Visualisierung der Aktivierungen eines rekurrenten Netzwerks	72
6	Übereinstimmende Fragen	75
6.1	Daten aus Stack Exchange beschaffen	75
6.2	Erkundung der Daten mit Pandas	77
6.3	Textkodierung in Keras	78
6.4	Ein Frage-Antwort-Modell entwickeln	79
6.5	Training eines Modells mit Pandas	81
6.6	Überprüfung von Gemeinsamkeiten	82
7	Emojis vorschlagen	85
7.1	Einen einfachen Stimmungsklassifikator entwickeln	85
7.2	Inspizieren eines einfachen Klassifikators	88
7.3	Ein Konvolutionsnetz zur Stimmungsanalyse verwenden	89
7.4	Twitter-Daten sammeln	92
7.5	Ein simples Vorhersagemodell für Emojis	93
7.6	Drop-out und variierende Fenstergröße	95
7.7	Ein wortbasiertes Modell erstellen	96
7.8	Eigene Einbettungen erzeugen	98
7.9	Ein rekurrentes neuronales Netzwerk zur Klassifikation verwenden	100
7.10	Übereinstimmung visualisieren	102
7.11	Modelle miteinander kombinieren	104
8	Sequenz-zu-Sequenz-Mapping	107
8.1	Trainieren eines einfachen Sequenz-zu-Sequenz-Modells	107
8.2	Dialoge aus Texten extrahieren	109
8.3	Einen frei verfügbaren Wortschatz handhaben	111
8.4	Einen seq2seq-Chatbot trainieren	112
9	Ein vortrainiertes Netzwerk zur Bilderkennung verwenden	117
9.1	Ein vortrainiertes Netzwerk laden	117
9.2	Vorverarbeitung der Bilder	118
9.3	Vorhersagen des Bildinhalts (Inferenz)	120
9.4	Einen gelabelten Bilddatensatz mit der Flickr-API sammeln	121
9.5	Einen Hund-Katze-Klassifikator erstellen	122

9.6	Suchergebnisse verbessern	124
9.7	Trainieren vortrainierter Netzwerke zur Bilderkennung	126
10	Eine Reverse-Image-Suchmaschine erstellen	129
10.1	Zugriff auf Bilder von Wikipedia	129
10.2	Projektion von Bildern in einen N-dimensionalen Raum.	132
10.3	Nächste Nachbarn in hochdimensionalen Räumen finden	133
10.4	Lokale Nachbarschaften in Einbettungen erkunden	134
11	Mehrere Bildinhalte erkennen	137
11.1	Erkennen mehrerer Bildinhalte mithilfe eines vortrainierten Klassifikators	137
11.2	Bilderkennung mithilfe eines Faster RCNN.	141
11.3	Eigene Bilder in einem Faster RCNN verwenden.	144
12	Mit Bildstilen arbeiten	147
12.1	Aktivierungen eines CNN visualisieren	148
12.2	Oktaven und Vergrößerung	151
12.3	Veranschaulichen, was ein neuronales Netzwerk in etwa wahrnimmt.	153
12.4	Den Stil eines Bilds erfassen	156
12.5	Verbessern der Verlustfunktion zur Erhöhung der Bildkohärenz	159
12.6	Einen Stil auf ein anderes Bild übertragen	161
12.7	Stilinterpolation	162
13	Bilder mit Autoencodern erzeugen	165
13.1	Zeichnungen aus Google Quick Draw importieren	166
13.2	Einen Autoencoder für Bilder erstellen	167
13.3	Visualisierung der Ergebnisse von Autoencodern	170
13.4	Sampling von Bildern aus einer korrekten Verteilung	171
13.5	Den latenten Raum eines Variational Autoencoders visualisieren	175
13.6	Conditional Variational Autoencoder	176
14	Piktogramme mithilfe von neuronalen Netzwerken erzeugen	181
14.1	Piktogramme zum Trainieren beschaffen.	182
14.2	Piktogramme in eine Tensor-Darstellung umwandeln.	184
14.3	Piktogramme mithilfe eines Variational Autoencoders erzeugen	185
14.4	Datenanreicherung zur Verbesserung der Leistung des Autoencoders	188

14.5	Ein Generative Adversarial Network aufbauen	189
14.6	Generative Adversarial Networks trainieren	191
14.7	Mit einem GAN erzeugte Piktogramme anzeigen	193
14.8	Piktogramme als Zeichenanleitung kodieren	195
14.9	Trainieren eines RNN zum Zeichnen von Piktogrammen	196
14.10	Piktogramme mithilfe eines RNN erzeugen	197
15	Musik und Deep Learning	201
15.1	Einen Trainingsdatensatz zur Musikklassifikation erstellen	202
15.2	Einen Musikgenre-Detektor trainieren	204
15.3	Visualisierung von Klassifikationsirrtümern	206
15.4	Indexierung vorhandener Musik	208
15.5	Die Spotify-API einrichten.	210
15.6	Playlisten und Musikstücke von Spotify sammeln	211
15.7	Ein Musikempfehlungssystem trainieren	214
15.8	Musikstücke empfehlen mithilfe eines Word2vec-Modells	215
16	Machine-Learning-Systeme in Produktion bringen	219
16.1	Ein Nächste-Nachbarn-Klassifikationsmodell für Einbettungen mit scikit-learn verwenden	220
16.2	Postgres zum Speichern von Einbettungen verwenden	221
16.3	Einpflegen und Abfragen von in Postgres gespeicherten Einbettungen	222
16.4	Hochdimensionale Modelle in Postgres speichern	223
16.5	Microservices in Python erstellen	225
16.6	Keras-Modelle als Microservice bereitstellen	226
16.7	Einen Microservice aus einem Web-Framework aufrufen	227
16.8	seq2seq-Modelle in TensorFlow	228
16.9	Deep-Learning-Modelle im Browser ausführen.	230
16.10	Ein Keras-Modell mit TensorFlow Serving ausführen.	232
16.11	Ein Keras-Modell unter iOS verwenden	235
	Index	237