

# Künstliche Intelligenz verstehen

Eine spielerische Einführung

# DAS INHALTS- VERZEICHNIS

» Hier geht's  
direkt  
zum Buch

# Inhalt

Materialien zum Buch .....	16
Vorwort zur zweiten Auflage .....	17

## **1 Einleitung** 19

---

<b>1.1 Worum es uns in diesem Buch geht</b> .....	20
<b>1.2 Für wen wir dieses Buch geschrieben haben</b> .....	21
<b>1.3 Aufbau der einzelnen Kapitel</b> .....	22
<b>1.4 Ein Wort an die Programmierunkundigen</b> .....	22
<b>1.5 Beispielprogramme und die Webseite zum Buch</b> .....	23
<b>1.6 Warum wir JavaScript und p5.js verwendet haben</b> .....	25
<b>1.7 Begriffliche Abgrenzung und Fachbegriffe</b> .....	26
<b>1.8 Inhalte, Themen, Kapitel</b> .....	27
<b>1.9 Dank</b> .....	30

## **2 Texte bauen mit Markow** 31

---

<b>2.1 Das Beispielprogramm Nonsense-Texter</b> .....	35
<b>2.2 Der Code des Nonsense-Texters unter der Lupe</b> .....	37
2.2.1 Die Komponenten des Markow-Objekts .....	38
2.2.2 Übergänge lernen .....	38
2.2.3 Nonsense-Texte produzieren .....	40
<b>2.3 Das Beispielprogramm Wörter vorschlagen</b> .....	43
2.3.1 Die Komponenten des Markow-Objekts .....	43
2.3.2 Übergänge und Häufigkeiten lernen .....	44
<b>2.4 Wörter vorschlagen</b> .....	47
<b>2.5 Gewichteter Zufall</b> .....	48
<b>2.6 Ideen zum Weitermachen</b> .....	50
<b>2.7 Zusammenfassung und Ausblick</b> .....	51

<b>3</b>	<b>Schreibfehler automatisch korrigieren</b>	53
<b>3.1</b>	<b>Das Beispielprogramm Wortvergleich</b>	54
<b>3.2</b>	<b>Die Matrix befüllen</b>	57
3.2.1	Die Füllung der oberen Zeile und der linken Spalte	57
3.2.2	Die Füllung der verbleibenden Zellen	58
3.2.3	Die drei Schritte unter der Lupe	62
<b>3.3</b>	<b>Die Umsetzung im Beispielprogramm</b>	62
3.3.1	Das Levenshtein-Objekt	62
3.3.2	Die Funktion matrix()	63
<b>3.4</b>	<b>Das Beispielprogramm Korrekturvorschläge</b>	65
<b>3.5</b>	<b>Ideen zum Weitermachen</b>	67
<b>3.6</b>	<b>Zusammenfassung und Ausblick</b>	68
<b>4</b>	<b>Wörter gruppieren</b>	69
<b>4.1</b>	<b>Items und Transaktionen</b>	71
<b>4.2</b>	<b>Kenngrößen der Assoziationsanalyse</b>	72
4.2.1	Support	72
4.2.2	Confidence	73
4.2.3	Lift	74
<b>4.3</b>	<b>Ein Beispiel von Hand gerechnet</b>	76
<b>4.4</b>	<b>Das Beispielprogramm Begriffsnetz</b>	79
4.4.1	Die Datenquelle	81
4.4.2	Beschränkung der Anwendung auf Item-Paare	81
<b>4.5</b>	<b>Eine Tour durch den Code</b>	82
4.5.1	Die Klasse Begriffsnetz	82
4.5.2	Enkodierung der Transaktionen	83
4.5.3	Befüllung der Arrays für Support	85
4.5.4	Befüllung der Arrays für Confidence und Lift	87
4.5.5	Die Funktion assoziationen()	87
<b>4.6</b>	<b>Ideen zum Weitermachen</b>	88

<b>4.7</b>	<b>Zusammenfassung und Ausblick</b> .....	90
4.7.1	Der Apriori-Algorithmus .....	90
4.7.2	Transaktionstabellen .....	90
4.7.3	Eine Übersicht aller Fachbegriffe aus diesem Kapitel .....	91

## **5 Spiele für eine Person lösen** 93

---

<b>5.1</b>	<b>Das Spiel Fruchtkräsch</b> .....	93
<b>5.2</b>	<b>Wie findet die KI den besten Zug?</b> .....	95
<b>5.3</b>	<b>Eine vielseitig einsetzbare Spiel-KI</b> .....	98
<b>5.4</b>	<b>Die Klasse Spielzustand</b> .....	99
5.4.1	Züge liefern modifizierte Spielzustände .....	100
5.4.2	Die möglichen Züge .....	100
5.4.3	Die Bewertung eines Spielzustands .....	100
5.4.4	Ein Gedächtnis für Züge .....	101
5.4.5	Die Schnittstelle im Überblick .....	101
<b>5.5</b>	<b>Die Klasse KI</b> .....	102
5.5.1	Alle Folgezustände eines Spielzustands berechnen .....	102
5.5.2	Alle Spielverläufe per Warteschlangenverfahren berechnen .....	103
5.5.3	Die Spielzustände nach Bewertung sortieren .....	106
5.5.4	Die Funktion <code>besterZug()</code> .....	106
5.5.5	Die Funktionen des KI-Objekts im Überblick .....	107
<b>5.6</b>	<b>Ideen zum Weitermachen</b> .....	107
<b>5.7</b>	<b>Zusammenfassung und Ausblick</b> .....	108

## **6 Spiele für zwei Personen gewinnen** 109

---

<b>6.1</b>	<b>Das Spiel Reversi</b> .....	110
<b>6.2</b>	<b>Das Beispielprogramm Reversi KI</b> .....	111
<b>6.3</b>	<b>Der Minimax-Algorithmus</b> .....	112
6.3.1	Anwendungsgebiete und Grenzen des Minimax-Algorithmus .....	114

<b>6.4</b>	<b>Tiefensuche und Rekursion</b> .....	115
6.4.1	Breitensuche und Tiefensuche .....	116
6.4.2	Die Paradoxie der Rekursion .....	117
6.4.3	Verzweigte Rekursion .....	121
<b>6.5</b>	<b>Die Klasse Spielzustand</b> .....	123
6.5.1	Die Bewertungsfunktion .....	123
6.5.2	Die Schnittstelle im Überblick .....	125
<b>6.6</b>	<b>Die Klasse KI</b> .....	126
<b>6.7</b>	<b>Beschleunigung mit Alpha-Beta-Pruning</b> .....	130
<b>6.8</b>	<b>Ideen zum Weitermachen</b> .....	131
<b>6.9</b>	<b>Zusammenfassung und Ausblick</b> .....	132
<b>7</b>	<b>Q-Learning</b> .....	133
<hr/>		
<b>7.1</b>	<b>Das Eichhörnchen und das Nussversteck</b> .....	134
<b>7.2</b>	<b>Umwelt, Agent, Aktion und Belohnung</b> .....	139
7.2.1	Das Verhältnis von Agent und Umwelt .....	140
<b>7.3</b>	<b>Die Q-Tabelle</b> .....	141
7.3.1	Q steht für Qualität .....	142
<b>7.4</b>	<b>Das Beispielprogramm Q-Lerner</b> .....	142
<b>7.5</b>	<b>Die Q-Tabelle befüllen</b> .....	147
7.5.1	Warum funktioniert das? .....	149
<b>7.6</b>	<b>Der Code unter der Lupe</b> .....	150
7.6.1	Die Umwelt .....	150
7.6.2	Der Q-Lerner .....	151
<b>7.7</b>	<b>Gamma bestimmt die Weitsicht</b> .....	152
<b>7.8</b>	<b>Epsilon: Erforschung oder Anwendung</b> .....	154
<b>7.9</b>	<b>Ein zweiter Blick auf den Code</b> .....	156
<b>7.10</b>	<b>Alpha</b> .....	158
<b>7.11</b>	<b>Was wir weggelassen haben</b> .....	159
7.11.1	Komplexere Umwelten .....	160
7.11.2	Kosten für Aktionen .....	160

7.11.3	Belohnungen mit Zustands-Aktions-Paaren verknüpfen .....	160
7.11.4	Mehrere Belohnungszustände .....	160
<b>7.12</b>	<b>Ideen zum Weitermachen .....</b>	<b>161</b>
7.12.1	OpenAI Gym .....	162
7.12.2	Das Buch von Sutton und Barto .....	163
<b>7.13</b>	<b>Zusammenfassung und Ausblick .....</b>	<b>163</b>
7.13.1	Menschliches Lernen vs. Q-Learning .....	163
7.13.2	Die Grenzen des Verfahrens .....	164
<b>8</b>	<b>K-nächste-Nachbarn .....</b>	<b>167</b>
<hr/>		
<b>8.1</b>	<b>Häschen, Igel, Vogelspinne oder Hai? .....</b>	<b>168</b>
<b>8.2</b>	<b>Das Beispielprogramm Tiere erkennen .....</b>	<b>169</b>
<b>8.3</b>	<b>Entfernungen bestimmen mit Pythagoras .....</b>	<b>172</b>
<b>8.4</b>	<b>Der Code im Detail .....</b>	<b>175</b>
<b>8.5</b>	<b>Ideen zum Weitermachen .....</b>	<b>178</b>
<b>8.6</b>	<b>Zusammenfassung und Ausblick .....</b>	<b>179</b>
<b>9</b>	<b>K-means-Clustering .....</b>	<b>181</b>
<hr/>		
<b>9.1</b>	<b>Clusterbildung in Aktion .....</b>	<b>183</b>
9.1.1	Mittelwert, Zentrum, Schwerpunkt .....	183
9.1.2	Die Schrittfolge des k-means-Clustering-Algorithmus .....	185
<b>9.2</b>	<b>Das Beispielprogramm Wetterdaten gruppieren .....</b>	<b>186</b>
<b>9.3</b>	<b>Der Code .....</b>	<b>188</b>
9.3.1	Zentren zufällig setzen .....	189
9.3.2	Datenpunkte zuordnen .....	189
9.3.3	Zentren neu berechnen .....	190
<b>9.4</b>	<b>Grenzen des Verfahrens .....</b>	<b>191</b>
9.4.1	Unsinnige Gruppierungen .....	191
9.4.2	Zu viele Dimensionen .....	192
9.4.3	Linear nicht trennbare Datenpunktmengen .....	194

9.5	Ideen zum Weitermachen .....	195
9.6	Zusammenfassung und Ausblick .....	195
<b>10</b>	<b>Neuronale Netze I: Das Häschenproblem</b> .....	<b>197</b>
10.1	Bildererkennung: ein klassisches Problem .....	198
10.2	Was ist ein Modell? .....	199
10.3	Der Aufbau eines neuronalen Netzes .....	201
10.4	Das Häschenneuron und seine Kollegen .....	204
10.4.1	Die biologische Nervenzelle als Vorbild .....	205
10.4.2	Das künstliche Neuron .....	207
10.4.3	b steht für Bias .....	208
10.4.4	Die Aktivierungsfunktion .....	208
10.5	Das Beispielprogramm Tiere erkennen II .....	209
10.6	Der Code .....	211
10.7	Ideen zum Weitermachen .....	211
10.8	Zusammenfassung und Ausblick .....	212
<b>11</b>	<b>Neuronale Netze II: Auf dem Weg ins Tal</b> .....	<b>213</b>
11.1	Das überwachte Lernen .....	214
11.2	Die schrittweise Justierung des Modells .....	216
11.2.1	Die grundlegende Idee .....	217
11.2.2	Steigung .....	218
11.2.3	Tangente .....	219
11.2.4	Ableitung .....	219
11.2.5	Der Gradientenabstieg .....	220
11.2.6	Die Lernrate .....	222
11.3	Das Beispielprogramm Gradientenabstieg .....	223
11.4	Der Code .....	225
11.5	Tipps zum Weitermachen .....	226
11.6	Zusammenfassung und Ausblick .....	226

<b>12</b>	<b>Neuronale Netze III: Fehler zurückverfolgen mit dem Neuronentrainer</b>	229
<b>12.1</b>	<b>Was ist Backpropagation?</b>	230
<b>12.2</b>	<b>Das Beispielprogramm Neuronentrainer</b>	231
12.2.1	Aufgaben und Netzarchitekturen	232
12.2.2	Ein Wiedersehen mit dem Häschenproblem	235
12.2.3	Lineare Trennbarkeit	236
<b>12.3</b>	<b>Validierungsdaten, Überanpassung, Generatoren</b>	237
12.3.1	Validierungsdaten und Überanpassung	238
12.3.2	Generatoren	239
<b>12.4</b>	<b>Weitere Beispielaufgaben</b>	240
12.4.1	Kreis und Hintergrund	240
12.4.2	Quadrat und Hintergrund	241
12.4.3	Farbtunnel sieben Farben	242
<b>12.5</b>	<b>Die Anzahlen der verdeckten Schichten und der Neuronen</b>	244
12.5.1	Viel hilft viel?	244
<b>12.6</b>	<b>Was wir weggelassen haben</b>	245
12.6.1	Dynamisierung der Lernrate	245
12.6.2	Batch und Epoche	245
12.6.3	Verlustfunktionen und Softmax	245
<b>12.7</b>	<b>Ideen zum Weitermachen</b>	246
<b>12.8</b>	<b>Zusammenfassung und Ausblick</b>	248
<b>13</b>	<b>Neuronale Netze IV: Faltungsnetze, Autoencoder, GANs und DQL</b>	249
<b>13.1</b>	<b>Faltungsnetze</b>	249
13.1.1	Das Beispielprogramm Filterlabor I	250
13.1.2	Der Filterkernel in Aktion	251
13.1.3	Padding und Striding	253
13.1.4	Das Beispielprogramm Filterlabor II	254
13.1.5	Eine Filterkombination, die den Buchstaben K erkennt	255
13.1.6	Die Struktur eines Faltungsnetzes	256
13.1.7	Faltungsnetze trainieren	257



<b>13.2</b>	<b>Modelle, die Bilder erzeugen</b> .....	258
<b>13.3</b>	<b>Autoencoder</b> .....	260
13.3.1	Dimensionsreduktion .....	260
13.3.2	Daten ausdenken .....	261
<b>13.4</b>	<b>Generative Adversarial Networks</b> .....	261
<b>13.5</b>	<b>Deep Q-Learning</b> .....	264
13.5.1	Wie kommt die Umwelt ins Modell? .....	265
<b>13.6</b>	<b>Zusammenfassung und Ausblick</b> .....	265
<b>14</b>	<b>Transformer verstehen</b> .....	267
<hr/>		
<b>14.1</b>	<b>Ein Sprachmodell von außen betrachtet</b> .....	267
14.1.1	Unterschiede zum Markow-Prozess .....	268
<b>14.2</b>	<b>Wörter in Zahlen codieren für Fortgeschrittene</b> .....	269
<b>14.3</b>	<b>Worteinbettungen</b> .....	270
14.3.1	Relationen übertragen .....	274
<b>14.4</b>	<b>Das Beispielprogramm Wort-Navigator</b> .....	276
14.4.1	Textquelle I: Dünstende Köchinnen und frittierende Konditoren .....	276
14.4.2	Die Bedienoberfläche .....	277
14.4.3	Relationen übertragen im Wort-Navigator .....	279
<b>14.5</b>	<b>Vom Text zur Worteinbettung</b> .....	282
14.5.1	One Hot Encoding .....	283
14.5.2	Paare aus Ziel- und Kontextwörtern .....	284
14.5.3	Das Training .....	286
14.5.4	Vom Training zur Worteinbettung .....	287
14.5.5	Warum funktioniert das? .....	288
14.5.6	Kosinus-Ähnlichkeit .....	289
<b>14.6</b>	<b>Vom Wort zum Satz zum Text: Sequenzeinbettungen</b> .....	290
<b>14.7</b>	<b>Das Beispielprogramm Sequenz-Navigator</b> .....	292
14.7.1	Textquelle II: Ein Tag im Leben von Kati Katzenstein .....	292
14.7.2	Die Bedienoberfläche .....	294
14.7.3	Relationen zwischen Sequenzen .....	296

<b>14.8</b>	<b>Transformer am Horizont</b> .....	297
14.8.1	Ein verbesserter Markow-Prozess .....	297
14.8.2	Und täglich grüßt Frau Katzenstein .....	298
14.8.3	Positional Encodings .....	299
14.8.4	Gewichtungen .....	299
14.8.5	Sprache ist Kontext .....	299
14.8.6	Aufmerksamkeit .....	300
<b>14.9</b>	<b>Zusammenfassung und Ausblick</b> .....	301
14.9.1	Was wir ausgelassen haben .....	301
14.9.2	Unterschiede zu ausgewachsenen Transformatoren .....	302
14.9.3	Das erstaunliche Abstraktionsvermögen von Transformatoren .....	303
<b>14.10</b>	<b>Ideen zum Weitermachen</b> .....	304

<b>Nachwort: Auf der Suche nach Trurls Elektrobarden</b>	305
--	-----

---

<b>Anhang</b>	313
---------------	-----

---

<b>A</b>	<b>Eine kurze Einführung in JavaScript und p5.js</b> .....	315
<b>B</b>	<b>Glossar</b> .....	359
<b>C</b>	<b>Quellen und weiterführende Literatur</b> .....	369
<b>D</b>	<b>Abbildungsverzeichnis</b> .....	373

Index .....	377
-------------	-----

# Materialien zum Buch

Auf der Webseite des Autors Pit Noack finden Sie alle Beispielprogramme zum direkten Ausprobieren im Browser und weitere Materialien zum Buch:

<https://www.maschinennah.de/ki-buch/>


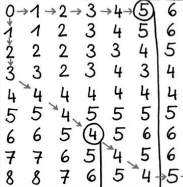
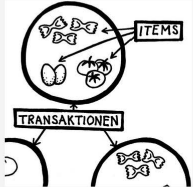
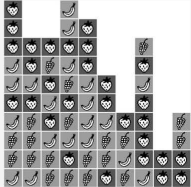
<b>Kapitel 2</b> Texte bauen mit Markow	<b>Kapitel 3</b> Schreibfehler automatisch korrigieren	<b>Kapitel 4</b> Wörter gruppieren	<b>Kapitel 5</b> Spiele für eine Person lösen
 <p><b>Beispielprogramm Nonsense-Texter</b></p> <p><b>STARTEN</b></p> <p><input type="checkbox"/> im Editor öffnen</p> <p><b>Beispielprogramm Wörter vorschlagen</b></p>	 <p><b>Beispielprogramm Wortvergleich</b></p> <p><b>STARTEN</b></p> <p><input type="checkbox"/> im Editor öffnen</p> <p><b>Beispielprogramm Korrekturvorschläge</b></p>	 <p><b>Beispielprogramm Assoziationsanalyse</b></p> <p><b>STARTEN</b></p> <p><input type="checkbox"/> im Editor öffnen</p> <p>&gt; mehr...</p>	 <p><b>Beispielprogramm Fruchtkräsch</b></p> <p><b>STARTEN</b></p> <p><input type="checkbox"/> im Editor öffnen</p> <p>&gt; mehr...</p>

Abbildung 1 Online können Sie die Projekte ausprobieren und den Code bearbeiten.

Auf der Verlagswebseite zu diesem Buch stehen Ihnen die Beispielprogramme 5 zusätzlich zum Download zur Verfügung.

Gehen Sie auf <http://www.rheinwerk-verlag.de/5810>. Klicken Sie auf den Reiter MATERIALIEN. Klicken Sie auf den Button HERUNTERLADEN, um den Download zu starten.