

Rudolf Kruse | Christian Borgelt | Frank Klawonn
Christian Moewes | Georg Ruß | Matthias Steinbrecher

Computational Intelligence

Aus den Kinderschuhen der „Künstlichen Intelligenz“ entwachsen bietet die Reihe breitgefächertes Wissen von den Grundlagen bis in die Anwendung, herausgegeben von namhaften Vertretern ihres Faches.

Computational Intelligence hat das weitgesteckte Ziel, das Verständnis und die Realisierung intelligenten Verhaltens voranzutreiben. Die Bücher der Reihe behandeln Themen aus den Gebieten wie z. B. Künstliche Intelligenz, Softcomputing, Robotik, Neuro- und Kognitionswissenschaften. Es geht sowohl um die Grundlagen (in Verbindung mit Mathematik, Informatik, Ingenieurs- und Wirtschaftswissenschaften, Biologie und Psychologie) wie auch um Anwendungen (z. B. Hardware, Software, Webtechnologie, Marketing, Vertrieb, Entscheidungsfindung). Hierzu bietet die Reihe Lehrbücher, Handbücher und solche Werke, die maßgebliche Themengebiete kompetent, umfassend und aktuell repräsentieren.

Unter anderem sind erschienen:

Methoden wissensbasierter Systeme

von Christoph Beierle und Gabriele Kern-Isberner

Grundkurs Spracherkennung

von Stephan Euler

Grundkurs Künstliche Intelligenz

von Wolfgang Ertel

Quantum Computing verstehen

von Matthias Homeister

Data Mining

von Thomas A. Runkler

Rudolf Kruse | Christian Borgelt | Frank Klawonn
Christian Moewes | Georg Ruß | Matthias Steinbrecher

Computational Intelligence

Eine methodische Einführung
in Künstliche Neuronale Netze, Evolutionäre Algorithmen,
Fuzzy-Systeme und Bayes-Netze

Mit 215 Abbildungen, 43 Tabellen und 24 Algorithmen

STUDIUM



VIEWEG+
TEUBNER

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der
Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über
<<http://dnb.d-nb.de>> abrufbar.

Das in diesem Werk enthaltene Programm-Material ist mit keiner Verpflichtung oder Garantie irgendeiner Art verbunden. Der Autor übernimmt infolgedessen keine Verantwortung und wird keine daraus folgende oder sonstige Haftung übernehmen, die auf irgendeine Art aus der Benutzung dieses Programm-Materials oder Teilen davon entsteht.

Höchste inhaltliche und technische Qualität unserer Produkte ist unser Ziel. Bei der Produktion und Auslieferung unserer Bücher wollen wir die Umwelt schonen: Dieses Buch ist auf säurefreiem und chlorfrei gebleichtem Papier gedruckt. Die Einschweißfolie besteht aus Polyäthylen und damit aus organischen Grundstoffen, die weder bei der Herstellung noch bei der Verbrennung Schadstoffe freisetzen.

1. Auflage 2011

Alle Rechte vorbehalten

© Vieweg+Teubner Verlag | Springer Fachmedien Wiesbaden GmbH 2011

Lektorat: Christel Roß | Maren Mithöfer

Vieweg+Teubner Verlag ist eine Marke von Springer Fachmedien.

Springer Fachmedien ist Teil der Fachverlagsgruppe Springer Science+Business Media.

www.viewegteubner.de



Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlags unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Umschlaggestaltung: KünkelLopka Medienentwicklung, Heidelberg

Druck und buchbinderische Verarbeitung: AZ Druck und Datentechnik, Berlin

Gedruckt auf säurefreiem und chlorfrei gebleichtem Papier

Printed in Germany

ISBN 978-3-8348-1275-9

Vorwort

Das Gebiet Computational Intelligence als Teilgebiet der Künstlichen Intelligenz umfasst Konzepte, Paradigmen, Algorithmen und Implementierungen zur Entwicklung von Systemen, die intelligentes Verhalten in komplexen Umgebungen automatisieren sollen. Dazu werden subsymbolische, vornehmlich naturanaloge Methoden verwendet, die unvollständiges, unpräzises und unsicheres Wissen tolerieren und auf diese Weise approximative, handhabbare, robuste und ressourcengünstige Lösungen ermöglichen.

Die Themenauswahl des Buches spiegelt die wichtigsten Gebiete des Bereichs Computational Intelligence wider. Die klassischen Gebiete *Künstliche Neuronale Netze*, *Fuzzy-Systeme* und *Evolutionäre Algorithmen* werden detailliert beschrieben, jedoch werden auch neuere Methoden wie *Schwarmintelligenz* und *Probabilistische Graphische Modelle* in das Buch integriert.

Unser Ziel ist es, mit diesem Lehrbuch eine methodische Einführung in das Gebiet Computational Intelligence zu geben. Uns geht es nicht nur um die Vermittlung fundamentaler Konzepte und deren Umsetzung; es geht auch darum, den theoretischen Hintergrund der vorgeschlagenen Problemlösungen zu erklären und den Lesern die für den fundierten Einsatz dieser Methoden notwendige Sensibilität zu vermitteln.

Dieses Lehrbuch ist primär als Begleitbuch für Vorlesungen im Gebiet Computational Intelligence nutzbar, es kann aber auch von Studenten und Praktikern aus Industrie und Wirtschaft für ein Selbststudium verwendet werden.

Das Buch basiert auf Aufzeichnungen zu Vorlesungen, Übungen und Praktika, die von den Autoren seit vielen Jahren gehalten werden. Auf der Webseite

<http://www.computational-intelligence.eu>

findet man für die vier Vorlesungen Neuronale Netze, Evolutionäre Algorithmen, Fuzzy-Systeme und Bayes-Netze folgende Unterlagen: Modulbeschreibungen, Vorlesungsfolien, Übungsaufgaben mit Lösungen, Hinweise zu Softwaretools und Ergänzungsmaterial.

Wir bedanken uns beim Vieweg+Teubner-Verlag für die gute Zusammenarbeit.

Magdeburg, Juli 2011

Rudolf Kruse
Christian Borgelt
Frank Klawonn
Christian Moewes
Georg Ruß
Matthias Steinbrecher

Inhaltsverzeichnis

1	Einleitung	1
1.1	Intelligente Systeme	1
1.2	Computational Intelligence	2
1.3	Über dieses Buch	4
I	Neuronale Netze	5
2	Künstliche neuronale Netze	7
2.1	Biologische Grundlagen	9
3	Schwellenwertelemente	13
3.1	Definition und Beispiele	13
3.2	Geometrische Deutung	15
3.3	Grenzen der Ausdrucksmächtigkeit	17
3.4	Netze von Schwellenwertelementen	18
3.5	Training der Parameter	21
3.6	Varianten	31
3.7	Training von Netzen	32
4	Allgemeine neuronale Netze	33
4.1	Struktur neuronaler Netze	33
4.2	Arbeitsweise neuronaler Netze	36
4.3	Training neuronaler Netze	40
5	Mehrschichtige Perzeptren	43
5.1	Definition und Beispiele	43
5.2	Funktionsapproximation	49
5.3	Logistische Regression	55
5.4	Gradientenabstieg	58
5.5	Fehler-Rückpropagation	63
5.6	Beispiele zum Gradientenabstieg	64
5.7	Varianten des Gradientenabstiegs	69
5.8	Beispiele zu einigen Varianten	74
5.9	Sensitivitätsanalyse	77

6 Radiale-Basisfunktionen-Netze	79
6.1 Definition und Beispiele	79
6.2 Funktionsapproximation	84
6.3 Initialisierung der Parameter	86
6.4 Training der Parameter	94
6.5 Verallgemeinerte Form	98
7 Selbstorganisierende Karten	101
7.1 Definition und Beispiele	101
7.2 Lernende Vektorquantisierung	104
7.3 Nachbarschaft der Ausgabeneuronen	112
8 Hopfield-Netze	117
8.1 Definition und Beispiele	117
8.2 Konvergenz der Berechnungen	121
8.3 Assoziativspeicher	126
8.4 Lösen von Optimierungsproblemen	131
8.5 Simuliertes Ausglühen	137
9 Rückgekoppelte Netze	139
9.1 Einfache Beispiele	139
9.2 Darstellung von Differentialgleichungen	144
9.3 Vektorielle neuronale Netze	146
9.4 Fehler-Rückpropagation in der Zeit	149
II Evolutionäre Algorithmen	153
10 Evolutionäre Algorithmen	155
10.1 Biologische Evolution	155
10.2 Simulierte Evolution	158
10.3 Das n-Damen-Problem	161
10.4 Vergleich mit der biologischen Evolution	165
10.5 Vergleich mit klassischen Optimierungsverfahren	166
11 Elemente evolutionärer Algorithmen	169
11.1 Kodierung	169
11.2 Fitness und Selektion	175
11.3 Genetischen Operatoren	183
12 Evolutionäre Basisalgorithmen	193
12.1 Genetischer Algorithmus	194
12.2 Lokale Suchverfahren	202
12.3 Evolutionsstrategien	209
12.4 Genetische Programmierung	217
12.5 Weitere populationsbasierte Verfahren	226

13	Spezielle Anwendungen und Techniken evolutionärer Algorithmen	237
13.1	Verhaltenssimulation	237
13.2	Mehrkriterienoptimierung	241
13.3	Parallelisierung	248
III	Fuzzy-Systeme	251
14	Fuzzy-Mengen und Fuzzy-Logik	253
14.1	Natürliche Sprache und formale Modelle	253
14.2	Fuzzy-Mengen	254
14.3	Repräsentation von Fuzzy-Mengen	257
14.4	Fuzzy-Logik	263
14.5	Operationen auf Fuzzy-Mengen	272
15	Das Extensionsprinzip	279
15.1	Abbildungen von Fuzzy-Mengen	279
15.2	Abbildungen von Niveaumengen	281
15.3	Kartesisches Produkt und zylindrische Erweiterung	282
15.4	Extensionsprinzip für mehrelementige Abbildungen	284
16	Fuzzy-Relationen	287
16.1	Gewöhnliche Relationen	287
16.2	Anwendung von Relationen und Inferenz	289
16.3	Inferenzketten	292
16.4	Einfache Fuzzy-Relationen	293
16.5	Verkettung von Fuzzy-Relationen	297
17	Ähnlichkeitsrelationen	301
17.1	Fuzzy-Mengen und extensionale Hüllen	302
17.2	Skalierungskonzepte	304
17.3	Interpretation von Fuzzy-Mengen	307
18	Possibilitätstheorie und verallgemeinerte Maße	313
19	Fuzzy-Regelsysteme	315
19.1	Mamdani-Regler	315
19.2	Takagi-Sugeno-Kang-Regler	325
19.3	Mamdani-Regler und Ähnlichkeitsrelationen	326
19.4	Logikbasierte Regler	331
20	Fuzzy-Relationale Gleichungen	335
20.1	Lösbarkeit von Fuzzy-Relationale Gleichungen	335
20.2	Fuzzy-Regelsysteme und -Relationale Gleichungen	336

21 Fuzzy-Clusteranalyse	339
21.1 Clusteranalyse	339
21.2 Fuzzy-c-Means-Clustering	341
21.3 Bestimmung der Clusteranzahl	344
21.4 Verallgemeinerungen von Fuzzy-c-Means	344
IV Bayes-Netze	347
22 Bayes-Netze	349
23 Grundlagen der Wahrscheinlichkeits- und Graphentheorie	355
23.1 Wahrscheinlichkeitstheorie	355
23.2 Graphentheorie	364
24 Zerlegungen	387
24.1 Abhängigkeits- und Unabhängigkeitsgraphen	391
24.2 Eine reale Anwendung	399
25 Evidenzpropagation	403
26 Lernen Graphischer Modelle	419
A Anhänge	429
A.1 Geradengleichungen	429
A.2 Regression	431
A.3 Aktivierungsumrechnung	435
Literaturverzeichnis	437
Index	449

Kapitel 1

Einleitung

1.1 Intelligente Systeme

Komplexe Problemstellungen in verschiedensten Anwendungsbereichen führen verstärkt zu Computeranwendungen, die „intelligentes Verhalten“ aufweisen müssen. Diese Anwendungen leisten z. B. Entscheidungsunterstützung, steuern und kontrollieren Prozesse, erkennen und interpretieren Muster oder bewegen sich autonom in unbekanntem Umgebungen. Zur Bewältigung solcher Aufgaben sind neuartige Vorgehensweisen, Methoden, Programmierumgebungen und Tools entwickelt worden.

Die allgemeine Aufgabenstellung bei der Entwicklung solcher „intelligenten Systeme“ ist auf einer höheren Abstraktionsebene letztlich immer die gleiche. Immer geht es um die Simulation intelligenten Denkens und Handelns in einem bestimmten Anwendungsbereich [Russell u. Norvig 2009]. Das Wissen über diesen Anwendungsbereich muss zu diesem Zweck dargestellt und verarbeitet werden. Die Qualität des sich ergebenden Systems hängt zum Großteil von der Lösung gerade dieses Darstellungsproblems des Wissens im Entwicklungsprozess ab. Es gibt nicht die „beste“ Methode, sondern es gilt vielmehr, aus den vielen bereitstehenden Ansätzen diejenigen herauszusuchen, die optimal zum Einsatzgebiet des Systems passen.

Die Mechanismen, die intelligentem Verhalten zugrunde liegen, werden im Forschungsgebiet Künstliche Intelligenz untersucht. Die Künstliche Intelligenz (KI) ist ein Teilgebiet der Informatik und hat, wie andere Teilgebiete auch, sowohl theoretische Aspekte (*Wie und warum funktionieren die Systeme?*) und anwendungsorientierte Aspekte (*Wo und wann können die Systeme verwendet werden?*) [Luger 2001].

Zu Beginn der Entwicklung von intelligenten Systemen hat man sich oft an der aus der Aufklärung stammenden Vorstellung vom „Menschen als Maschine“ orientiert. Man wollte eine Intelligenz erschaffen, die wie der Mensch kreativ denken sowie Probleme lösen kann, und die sich durch eine Form von Bewusstsein sowie Emotionen auszeichnet. Der übliche Weg zum Entwurf künstlicher Intelligenz war in der Anfangsphase der KI immer die Beschreibung einer symbolischen Basis dieser Mechanismen. Dazu gehört ebenso die top-down-Perspektive der Problemlösung, die als wichtigsten Punkt die Frage behandelt, warum die Systeme funktionieren [Minsky 1991]. Die Antwort auf diese Frage wird meist mit Hilfe symbolischer Repräsentationen und logischer Prozesse formuliert. Zu diesen Ansätzen gehören beispielsweise

se spezielle Verfahren wie regelbasierte Expertensysteme, automatische Theorembe-
weiser und viele Operations-Research-Ansätze, die moderner Planungssoftware zu-
grundeliegen. Obwohl diese traditionellen Ansätze teilweise sehr erfolgreich waren
und sind, haben sie deutliche Grenzen, insbesondere was die Skalierbarkeit angeht.
Eine leichte Erschwerung des zu lösenden Problems geht meist mit einer nicht hand-
habbaren Komplexitätssteigerung einher. Somit sind diese Verfahren allein, obwohl
sie eine optimale, präzise oder wahre Lösung garantieren, für praktische Probleme
häufig nicht einsetzbar [Beierle u. Kern-Isberner 2008].

Deswegen werden weiterhin effiziente Methoden zur Verarbeitung und Reprä-
sentation von Wissen gesucht. Bewährt haben sich für einige Problemstellungen Ver-
fahren, die sich an natürlichen bzw. biologischen Prozessen orientieren [Brownlee
2011]. Diese Herangehensweise stellt einen Paradigmenwechsel weg von der sym-
bolischen Repräsentation und hin zu Inferenzstrategien für Anpassung und Lernen
dar. Zu diesen Ansätzen gehören Künstliche Neuronale Netze, Evolutionäre Algo-
rithmen und Fuzzy-Systeme [Pedrycz 1997, Engelbrecht 2007]. Diese neuen Metho-
den haben sich, meist in Kombination mit traditionellen Problemlösungstechniken,
in vielen Anwendungsbereichen bewährt.

1.2 Computational Intelligence

Das Forschungsgebiet der Computational Intelligence (CI) als Teilgebiet der Künstli-
chen Intelligenz umfasst Konzepte, Paradigmen, Algorithmen und Implementierun-
gen zur Entwicklung von Systemen, die intelligentes Verhalten in komplexen Umge-
bungen automatisieren sollen. Es werden subsymbolische, naturanaloge Methoden
verwendet, die unvollständiges, unpräzises und unsicheres Wissen tolerieren und
auf diese Weise approximative, handhabbare, robuste und ressourcengünstige Lö-
sungen ermöglichen.

Die im Bereich Computational Intelligence verwendete Problemlösungsstrategie
besteht darin, approximative Techniken und Methoden zu verwenden, die ungefähre,
unvollständige oder nur partiell wahre Lösungen zu Problemen finden können,
dies aber im Gegenzug in einem akzeptablen Zeit- und Kostenrahmen bewerkstelligen.
Diese Ansätze bestehen aus relativ einfachen Teilabläufen, die im Zusammen-
spiel zu komplexem und selbstorganisierendem Verhalten führen. Dadurch entzie-
hen sich diese heuristischen Verfahren oft einer klassischen Analyse, sie sind aber in
der Lage, schnell ungefähre Lösungen zu ansonsten nur schwierig lösbaren Proble-
men zu generieren.

Bei einem Umfang von 400 Seiten kann das Gebiet Computational Intelligence
in diesem Buch nicht vollständig behandelt werden. Wir beschränken uns daher auf
die Beschreibung von vier in der Praxis oft verwendeten Techniken.

In den ersten beiden Teilen dieses Buches werden sogenannte „naturanaloge“
Verfahren beschrieben. Die hier verfolgte Idee besteht darin, in der Natur vorkom-
mende Problemlösungsstrategien zu analysieren. Teilaspekte der Lösungsstrategie
werden dann auf dem Computer modellieren oder simuliert, ohne dabei die Ur-
sprungssysteme selbst korrekt modellieren zu wollen und ohne die biologische Plau-
sibilität der Systeme zu berücksichtigen. Besonders erfolgreiche wichtige Vertreter

dieser Gattung sind derzeit Neuronale Netze [Bishop 1995], Evolutionäre Algorithmen [Goldberg 1989] und Schwarmbasierte Verfahren [Kennedy u. a. 2001].

Viele Ideen und Prinzipien im Gebiet der neuronalen Netze wurden durch die Hirnforschung inspiriert. Künstliche Neuronale Netze sind informationsverarbeitende Systeme, deren Struktur und Funktionsweise dem Nervensystem und speziell dem Gehirn von Tieren und Menschen nachempfunden sind. Sie bestehen aus einer großen Anzahl einfacher, parallel arbeitender Einheiten, den sogenannten Neuronen. Diese Neuronen senden sich Informationen in Form von Aktivierungssignalen über gerichtete Verbindungen zu. Ausgehend von dem Wissen über die Funktion biologischer neuronaler Netze versucht man, diese zu modellieren und zu simulieren.

Die Idee zu evolutionären Algorithmen stammt aus der biologischen Evolution, in deren Rahmen sich Organismen an Umweltbedingungen anpassen. Evolutionäre Algorithmen stellen eine Klasse von Optimierungsverfahren dar, die Prinzipien der biologischen Evolution nachahmen. Sie gehören zur Gruppe der Metaheuristiken, die Algorithmen zur näherungsweise Lösung, z. B. eines kombinatorischen Optimierungsproblems beinhalten. Diese sind definiert durch eine abstrakte Folge von Schritten, die auf beliebige Problemstellungen anwendbar ist. Jeder einzelne Schritt muss allerdings problemspezifisch implementiert werden. Aufgrund dessen wird auch von problemspezifischen Heuristiken gesprochen. Metaheuristiken kommen bei Problemen zum Einsatz, bei denen kein effizienterer Lösungsalgorithmus bekannt ist. Das Finden einer optimalen Lösung ist in der Regel nicht garantiert. Jede gute Lösung kann beliebig schlecht sein, wenn sie mit der optimalen Lösung verglichen wird. Der Erfolg und die Laufzeit hängen von der Problemdefinition und der Implementierung der einzelnen Schritte ab.

In den beiden folgenden Teilen dieses Buches geht es um die Einbeziehung von unsicherem, vagem und unvollständigem Wissen in die Problemlösungsstrategie. Die hier verfolgte Idee besteht darin, dass Menschen sehr gut mit imperfektem Wissen umgehen können und man diese Art von Wissen dem Computer zugänglich machen möchte. Besonders erfolgreiche Ansätze, die mit vagem und unsicherem Wissen umgehen, sind Fuzzy-Systeme [Kruse u. a. 1995] und Bayes-Netze [Borgelt u. a. 2009].

In Fuzzy-Systemen wird das vage Wissen, das von einem Experten bereitgestellt oder von Entwickler des Systems intuitiv formuliert wird, mit Hilfe von Fuzzy-Logiken und Methoden des approximativen Schließens formalisiert und in die Problemlösungsstrategie eingebunden. Diese Verfahren werden routinemäßig in der Regelungstechnik eingesetzt, weil in vielen Anwendungsfällen eine präzise und vollständige Systemmodellierung impraktikabel oder gar unmöglich ist.

Bayes-Netze dienen der effizienten Speicherung und Verarbeitung unsicheren Wissens in komplexen Anwendungsbereichen. Formal ist ein Bayes-Netz ein probabilistisches graphisches Modell, das eine Menge von Zufallsvariablen und deren bedingte Abhängigkeiten in einem gerichteten azyklischen Graph repräsentiert. Aufgrund der probabilistischen Repräsentation kann man sehr gut Schlussfolgerungen anhand von neuen Informationen durchführen, Abhängigkeitsanalysen durchführen und Lernverfahren nutzen.

In vielen Anwendungsfällen werden hybride Computational-Intelligence-Systeme wie Neuro-Fuzzy Systeme genutzt. Manchmal werden diese Verfahren auch mit verwandten Methoden kombiniert, wie beispielsweise im Gebiet des maschinellen