

## Computational Intelligence

Eine Einführung

Bearbeitet von  
Oliver Kramer

1. Auflage 2009. Taschenbuch. x, 180 S. Paperback

ISBN 978 3 540 79738 8

Format (B x L): 12,7 x 19 cm

Gewicht: 178 g

[Weitere Fachgebiete > EDV, Informatik > Informationsverarbeitung > Künstliche Intelligenz, neuronale Netze](#)

Zu [Inhaltsverzeichnis](#)

schnell und portofrei erhältlich bei



Die Online-Fachbuchhandlung beck-shop.de ist spezialisiert auf Fachbücher, insbesondere Recht, Steuern und Wirtschaft. Im Sortiment finden Sie alle Medien (Bücher, Zeitschriften, CDs, eBooks, etc.) aller Verlage. Ergänzt wird das Programm durch Services wie Neuerscheinungsdienst oder Zusammenstellungen von Büchern zu Sonderpreisen. Der Shop führt mehr als 8 Millionen Produkte.

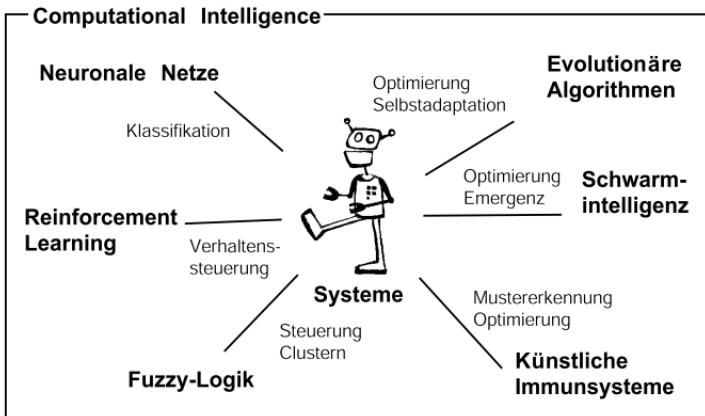
# 1

---

## Überblick Computational Intelligence

Die Entstehungsgeschichte der Computational Intelligence beginnt Mitte des letzten Jahrhunderts. Während die ersten neuronalen Netze 1958 von Rosenblatt entwickelt wurden, beginnt die Geschichte der evolutionären Algorithmen wie auch der Fuzzy-Logik in den Sechzigern. Diese drei Verfahren wurden seitdem nicht nur stetig weiterentwickelt, sie sind auch heute Gegenstand aktueller Forschung und stellen die Grundsäulen der Computational Intelligence dar. Ihre Bedeutung spiegelt sich in zahlreichen Publikationen wider wie den Büchern der Reihe *Studies in Computational Intelligence* von Springer oder den Journalen *IEEE Transactions on Neural Networks*, *IEEE Transactions on Fuzzy Systems* und *IEEE Transactions on Evolutionary Computation* der *IEEE Computational Intelligence Society*.

In diesem Buch werden wir die wichtigsten Techniken kennen lernen, die heute zur Computational Intelligence gezählt werden. Eine Übersicht über diese Verfahren gibt Abbildung 1.1. Einige Methoden sind verwandt mit den Techniken der



**Abbildung 1.1.** Übersicht über die wichtigsten Verfahren der Computational Intelligence.

künstlichen Intelligenz. Diese Verwandtschaftsbeziehung wird im folgenden Abschnitt deutlicher.

## 1.1 Intelligente Informationsverarbeitung

Die Vision intelligenter Maschinen stellte schon in den Pionierzeiten der Informatik die Antriebsfeder für die Entwicklung leistungsfähiger und intelligenter Algorithmen dar. Bereits John von Neumann und Alan Turing hatten die Vision menschenähnlicher Maschinenintelligenz. Seit Beginn der Informatik entstand eine Vielzahl von Forschungsfeldern mit dem Ziel, Algorithmen zu entwickeln, die intelligente Leistungen vollbringen. Dabei bezeichnen wir Informationsverarbeitung als intelligent, wenn die Algorithmen menschenähnliche Leistungen zu vollbringen in der Lage sind. Dazu zählen insbesondere Lernfähigkeit und die Fähigkeit zur Anpassung an sich

verändernde Umstände. Eine weitere Charakterisierung des Begriffs *intelligenter Informationsverarbeitung* erfolgt am leichtesten über die Aufgaben, die mit Hilfe der Methoden gelöst werden. Zu den wichtigsten Aufgaben gehören

- Suche und Optimierung,
- Klassifikation und Gruppierung,
- Erkennung von Mustern,
- Steuerung von Verhalten und komplexe Regelung.

Einige dieser Aufgaben werden wir im Laufe dieses Buches im Zusammenhang mit den vorgestellten Techniken näher kennen lernen. Im Laufe der Dekaden entstand ein unübersichtliches Durcheinander von Begrifflichkeiten wie künstliche Intelligenz, Computational Intelligence, maschinelles Lernen, Biologie, *Soft Computing* oder *Natural Computation* – um nur einige zu nennen. Eine übersichtliche Einordnung ist jedoch bereits mit wenigen Begriffen möglich. Algorithmen zur intelligenten Informationsverarbeitung gehören zu den beiden sich durchaus überschneidenden Hauptgebieten Computational Intelligence und künstliche Intelligenz. Zu diesen beiden Hauptgebieten können die meisten Methoden gezählt werden, die die oben genannten Aufgaben lösen.

Der Begriff **künstliche Intelligenz** hat seinen Ursprung in der berühmten Dartmouth Konferenz im Sommer 1956, die von Pionieren wie Marvin Minsky und Claude Shannon organisiert wurde. Im Rahmen der so genannten *schwachen* künstlichen Intelligenz wird ein Algorithmus als intelligent bezeichnet, wenn er zur Problemlösung menschenähnliche Leistungen vollbringt wie eine Form des Lernens, der Anpassung oder der Schlussfolgerung. Die *starke* künstliche Intelligenz hingegen zielt darauf ab, menschliche Kognition nachzubauen, d.h. insbesondere Bewusstsein, Emotionen und Kreativität zu erschaffen. Innerhalb der künstlichen Intelligenz sind eine Reihe von Methoden angesiedelt, die von **symbolischen** Ansätzen wie Entschei-

dungsbäumen über Logik-basierte Verfahren und fallbasiertes Schließen bis hin zu stochastischen Automaten reichen. Diese Ansätze nutzen meist diskrete Konzepte, mit deren Hilfe logische Aussagen oder Zustände repräsentiert werden können. In diesem Zusammenhang haben sich Aussagen- und Prädikatenlogik sowie die Programmiersprache **Prolog** als unerlässlich erwiesen. Prolog-Programme stellen eine Regelbasis logischer Aussagen dar. Anfragen an diese Regelbasis sind wiederum logische Aussagen. Der Prolog-Interpreter versucht, durch Inferenz zu prüfen, ob die Anfrage-Aussage aus den Fakten der Regelbasis logisch ableitbar ist. Entscheidungsbäume ermöglichen induktives Lernen von Begriffen oder Klassen mit Hilfe einer Menge von Beispielen. Die Datenbeispiele liegen in einer Attribut-Werte-Repräsentation vor. An jedem Knoten des Baumes wird der Wert des Attributes geprüft bis schließlich ein Blatt erreicht wird, das für den gelernten Begriff steht. In der Lernphase wird versucht, aufgrund der Datenbeispiele einen kleinen Baum zu finden, der in der Lage ist, möglichst viele Datenbeispiele zu repräsentieren.

Viele auf diskreten symbolischen Repräsentationen und statischen Zuständen basierende Methoden haben jedoch ihre Grenzen. Für natürliche und alltägliche Phänomene reicht die diskrete Modellierung vieler klassischer Verfahren nicht aus und das Bedürfnis nach fehlertoleranten Methoden entsteht. Verfahren der Computational Intelligence sind ein Ansatz, diesen Bedürfnissen gerecht zu werden. Sie werden auch als **sub-symbolische** Techniken bezeichnet. Denn sie arbeiten häufig mit Repräsentationen unterhalb der Symbolebene, etwa durch Repräsentation eines Zustandes oder einer Probleminstanz mit einer Menge numerischer Werte. Die meisten Techniken der Computational Intelligence zeichnen sich durch folgende Eigenschaften aus:

- **Fehlertoleranz:** Viele Methoden der Computational Intelligence sind fehlertolerant gegenüber falschen, unscharfen

oder unzulässigen Eingaben. Fehlertolerant sind beispielsweise neuronale Netze, die ein Klassifikationsergebnis mit ihren nichtlinearen Schwellwertfunktionen approximieren oder Fuzzy-Systeme, die unscharfe Regeln modellieren.

- **Parallelität:** Mit wenigen Ausnahmen basieren alle Methoden auf einer parallelen Ausführung einer großen Menge von Berechnungseinheiten, z.B. die Population evolutionärer Verfahren oder die Neuronen eines Backpropagation-Netzes. Auch wenn viele Implementierungen auf der sequentiellen Ausführung basieren, ist die Natur der Algorithmen parallel.
- **Einfachheit der Modellierung:** Der Modellierung eines Problems kommt in der Informatik ein hoher Stellenwert zu. Ein durchdacht modelliertes Problem kann mit einem geeigneten Algorithmus in der Regel effizient gelöst werden. Auch Methoden der Computational Intelligence erfordern eine durchdachte Modellierung, ermöglichen jedoch auch mit wenig Modellierungsaufwand schnellen Erfolg. Sie eignen sich daher besonders für schnelle und damit kostengünstige Lösungen.
- **Effiziente Näherung:** Exakte Verfahren finden garantiert die optimale Lösung, ihre Laufzeit verschlechtert sich jedoch oft rapide mit der Größe der Probleminstanz. Bei praktischen Problemen steigt die Anzahl möglicher Zustände in der Regel kombinatorisch an. Dadurch verschlechtert sich die Laufzeit vieler Verfahren und steigt exponentiell. Einige Methoden der Computational Intelligence approximieren die Lösung häufig auf Basis stochastischer Komponenten und finden die optimale Lösung nicht garantiert. Dafür ist ihre Laufzeit auf großen Probleminstanzen für Näherungslösungen akzeptabel.

## 1.2 Naturinspirierte Algorithmen

Die Natur hat im Laufe der Jahrtausende auf der Erde eine große Vielfalt von Problemlösungsstrategien für die Aufgaben Überleben und Fortpflanzung entwickelt. Von diesen Techniken zu lernen heißt, biologische Konzepte in algorithmische Modelle zu übersetzen und auf diese Weise für Problemlösungsprozesse nutzbar zu machen. Interessanter Weise sind viele Verfahren der Computational Intelligence an ein biologisches Vorbild angelehnt und bedienen sich der Sprache der Biologie. Wir werden in dieser Einführung immer wieder auf biologische und natürliche Modelle stoßen, die den Methoden der Computational Intelligence Pate stehen. Viele Algorithmen verdanken ihre Entstehung einer Analyse biologischer Vorgänge. Diese führen zu einem einfachen biologischen Modell, das in eine Rechenvorschrift übersetzt wird. Die folgenden drei Einflussgrößen begleiten den Weg vom biologischen Modell zum anwendungstauglichen Algorithmus:

- **Biologische Inspiration:** Neue und detailliertere Inspirationen durch das biologische Vorbild erweitern das vorhandene einfachere Modell.
- **Theoretisches Modell:** Analysen führen zu einem theoretischen Modell, das zur Vereinheitlichung vorhandener Modelle oder zu Erweiterungen führen kann.
- **Anwendungsspezifische Anpassung:** Bei der Anwendung des Verfahrens ergeben sich problemspezifische Anforderungen, die eine Anpassung und Erweiterung des einfachen Modells erfordern.

Eng verwandt mit den naturinspirierten Rechenvorschriften sind Technologien der **Bionik**, denn auch diese Disziplin konzentriert sich auf naturinspirierte Problemlösungsprozesse. Jedoch bezieht sich die Bionik hauptsächlich auf die Analyse und Nachahmung von *physischen Strukturen* und nicht auf algorithmische Konzepte. Dazu zählt beispielsweise die berühmte

Analyse des Vogelflugs zum Nachbau von Fluggeräten oder die Nachahmung des nanoskopischen Effektes der Wasser abweisenden Lotuspflanze. Dabei kann zwischen zwei Bionik-Ansätzen unterschieden werden. Der Abstraktions-Ansatz analysiert biologische Systeme und abstrahiert ihre zugrunde liegenden Prinzipien, um dafür mögliche Anwendungsgebiete zur Lösung von Problemen zu finden. Umgekehrt geht der Analogie-Ansatz vor. Dort wird ausgehend von konkreten Problemstellungen in der Natur nach Analogien gesucht, um deren Problemlösungsweg zu identifizieren und zu nutzen. Weitere erfolgreiche Beispiele für Bionik-Forschung sind der Klettverschluss, der von Georges de Mestral nach dem Vorbild der Klettfrüchte konstruiert wurde oder die Winglets von Flugzeugflügeln, die die Handschwingen verschiedener Vogelarten imitieren und durch Verursachung von Wirbeln am Flügelende den Energieverbrauch reduzieren.

## 1.3 Übersicht der Verfahren

Der Leser wird in diesem Buch die wichtigsten Konzepte der Computational Intelligence kennen lernen. Die vorgestellten Algorithmen haben sich als leistungsfähige Problemlösungsmethoden etabliert und Einzug in die verschiedensten Bereiche der Informatik und der Ingenieurwissenschaften gehalten. Es folgt ein Kurzabriss der in diesem Buch vorgestellten Techniken.

### Kapitel 2: Evolutionäre Algorithmen

Seit gut 4,6 Milliarden Jahren existiert die Erde, der einzige uns bekannte belebte Himmelskörper. Das Leben auf der Erde ist das Ergebnis eines Prozesses, der auf genetischer Variation und natürlicher Auslese beruht und mit Evolution bezeichnet

wird. Die Grundprinzipien lassen sich in algorithmische Konzepte übersetzen und sind seit den 1960er Jahren als evolutionäre Algorithmen bekannt. Kapitel 2 führt in die Grundlagen der evolutionären Verfahren ein. Sind kaum Informationen über den Suchraum verfügbar oder ist das zu optimierende Problem weder stetig noch differenzierbar, bieten sich die vorgestellten Techniken für Optimierungsprobleme an. Die Vererbung der Eigenschaften mehrerer Lösungen ist als Rekombination bekannt. Die Variation von Lösungen wird als Mutation bezeichnet. Die Auswahl der besten Lösungen verleiht der Suche eine Richtung. Abhängig von der verwendeten Repräsentation und vom Problemtyp existieren unzählige algorithmische Varianten.

### Kapitel 3: Schwarmintelligenz

Das Ganze ist mehr als die Summe seiner Teile. Diese Aussage bewahrheitet sich insbesondere bei natürlichen Schwärmen. Ob in Bienenstaaten, Ameisenkolonien, Fisch- oder Vogelschwärmen, das Individuum hat keine große Bedeutung. Erst in seiner Interaktion mit der Masse weiterer meist gleichartiger Artgenossen kommt es zu einem intelligenten und emergenten Zusammenspiel, das als Schwarmintelligenz bekannt ist. Auch dieses natürliche Vorbild wurde erfolgreich algorithmisch übersetzt. Die Partikelschwarmoptimierung eignet sich ähnlich wie die evolutionären Verfahren für die Approximation von Lösungen für Optimierungsprobleme, über die kaum Wissen zur Verfügung steht. Ameisenalgorithmen eignen sich für kombinatorische Optimierungsprobleme, bei denen zusätzliches heuristisches Wissen über Teillösungen bereit steht. Die Lösungsqualität wird über Pheromone auf die einzelnen Komponenten der Lösung verteilt. Die Ähnlichkeit zwischen Verfahren der Schwarmintelligenz und der evolutionären Algorithmen geht weit über ihre gemeinsame Aufgabe zu optimieren hinaus.

Beide Techniken sind populationsbasiert und verwenden stochastische Operatoren zur Variation ihrer Lösungskandidaten.

## Kapitel 4: Künstliche Immunsysteme

Auch die künstlichen Immunsysteme sind mit den evolutionären Verfahren und der Schwarmintelligenz verwandt. Künstliche Immunsysteme imitieren die Konzepte natürlicher Immunsysteme bei der Bekämpfung von Antigenen wie Bakterien oder Viren. Auch Immunsysteme basieren auf einer Population von Akteuren. Zu ihren wesentlichen funktionalen Prinzipien gehören vor allem Selektionsoperatoren wie positive oder klonale Selektion. Diese ermöglichen die Erkennung und somit die Anpassung an Antigene. Ein typisches Einsatzgebiet künstlicher Immunsysteme sind Mustererkennungsaufgaben. Sie dienen aber auch mit ihren funktionalen Komponenten als Vorlage für eine Vielzahl weiterer verteilter Informationsverarbeitungsmodelle.

## Kapitel 5: Fuzzy-Logik

Der Modellierung unscharfer Begriffe und Inferenz auf unscharfen Informationen widmet sich das Gebiet der Fuzzy-Logik. Unschärfe ist in der Regel ein Kennzeichen menschlicher Sprache. Die Methoden der Fuzzy-Logik stehen Pate für kognitionsähnliche unscharfe Modellierung. Konzepte und logische Aussagen können nicht allein nur zwei, sondern eine ganze *Menge von Wahrheitswerten* annehmen. Fuzzy-Logik ermöglicht auch, Schlussfolgerungen auf diesen unscharfen Aussagen zu ziehen. Auf diese Weise werden Fuzzy-Regler konstruiert, die sich in der Praxis als leistungsfähige Regler etablieren konnten. Fuzzy-Regelbasen können für einfache Regler, aber auch für die Steuerung komplexer Systeme eingesetzt werden. Außerdem werden wir eine fuzzifizierte Variante des Verfahrens k-Means zum Clustern von Daten kennen lernen.

## Kapitel 6: Reinforcement Learning

Ein weiteres Prinzip menschlicher Kognition ist neben unscharfer Informationsverarbeitung das Prinzip von Belohnung und Bestrafung. Nützliches Verhalten wird entweder von außen durch einen *Lehrer* oder intern durch entsprechende Bereiche des Gehirns belohnt, während unerwünschtes oder nicht zielgerichtetes Verhalten häufig bestraft wird. Dieses Prinzip für künstliche Systeme nutzbar zu machen, hat zur Entwicklung der Verfahren geführt, die unter dem Oberbegriff *Reinforcement Learning* bekannt sind. Mit Hilfe von *Reinforcement Learning* kann optimales Verhalten erlernt werden, indem jedem Zustand ein Wert zugewiesen wird. Value Iteration basiert auf dynamischer Programmierung und ist geeignet, die optimale Strategie zu finden, wenn der Zustandsraum vollständig bekannt ist. Bei unbekannten Zustandsräumen ermöglicht *Q-Lernen*, Zustands-Aktions-Paare zu bewerten, ohne über das vollständige Markov-Modell zu verfügen. Währenddessen ermöglicht *Temporal Difference Learning*, den Wert von Zustands-Aktions-Paaren über mehrere Zustände hinweg genauer zu approximieren.

## Kapitel 7: Neuronale Netze

Den biologischen Grundlagen natürlicher Kognition widmet sich das Gebiet der neuronalen Netze. Diese modellieren verschiedene in natürlichen neuronalen Netzen identifizierte Prinzipien der Informationsverarbeitung und ermöglichen so, eine ganze Reihe von Klassifikations- und Optimierungsaufgaben zu bewältigen. Das Perzeptron von Rosenblatt ist ein einfacher Algorithmus, um Daten zu klassifizieren. Ein einschichtiges *Perzepron* ist auf lineare Separation begrenzt. Mehrschichtige Perzeptrons sind in der Lage, beliebige Daten zu klassifizieren. Einen mächtigeren Klassifikator stellt das *Backpropagation*-Netz dar. Es basiert auf einem Verfahren, das Gradientenab-

stieg in der Fehlerfunktion ähnelt. Selbstorganisierende Karten ermöglichen die unüberwachte Abbildung hochdimensionaler Daten auf niedrigdimensionale Vektoren. Da benachbarte Daten denselben Vektoren zugeordnet werden, können sie auch als Clusterverfahren eingesetzt werden.

Neben der Vorstellung der Begrifflichkeiten jeder Methodenklasse steht eine kompakte Darstellung der grundlegenden Verfahrens- und Denkweisen im Vordergrund dieses Buches. Jedes Kapitel enthält ein beispielhaftes Anwendungsszenario und endet mit einer Literaturliste. Bei dem Anwendungsszenario handelt es sich meist um die Vorstellung einer aktuellen Arbeit, deren Auswahl in vollem Maße der Subjektivität des Autors obliegt und hauptsächlich das Ziel verfolgt, das Interesse des Lesers für das jeweilige Gebiet zu wecken. Anwendungsbeispiel und Literaturliste dienen weiterhin als Ausgangspunkt für die eigene Literaturrecherche.

## Literaturempfehlung

GÖRZ, G.: *Handbuch der Künstlichen Intelligenz*. Oldenbourg, 2003, [19].

KONAR, A.: *Computational Intelligence*. Springer, 2005, [33].

LIPPE, W.-M.: *Soft-Computing*. Springer, 2006, [36].

MITCHELL, T. M.: *Machine Learning*. McGraw-Hill, 1997, [38].

RUSSEL, S.; NORVIG, P.: *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995, [47].

RUTKOWSKI, L.: *Computational Intelligence - Methods and Techniques*. Springer, 2008, [48].