Uwe Lorenz

Reinforcement Learning

Aktuelle Ansätze verstehen – mit Beispielen in Java und Greenfoot

2. Auflage

MOREMEDIA

Springer Vieweg

Reinforcement Learning

Uwe Lorenz

Reinforcement Learning

Aktuelle Ansätze verstehen – mit Beispielen in Java und Greenfoot

2. Auflage

Springer Vieweg

Uwe Lorenz
Neckargemünd, Baden-Württemberg
Deutschland

ISBN 978-3-662-68310-1 ISBN 978-3-662-68311-8 (eBook)
https://doi.org/10.1007/978-3-662-68311-8

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

© Der/die Herausgeber bzw. der/die Autor(en), exklusiv lizenziert an Springer-Verlag GmbH, DE, ein Teil von Springer Nature 2020, 2024

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsgesetz zugelassen ist, bedarf der vorherigen Zustimmung des Verlags. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.
Die Wiedergabe von allgemein beschreibenden Bezeichnungen, Marken, Unternehmensnamen etc. in diesem Werk bedeutet nicht, dass diese frei durch jedermann benutzt werden dürfen. Die Berechtigung zur Benutzung unterliegt, auch ohne gesonderten Hinweis hierzu, den Regeln des Markenrechts. Die Rechte des jeweiligen Zeicheninhabers sind zu beachten.
Der Verlag, die Autoren und die Herausgeber gehen davon aus, dass die Angaben und Informationen in diesem Werk zum Zeitpunkt der Veröffentlichung vollständig und korrekt sind. Weder der Verlag noch die Autoren oder die Herausgeber übernehmen, ausdrücklich oder implizit, Gewähr für den Inhalt des Werkes, etwaige Fehler oder Äußerungen. Der Verlag bleibt im Hinblick auf geografische Zuordnungen und Gebietsbezeichnungen in veröffentlichten Karten und Institutionsadressen neutral.

Planung/Lektorat: David Imgrund
Springer Vieweg ist ein Imprint der eingetragenen Gesellschaft Springer-Verlag GmbH, DE und ist ein Teil von Springer Nature.
Die Anschrift der Gesellschaft ist: Heidelberger Platz 3, 14197 Berlin, Germany

Das Papier dieses Produkts ist recycelbar.

Vorwort

Man muss die Dinge so einfach wie möglich machen. Aber nicht einfacher.

(A. Einstein)

Ziel des Buches ist es, nicht nur eine lose Auflistung gängiger Ansätze des „Verstärkenden Lernens“, engl. „Reinforcement Learning“ (RL) zu liefern, sondern auch einen inhaltlich zusammenhängenden Überblick über dieses faszinierende Gebiet der Künstlichen Intelligenz zu geben. Gleichzeitig sollen die Konzepte einem möglichst großen Leserkreis aufgeschlossen und z. B. auch Impulse für den Schulunterricht ermöglicht werden.

Wie kann dieser Spagat gelingen? Notwendig hierfür ist es, einen möglichst umfassenden Grundriss zu zeichnen, der zwar die wesentlichen Ideen beinhaltet, dabei aber von handwerklichen Details, die nur in spezifischen Anwendungskontexten relevant sind, absieht. Eine vereinfachte Darstellung ist nicht falsch, wenn sie innere Konsistenz und Zweckmäßigkeit aufweist. Im Sinne der „Spirale des Begreifens“ kann ein solcher Grundriss als Ausgangspunkt für tiefere und detailiertere Einsichten, weitere Untersuchungen und auch diverse praktische Anwendungen – auch mit Hochleistungswerkzeugen – dienen. Um dieses Buch zu verstehen, sollten die Mittel der höheren Schulmathematik ausreichen. Zudem benötigen Sie einige grundlegende Kenntnisse in der Programmiersprache Java.

Das Forschungsgebiet „Künstliche Intelligenz“ war Anfang der 2000er Jahre, in der Zeit meines Studiums in Berlin, in einem fundamentalen Umbruch. Der Ansatz der „good old fashioned artificial intelligence“ (GOFAI), der davon ausgeht, dass Kognition durch Suche in wissensbasierten Modellen der Außenwelt entsteht, war in einer Krise, es war noch die Zeit des sogenannten „KI-Winters“. Gleichzeitig sorgte die Entwicklung der allgemein zugänglichen Rechenleistung bereits dafür, dass mit künstlichen neuronalen Netzen immer erstaunlichere Resultate produziert wurden. Es gab eine Debatte zwischen „Konnektionismus“ vs. „Symbolischer K.I.“, also die Frage nach der prinzipiellen Leistungsfähigkeit von verteilten, subsymbolischen Repräsentationen einerseits und formalisiertem Spezialwissen, aus dem logisch geschlossen wird, andererseits.

Der fundamentale Unterschied zwischen den Ansätzen besteht nicht nur in der Art der Repräsentation von Wissen. Es geht vielmehr um einen Unterschied in der Stellung der Basiskonzepte „Modell“ (Abbildung, Darstellung für einen bestimmten Zweck) und „Verhalten“. Welches der beiden Prinzipien sollte eine dominante Position einnehmen? Die Bedeutung des verhaltensbasierten Ansatzes für das Thema KI wurde mir bei dem Versuch klar, einen objektiven Begriff von „Ähnlichkeit“ zu bilden, im Zusammenhang mit Forschungen an der Mustererkennung mit sogenannten „tiefen neuronalen Netzen“ Anfang der 2000er Jahre. Diese Bemühungen führten zu einer grundlegenden Einsicht: „Ähnlichkeit“ und somit auch „Klassifikation“ allgemein sind eigentlich nichts Anderes als evolutionär gewachsene Mittel, um die für das Leben notwendigen Unterscheidungen treffen zu können. So etwas wie eine „zweckfreie Beschreibung“ ist im Prinzip unmöglich. Dies führte mich zu der Idee, dass es bei den Bemühungen im Feld der Künstlichen Intelligenz nicht zuerst darum gehen kann, logisch „denkende“, sondern sich zweckmäßig verhaltende Maschinen zu konstruieren. Die weiteren Entwicklungen bestätigten diese Überlegungen.

Bei verhaltensbasierten Systemen wird Handlungskompetenz aus der Differenz von Vorhersage und Beobachtung generiert. Bei „Trial and Error“ können sehr viele Beobachtungen entstehen, deren Bedeutung sich erst zu einem späten Zeitpunkt mehr oder weniger zufällig herausstellt. Eine zentrale technische Herausforderung beim Reinforcement Learning ist es, „Beobachtungen“ die hierbei entstehen, für die Optimierung der Agentensteuerung nutzbar zu machen. In den letzten Jahren wurden auf diesem Gebiet große Fortschritte gemacht und das gesamte Gebiet entwickelt sich weiterhin außerordentlich dynamisch.

Meiner Schwester Ulrike alias „HiroNoUnmei“ (https://www.patreon.com/hironounmei; 27.12.2022) möchte ich noch ganz herzlich eigens manuell gefertigten lustigen Hamster-Illustrationen an den jeweiligen Kapitelanfängen danken. Über originelle Aufträge freut sie sich immer sehr.

Zu den Begleitmaterialien (Java-Programme, Erklärvideos usw.) gelangen Sie über die Produktseite des Buchs (https://github.com/sn-code-inside/Reinforcement-Learning) und über die Facebook-Seite (https://www.facebook.com/ReinforcementLearningJava; 27.12.2022). Posten Sie gerne inhaltliche Beiträge zum Thema oder interessante Ergebnisse. Dort finden Sie ggf. auch eine Möglichkeit Verständnisfragen zu stellen oder offen gebliebene Punkte anzusprechen.

Neckargemünd Uwe Lorenz
Juni 2023

Einleitung

Verständnis wächst mit aktiver Auseinandersetzung: Etwas zu ‚machen', zu beherrschen, bedeutet zugleich besseres Verstehen. Angewandt auf die Erforschung geistiger Prozesse führt das auf die Nachbildung intelligenten Verhaltens mit Maschinen.

(H.-D. Burkhardt[1])

Zusammenfassung

In diesem einleitenden Abschnitt wird dargestellt, worum es in diesem Buch geht und an wen es sich richtet: Das Thema „Reinforcement Learning" als ein spannendes Teilgebiet der Künstlichen Intelligenz bzw. des Maschinellen Lernens soll in einer Form dargestellt werden, die es Einsteigern ermöglicht, die wichtigsten Ansätze und einige der zentralen Algorithmen schnell zu verstehen und auch selbst damit zu experimentieren. Über einige „philosophische" Fragestellungen oder Kritiken am Forschungsgebiet der KI wird kurz reflektiert.

Dieses Buch ist vielleicht nicht ganz ungefährlich, denn es geht in ihm um lernfähige künstliche Agenten. Auf dem Digital-Festival „South by Southwest" im US-Bundesstaat Texas sagte Elon Musk im Jahr 2018 „Künstliche Intelligenz ist sehr viel gefährlicher als Atomwaffen". Vielleicht ist dies etwas dick aufgetragen, allerdings ist es sicherlich von Vorteil, wenn möglichst viele Menschen verstehen und beurteilen können, wie diese Technik funktioniert. Das ermöglicht nicht nur zu beurteilen, was die Technik leisten kann und sollte, sondern auch, ihren weiteren Entwicklungsweg mitzugestalten. Der Versuch einer Einhegung von verwertbarem Wissen wäre in der Welt von heute sicherlich auch ein vergebliches Unterfangen. Künstliche Intelligenz und disruptive Anwendungen wie „ChatGPT" sind derzeit in aller Munde. Das „Verstärkende Lernen" hat hierbei

[1] Prof. Dr. Hans-Dieter Burkhard war einer meiner ehemaligen Hochschullehrer an der Humboldt-Universität Berlin, – 2004, 2005 und 2008 Weltmeister in der Four-Legged Robots League beim RoboCup mit dem „German Team".

sogar eine gewisse Rolle gespielt, um das Verhalten des Chatbots an menschliche Erwartungen anzupassen. Der Ansatz des Reinforcement Learning ist jedoch nicht aus dem Versuch entstanden, Texte zu verarbeiten. Vielmehr geht es in diesem Ansatz um autonom agierende Maschinen, die ähnlich wie biologische Lebewesen in einer bestimmten Umgebung mit Problemen konfrontiert werden und durch aktive Prozesse ihr Verhalten verbessern.

Beim RL handelt es sich um einen der faszinierendsten Bereiche des Maschinellen Lernens, welcher im deutschsprachigen Raum oft noch wenig behandelt wird, obwohl immer wieder spektakuläre Erfolgsmeldungen aus diesem Gebiet der Künstlichen Intelligenz nicht nur das Fachpublikum, sondern auch die breite mediale Öffentlichkeit erreichen. Einige Beispiele: Eine in der Geschichte der Künstlichen Intelligenz mit am längsten untersuchten Domäne ist die des Schachspiels. Es gelang schon vor geraumer Zeit, Programme zu schreiben, die menschliche Champions schlagen konnten, diese nutzten allerdings ausgefeilte, spezialisierte Suchtechniken und handgefertigte Auswertungsfunktionen sowie enorme Datenbanken mit archivierten Spielzügen. Programme seit „Alpha Zero“ von „Google DeepMind“ dagegen können innerhalb weniger Stunden allein durch Lernen aus Spielen gegen sich selbst übermenschliche Leistungen erreichen (Silver et al. 2017). Es übertrifft schließlich auch die bislang besten Programme, welche die zuvor erwähnten Methoden nutzten, deutlich. Spektakulär hierbei ist auch, dass das System nicht nur auf ein einzelnes Spiel wie Schach festgelegt ist, sondern in der Lage ist, alle möglichen Brettspiele selbsttätig zu erlernen. Darunter zählt auch das wohl in vielfacher Hinsicht komplexeste Brettspiel „Go“, welches in Asien schon seit tausenden von Jahren gespielt wird und eigentlich eine Art „intuitive“ Interpretation der Situation des Spielbretts erfordert. Alpha Zero kann sich an vielfältige hochkomplexe Strategiespiele überaus erfolgreich anpassen und benötigt dafür kein weiteres menschliches Wissen, – es reichen hierfür allein die Spielregeln. Das System wird damit zu einer Art universellem Brettspiel-Lösungssystem.

Wie kann diese Maschine solche Erfolge erzielen, ohne auf das in Jahrtausenden gesammelte menschliche Wissen über Spiele zurückzugreifen? Bis vor kurzem bestand noch große Übereinstimmung darin, dass ein „intuitives“ Spiel mit einer derartig großen Zahl von Zuständen wie „Go“ in absehbarer Zeit nicht von seriellen Computern zu bewältigen ist. Ähnlich spektakulär sind Durchbrüche beim erfolgreichen Agieren in dynamischen Umgebungen, z. B. von „Deep Q Networks“ beim autonomen Erlernen beliebiger Atari-Arcade-Spiele (Kavukcuoglu et al. 2015) oder die Ergebnisse im Bereich der Robotik, wo Systeme durch selbständiges Ausprobieren lernen, komplexe Bewegungen wie Greifen, Laufen, Springen etc. erfolgreich auszuführen und Aufgaben in vorbereiteten Arenen zu meistern, wie sie z. B. in den zahlreichen Robotik-Wettbewerben gestellt werden, die es mittlerweile für alle möglichen Anforderungsniveaus und Altersgruppen gibt. Eine große Rolle spielen hierbei sogenannte „tiefe künstliche neuronale Netze“ mit besonderen Fähigkeiten bei der Generalisierung.

In letzter Zeit wurden aufwendige Machine Learning Frameworks von einschlägigen Playern teilweise kostenlos zur Verfügung gestellt. Warum schenken Unternehmen wie Google, OpenAI, Amazon usw. solche aufwendigen Produkte der Allgemeinheit? Es ist

anzunehmen, dass es auch darum geht, Standards zu definieren und somit auch Abhängigkeiten zu schaffen. Philanthropische Großzügigkeit ist bei Kapitalgesellschaften sicherlich nicht als primärer Beweggrund anzunehmen. Das Buch möchte auch Mut machen, das „Räderwerk“, das sich hinter den Frameworks verbirgt, genauer anzuschauen und von Grund auf zu verstehen, und zeigen, dass dies möglich ist, auch wenn man nicht bspw. mit der Programmiersprache Python sozialisiert worden ist.

Konkrete Umsetzungen des Reinforcement Learning erscheinen oft recht kompliziert. Die „Lernvorgänge“ hängen von vielen Parametern und praktischen Gegebenheiten ab. Sie benötigen viel Rechenzeit und gleichzeitig ist der Erfolg oft ungewiss. Die Ideen der Algorithmen, die hinter den lernfähigen Agenten stecken, sind jedoch meist sehr anschaulich und leicht verständlich, zudem werden wir in den Experimenten Live-Visualisierungen einsetzen, mit denen der Lernfortschritt und der aktuell erreichte Lernstand des Agenten beobachtet werden kann. Das Thema „Verstärkendes Lernen“ soll hier in einer Form präsentiert werden, welche auch Einsteigern zügig die wichtigsten Ansätze und zentrale Algorithmen vermittelt sowie eigene interessante Experimente ermöglicht.

Dabei kommen Werkzeuge, wie sie bspw. in Einsteigerkursen oder im Programmierunterricht verwendet werden, zur Anwendung. Sie werden im Buch auch Anregungen für Unterricht und Lehre finden. Es soll in erster Linie nicht um die Bedienung einer Hochleistungsblackbox gehen, sondern um das Verstehen, Begreifen, Beurteilen und vielleicht auch das innovative Weiterentwickeln der Algorithmen in eigenen Versuchen. Ein Auto zu fahren ist das eine, zu verstehen, wie der Motor eines Autos funktioniert ist eine andere Sache. Obwohl beides eng verknüpft ist: Zum einen erfordert das Fahren gewisse Kenntnisse in der Funktionsweise eines Autos und umgekehrt bestimmt der Konstrukteur eines Autos auch, wie das Fahrzeug gefahren wird. Im übertragenen Sinne werden wir jeweils nach einigen theoretischen Vorüberlegungen einige „Motoren“ und „Seifenkisten“ selbst bauen und ausprobieren, um zu begreifen, wie die Technik funktioniert. Darüber hinaus wird aber auch das Fahren mit marktreifen Produkten erleichtert. Es ist ja auch bei weitem nicht so, dass die Nutzung von fertigen Bibliotheken zum Reinforcement Learning auf Anhieb problemlos funktioniert.

„Schachbrettwelten“, sogenannte Gridworlds, wie Abb. 1, spielen in Einführungskursen in die Programmierung eine große Rolle. Hierbei handelt es sich jedoch nicht um Brettspiele, sondern um zweidimensionale Raster, in denen sich diverse Objekte, also „Schätze“, „Fallen“, „Mauern“ und ähnliches, sowie bewegliche Figuren befinden.

Weiterhin ist in der Lehre robotische Hardware, die in der Regel mit einem Differentialantrieb und einigen einfachen Sensoren, wie z. B. Berührungs- oder Farbsensoren, ausgestattet ist, weit verbreitet, bspw. mit Bausätzen des Spielzeugherstellers LEGO, wie in Abb. 2. zu sehen, Fischertechnik oder OpenSource-Projekten wie „Makeblock“.

Mit solchen Hilfsmitteln werden algorithmische Grundstrukturen, wie Sequenzen, bedingte Anweisungen oder Wiederholungsschleifen, gelehrt. Allerdings ist die Funktionsweise der elementaren algorithmischen Strukturen damit recht schnell erkannt. Die lustigen Figuren in den Schachbrettwelten, wie der Kara-Käfer oder der Java-Hamster oder auch die Roboter-Kreationen, wecken die Motivation, „wirklich“ intelligentes Verhalten

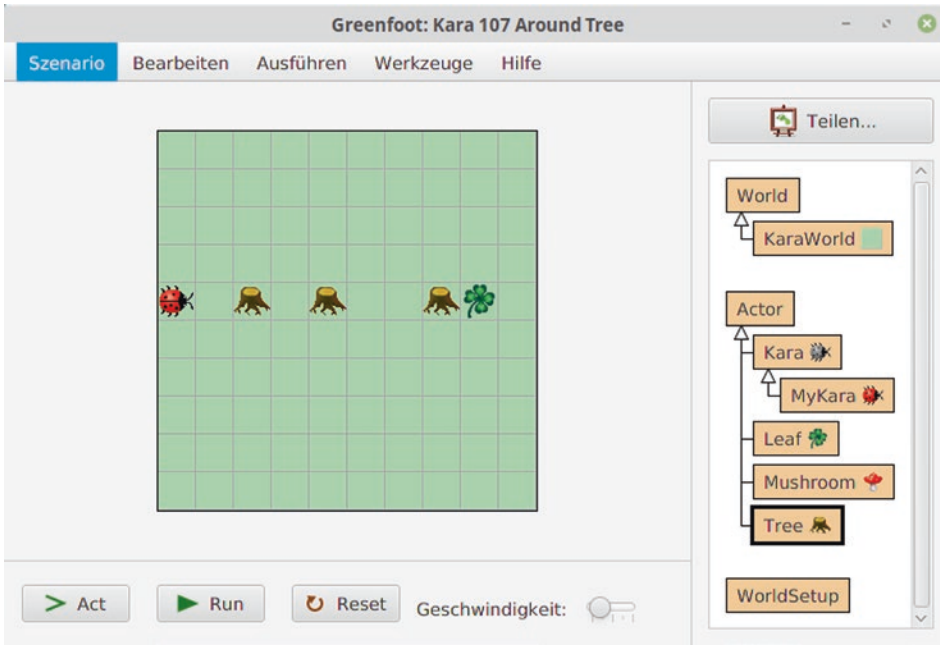

Abb. 1 Gridworld „Kara“

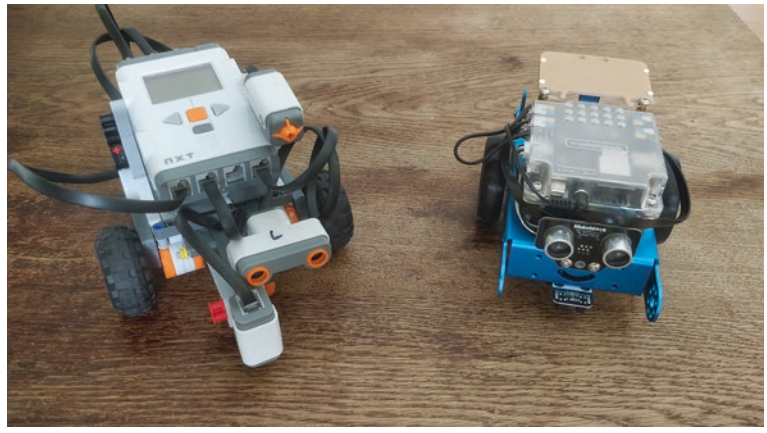

Abb. 2 Roboter mit Differentialantrieb

zu programmieren, allerdings bleibt auf der Ebene der algorithmischen Grundstrukturen das Verhalten solcher „Roboter“ i.d.R. sehr mechanisch und kaum flexibel – intelligentes Verhalten sieht anders aus.

Interessanterweise ist die akademische Standardliteratur zum Reinforcement Learning ebenfalls voll mit solchen „Gridworlds“ und einfachen Robotern. Denn diese bieten klare

Vorteile: Zum einen sind sie komplex genug für interessante Experimente und sehr anschaulich, zum anderen sind sie aber wegen ihrer Einfachheit gut durchschaubar und erlauben eine mathematische Durchdringung. In diesem einführenden Lehr- und Experimentierbuch werden uns diese einfachen „Welten“ zunächst einen anschaulichen Zugang zu den Algorithmen der lernfähigen Agenten ermöglichen – in späteren Kapiteln werden wir uns dann auch mit komplexeren, kontinuierlichen und dynamischen Szenarien beschäftigen.

Das Buch richtet sich an Lernende oder Interessierte, die sich mit diesem Gebiet der Künstlichen Intelligenz beschäftigen möchten (oder müssen). Darüber hinaus ist es auch für Lehrpersonen oder Techniker gedacht, die sich weiterbilden und anschauliche Übungen mit ihren Schülerinnen und Schülern oder Studierenden bzw. eigene Experimente durchführen möchten.

Dieses Buch hat die Besonderheit, dass die Algorithmen nicht in der Programmiersprache Python, sondern in der bei Softwareentwicklern und auch in der Lehre vorallem im Zusammenhang mit objektorientiertem Programmieren verbreiteten Sprache Java vorgestellt werden. Bei den meisten, die in der Künstlichen-Intelligenz-Szene der Anfang 2000er Jahre sozialisiert wurden, stellt Java oft auch noch so eine Art „Muttersprache“ dar. In letzter Zeit sind auch große Player wie Amazon oder Linkedin mit interessanten, kostenfreien Tools für die Java-Community auf den Markt gekommen.

Die Schlagwörter „Künstliche Intelligenz“ und „Maschinelles Lernen“ sind derzeit in aller Munde. Wie verhalten sich diese Begriffe zueinander? „Künstliche Intelligenz“ ist der wesentlich umfassendere Begriff. Hierunter werden z. B. auch regelbasierte Systeme der GOFAI („Good Old-Fashioned Artificial Intelligence“) gefasst, die nicht nur (altmodische) Schachprogramme, sondern zudem auch bestimmte Sprachassistenten, regelbasierte Chatbots und Ähnliches hervorgebracht haben. In solchen old-fashioned „Expertensystemen“ ist „Wissen“ symbolisch repräsentiert und wird mit Produktionsregeln, ähnlich von Wenn-Dann-Anweisungen, miteinander verbunden. Lernen wird hierbei vor allem als eine Aufnahme und Verarbeitung von symbolischem „Wissen“ verstanden.

Beim „Maschinellen Lernen“ werden Softwarefunktionen durch eine iterative, datengetriebene Optimierung („Trainingsprozess“) erzeugt, häufig wird dies aktuell mit dem überwachten Training von Mustererkennern insbesondere mit künstlichen neuronalen Netzen assoziiert. Diese Technologien besitzen eine große praktische Bedeutung und werden insbesondere von US-Unternehmen schon seit Jahrzehnten mit gewaltigem wirtschaftlichem Erfolg angewendet.

„Reinforcement Learning“ wird auch dem Bereich des Maschinellen Lernens zugeordnet, hat aber eine eigene Perspektive auf lernende Systeme, bei der z. B. die Einbettung des Lernsystems in seine Umwelt mitberücksichtigt wird. Beim Reinforcement Learning geht es im Kern darum, aktiv lernende, d. h. autonom agierende Agenten zu bauen, die sich in ihrer Umgebung zunehmend erfolgreich verhalten, wobei sie sich durch Versuch und Irrtum verbessern. Hierfür sind regelbasierte oder auch konnektionistische Implementationen denkbar. Die Herausforderung besteht darin, während des selbsttätigen Erkundungsprozesses innere Strukturen aufzubauen, die das Agentenverhalten immer zweckmäßiger steuern. Dazu werden Erkenntnisse aus verschiedenen

Bereichen der Künstlichen Intelligenz, aber auch interdisziplinäres Wissen miteinander kombiniert.

Dieser interdisziplinäre Charakter des Ansatzes ist nicht verwunderlich, wenn man bedenkt, dass das Szenario des Reinforcement Learning – erfolgreiches Handeln innerhalb eines Umweltsystems – sehr stark den biologischen Wurzeln der Kognition entspricht, wo alle kognitiven Fähigkeiten entsprechend den natürlichen Anforderungen kombiniert angewendet werden. Der biologische Ursprung des kognitiven Apparates spielt im RL-Ansatz eine vergleichsweise große Rolle. Begriffe wie „Situiertheit“ („situated AI approach“) und im weiteren Sinne auch „Embodiment“, also die Beschaffenheit des „Körpers“ und der sensorischen und motorischen Fähigkeiten, kommen ins Spiel. Teilweise wird versucht, auch in Abgrenzung dazu, die Funktionsweise biologischer kognitiver Systeme besser zu verstehen oder die Dynamik lebender Systeme künstlich nachzubilden und zu simulieren, bis hin zu Experimenten mit „künstlichem Leben“.

„Künstliche Intelligenz“-Forschung ist kein junges Gebiet. Sie begann mindestens schon im mechanischen Zeitalter, also lange vor Turings Zeiten, z. B. mit den mechanischen Rechenmaschinen von Leibniz oder Pascal, auf dem Pfad der erwähnten „aktiven Auseinandersetzung“ mit geistigen Prozessen und der entsprechenden maschinellen Nachbildungsversuche. Hier wurden manche kuriose Konstruktionen hervorgebracht, vgl. Abb. 3, so manche Irrwege beschritten, aber auch zahlreiche wichtige Erkenntnisse gesammelt, nicht nur in technischer, sondern auch in „philosophischer“ Hinsicht.

Das Thema künstliche Intelligenz weckt auch Ängste, nicht nur wegen der Gefahren, die die neue Technologie mit sich bringt. Manche haben auch grundsätzliche Kritik, wie z. B. Julian Nida-Rümelin, der mit dem Aufkommen eines „Maschinenmenschen“ das Ende des aufgeklärten Humanismus kommen sieht. Oder Weizenbaums klassische

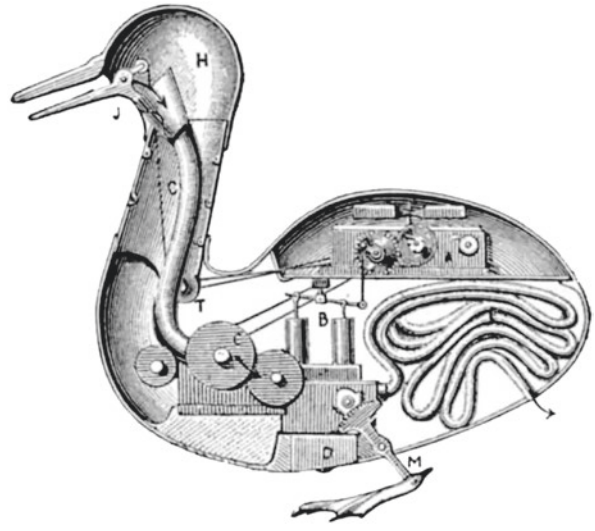

Abb. 3 Die mechanische Ente von Vaucanson (1738) konnte mit den Flügeln flattern, schnattern und Wasser trinken. Sie hatte sogar einen künstlichen Verdauungsapparat: Körner, die von ihr aufgepickt wurden, „verdaute“ sie in einer chemischen Reaktion in einem künstlichen Darm und schied sie daraufhin in naturgetreuer Konsistenz aus. (Quelle: Wikipedia)

Kritik „Die Macht der Computer und die Ohnmacht der Vernunft“, wo er den simplen Intelligenzbegriff der KI-Forscher kritisiert und die Idee von „Künstlicher Intelligenz“ als „perverse, grandiose Phantasie“ bezeichnet. Auch finden wir auf der anderen Seite „KI-Propheten“ die manches magere Ergebnis überhöhen, mystifizieren, um ihre Zunft zu beweihräuchern. Das Buch möchte dazu beitragen, verschiedene Aspekte dieser Technologie besser zu verstehen, die Grenzen, aber auch die gewaltigen Potenziale, realistischer einzuschätzen und mystifizierende Aussagen, aber auch kritische Anmerkungen besser zu beurteilen. Auf den letzten Seiten des Buches sollen noch einmal grundsätzliche Fragen, Leitideen im Wandel der Zeit und Perspektiven aufgegriffen werden. Dabei zeigt sich, dass grundsätzliche Kritik die Entwicklung der KI nicht aufhalten konnte, sondern bislang im Gegenteil oft zu interessanten Fragestellungen und wichtigen Weiterentwicklungen geführt hat.

Inhaltsverzeichnis

1 Verstärkendes Lernen als Teilgebiet des Maschinellen Lernens 1
1.1 Maschinelles Lernen als automatische Verarbeitung von Feedback aus der Umwelt 2
1.2 Verfahren des maschinellen Lernens 3
1.3 Reinforcement Learning mit Java 8
Literatur. 12

2 Grundbegriffe des Bestärkenden Lernens 13
2.1 Agenten 14
2.2 Die Steuerung des Agentensystems („Policy“) 16
2.3 Die Bewertung von Zuständen und Aktionen (Q-Funktion, Bellman-Gleichung). 18
Literatur. 20

3 Optimal entscheiden in einer bekannten Umwelt 21
3.1 Zustandsbewertung 23
3.1.1 Zielorientierte Zustandsbewertung (Rückwärtsinduktion) 23
3.1.2 Taktikbasierte Zustandsbewertung (Belohnungsvorhersage). 32
3.2 Taktiksuche 35
3.2.1 Taktikoptimierung 36
3.2.2 Policy-Iteration. 38
3.3 Optimale Taktik in einem Brettspiel-Szenario. 42
3.4 Zusammenfassung 46
Literatur. 48

4 Entscheiden und Lernen in einer unbekannten Umwelt. 49
4.1 Exploration vs. Exploitation 50
4.2 Rückwirkende Verarbeitung von Erfahrungen („Modellfreies Reinforcement Learning“) 53
4.2.1 Zielorientiertes Lernen („value-based“). 54
4.2.2 Taktiksuche 73

4.2.3 Kombinierte Methoden (Actor-Critic) 94
4.3 Erkunden mit vorausschauenden Simulationen („Modellbasiertes Reinforcement Learning“) 108
4.3.1 Dyna-Q 109
4.3.2 Monte-Carlo Rollout 115
4.3.3 Künstliche Neugier 121
4.3.4 Monte-Carlo-Baumsuche (MCTS) 125
4.3.5 Bemerkungen zum Intelligenzbegriff 131
4.4 Systematik der Lernverfahren 134
Literatur 135

5 Schätzer für Zustandsbewertung und Aktionsauswahl 137
5.1 Künstliche neuronale Netze 139
5.1.1 Mustererkennung mit dem Perzeptron 143
5.1.2 Die Anpassungsfähigkeit von künstlichen Neuronalen Netzen 146
5.1.3 Backpropagation-Lernen 162
5.1.4 Regression mit Multilayer Perzeptrons 165
5.2 Generalisierende Zustandsbewertung 169
5.3 Neuronale Schätzer für die Aktionsauswahl 182
5.3.1 Policy Gradient mit neuronalen Netzen 182
5.3.2 Proximal Policy Optimization 185
5.3.3 Evolutionäre Strategie mit einer neuronalen Policy 188
Literatur 192

6 Leitbilder in der Künstlichen Intelligenz 195
6.1 Grundvorstellungen im Wandel 196
6.2 Über das Verhältnis von Mensch und Künstlicher Intelligenz 201
Literatur 204

Verstärkendes Lernen als Teilgebiet des Maschinellen Lernens

1

© Der/die Autor(en), exklusiv lizenziert an Springer-Verlag GmbH, DE, ein Teil von Springer Nature 2024

U. Lorenz, *Reinforcement Learning,* https://doi.org/10.1007/978-3-662-68311-8_1

„In der Evolution gilt das klug sein nichts, wenn es nicht zu klugem Handeln führt.“ (Tomasello 2014) (Michael Tomasello)

Zusammenfassung

In diesem Kapitel geht es um einen agenten- oder verhaltensorientierten Begriff des Maschinellen Lernens und eine allgemeine Einordnung des Reinforcement Learnings in das Gebiet. Es wird ein grober Überblick über die verschiedenen Prinzipien des Maschinellen Lernens gegeben und erklärt, wodurch sie sich vom Ansatz her unterscheiden. Im Anschluss wird auf Besonderheiten der Implementierung von Reinforcement Learning Algorithmen mit der Programmiersprache Java eingegangen.

1.1 Maschinelles Lernen als automatische Verarbeitung von Feedback aus der Umwelt

Die beeindruckenden Fähigkeiten künstlicher neuronaler Netze haben viel Medieninteresse in der letzten Zeit erregt. Sogenannte „tiefe“ künstliche neuronale Netze können bspw. erlernen Bilder zu klassifizieren. Die Aufgabe Katzen- von Hundebildern automatisch zu unterscheiden klingt zwar trivial, stellte allerdings über Jahrzehnte eine schier unlösbare Aufgabe dar. Mit der technischen Lösung dieser Art von Mustererkennungsproblemen wurden viele neuartige Anwendungsmöglichkeiten für Computersysteme geschaffen, beispielsweise in der medizinischen Diagnostik, der Industrieproduktion, der wissenschaftlichen Auswertung von Daten, im Marketing, im Finanzwesen, im Bereich der Militär- bzw. Securitytechnik u.v.m. Diese Neuerungen sind gewaltig und im Star-Trek-Jargon gesprochen ist es doch höchst erstaunlich und faszinierend, dass wir in einem Zeitalter leben, in dem im großen Maßstab Dinge getan und Werke geschaffen werden, die nie zuvor in der nun doch schon einige zehntausend Jahre zählenden Geschichte der Menschheit getan worden sind.

Mustererkennung ist allerdings nur ein Anwendungsgebiet des Maschinellen Lernens. Technisch gesehen handelt es sich um das Gebiet des sogenannten „überwachten Lernens“, speziell um den Teil davon, der mit verteilten inneren Repräsentationen arbeitet. Obwohl später nochmal auf das Trainieren von künstlichen neuronalen Netzen eingegangen wird, so soll es in diesem Buch jedoch im Wesentlichen nicht um Mustererkennung gehen.

Der Turing-Award-Preisträger von 2011, Judea Pearl, wird in der Novemberausgabe 2018 von „Spektrum der Wissenschaft“ mit dem Satz zitiert: „Jede beeindruckende Errungenschaft des ‚deep learning‘ läuft darauf hinaus, eine Kurve an Daten anzupassen. Aus mathematischer Sicht ist es egal, wie geschickt man das tut – es bleibt eine Kurvenanpassung, wenn auch komplex und keinesfalls trivial.“

Funktionsanpassungen an eine Datenmenge stellen in diesem Sinne nur einen Teilaspekt von Systemverhalten dar, welches wir gemeinhin „intelligent“ nennen würden. Mithilfe einer an eine gegebene Inputdatenmenge gut angepassten Kurve können wir

zwar Funktionswerte wie z. B. „Katze“ oder „Hund“ für vorher nie gesehene hochdimensional vorliegende Funktionsargumente interpolieren, die Fähigkeiten „intelligenter“ Systeme gehen darüber allerdings deutlich hinaus. Insbesondere wenn wir von lernfähiger, künstlicher Intelligenz sprechen wollen, so möchten wir z. B. darunter auch Aktivitäten fassen, wie die sinnvolle Steuerung eines Staubsaugers, das Öffnen einer Tür durch einen Roboterarm oder kompetente Handlungsempfehlungen, z. B. an der Wertpapierbörse oder auch spannend agierende Gegner bei Brettspielen wie Schach und Go bzw. im Gamingbereich allgemein.

Hierbei muss KI-Software nicht nur vielfältige, teilweise voneinander abhängige Zustände bewerten, sondern muss auch weitblickend agieren. Das Verhalten eines Muster-Klassifikators beschränkt sich eigentlich nur auf die Einordnung von Eingabevektoren in bestimmte Kategorien.

Das Training eines solchen Klassifikationssystems erfolgt durch eine unmittelbare Rückmeldung durch einen wissenden Lehrer, „ja – richtig.“ oder „nein – falsch. Bitte verbessern.“. Für Anwendungsszenarien wie die oben genannten reicht das nicht aus. Hier erhalten wir die meiste Zeit überhaupt keine Rückmeldung darüber, ob wir uns auf einem zielführenden Pfad befinden oder ob wir uns in einer Situation besser anders verhalten hätten. Wir finden am Ende einer Episode mitunter nicht einmal eine wissende Rückmeldung darüber, was die richtige Aktion gewesen wäre, sondern wir kassieren nur mehr oder weniger große „Belohnungen“ oder „Strafen“, ja schlimmer noch, das Ende einer „Episode“ ist teilweise nicht einmal klar feststellbar, wie z. B. beim Erlernen des Laufens oder beim Verfassen eines Buchs über Reinforcement Learning. Wie lässt sich „intelligentes“ Systemverhalten in diesem allgemeineren Sinne automatisch erzeugen bzw. optimieren?

1.2 Verfahren des maschinellen Lernens

Zunächst müssen wir uns einen Begriff von „Intelligenz“ dahingehend bereitlegen, dass wir unter „Intelligenz“ im Wesentlichen „intelligentes Verhalten“ verstehen wollen und entsprechend „Lernen“ als eine Optimierung dieses Verhaltens. Allgemein kann man sagen, dass die Algorithmen des Maschinellen Lernens artifizielles Systemverhalten mithilfe von Rückmeldungen aus der Umwelt iterativ optimieren. Die künstlichen Lernverfahren versuchen dabei die Ausgaben zu verbessern, die bestimmten Systemeingaben zugeordnet werden. Hierfür werden durch die Lernverfahren auf unterschiedliche Weise innere Repräsentationen gebildet, die das Systemverhalten mit Blick auf die zu erfüllenden Aufgaben zunehmend gut steuern sollen.

Für die Einschätzung der Möglichkeiten und Grenzen der diversen Lernverfahren ist es sinnvoll, diese zunächst entlang der Art des Feedbacks einzuteilen, welches sie aus der Umwelt erhalten. Hierbei lassen sich allgemein drei Arten des maschinellen Lernens unterscheiden. Sie unterscheiden sich im Wesentlichen darin, auf welche Weise die „Kritik“ präsentiert wird, durch die sich das Verhalten des künstlichen Systems verbessern

soll: In der ersten Variante korrigiert ein „wissender“ Lehrer die Systemausgaben durch Präsentation der korrekten Ausgabe, in der zweiten erfolgt eine Bewertung von Ausgaben nur in Form von „Belohnung“ und „Strafe“ und in der dritten findet das System in den Eingabedaten autonom Einteilungen und Strukturen, die die Eingabedaten möglichst vorteilhaft abbildet.

Überwachtes Lernen („Supervised Learning“)
Der Datenstrom (x_1, y_1), (x_2, y_2), …, (x_n, y_n) besteht aus Paaren von Eingaben x mit dazugehörigen Sollwerten y für die Ausgabe. Im Lernvorgang, dem „Training“, produziert das System eine vorläufige fehlerbehaftete Ausgabe. Durch eine Anpassung der inneren Repräsentanz mithilfe des Sollwertes wird der künftige Output in Richtung der Vorgabe verschoben und der Fehler für weitere Ausgaben reduziert. I. d. R. wird das System mit einer Teilmenge der verfügbaren Daten trainiert und mit dem verbliebenen Rest geprüft.

Wichtige Verfahren sind die in letzter Zeit durch die Entwicklungen in der Hardware besonders hervorgetretenen künstlichen neuronale Netze mit Delta-Regel, bzw. Backpropagation bis hin zu „tiefen“ Netzen mit zahlreichen Layern, Faltungsschichten und anderen Optimierungen, aber auch k-Nächste Nachbarn Klassifikation (k-NN), Entscheidungsbäume, Support- Vector-Maschinen oder Bayes-Klassifizierung.

Unüberwachtes Lernen (engl. „Unsupervised Learning“)
Beim unüberwachten Lernen steht nur der Eingabedatenstrom $(x_1, x_2, \ldots, x_n)$ zur Verfügung. Aufgabe ist es hier, passende Repräsentationen zu finden, die z. B. die Erkennung von Charakteristika in Datenmengen, Wiedererkennung von Ausnahmen oder die Erstellung von Prognosen ermöglichen. Eine vorherige Einteilung der Daten in unterschiedliche Klassen ist nicht notwendig. Allerdings spielt die Vorauswahl der relevanten Merkmale, sowie die entsprechende „Generalisierungsfunktion“ eine große Rolle. „Unüberwachte Methoden“ können z. B. dafür verwendet werden, automatisch Klasseneinteilungen zu produzieren, – Stichwort Auswertung von Big Data. Es geht hierbei also nicht um die Zuordnung von Mustern in vorhandene Kategorien, sondern um das Auffinden von Clustern in einer Datenmenge. Wichtige Verfahren in diesem Zusammenhang sind: Clustering, k-Means Analyse, Wettbewerbslernen sowie statistische Methoden wie die Dimensionalitätsreduktion, z. B. durch Hauptachsentransformation (PCA) oder Einbettungen, wie das „Word-Embedding“.

Verstärkendes Lernen („Reinforcement Learning“)
Diese Lernmethode bezieht sich auf situierte Agenten, die auf bestimmte Aktionen a_1, a_2, …, a_n hin eine Belohnung („Reward“) r_1, r_2, …, r_n erhalten. Damit haben wir hier Eingaben, Ausgaben und eine externe Bewertung der Ausgabe. Zukünftige Aktionen sollen dahingehend verbessert werden, dass die Belohnung maximiert wird. Das Ziel ist die automatische Entwicklung einer möglichst optimalen Steuerung („Policy“). Beispiele hierfür sind die Methoden des Lernens mit temporaler Differenz wie z. B. Q-Learning

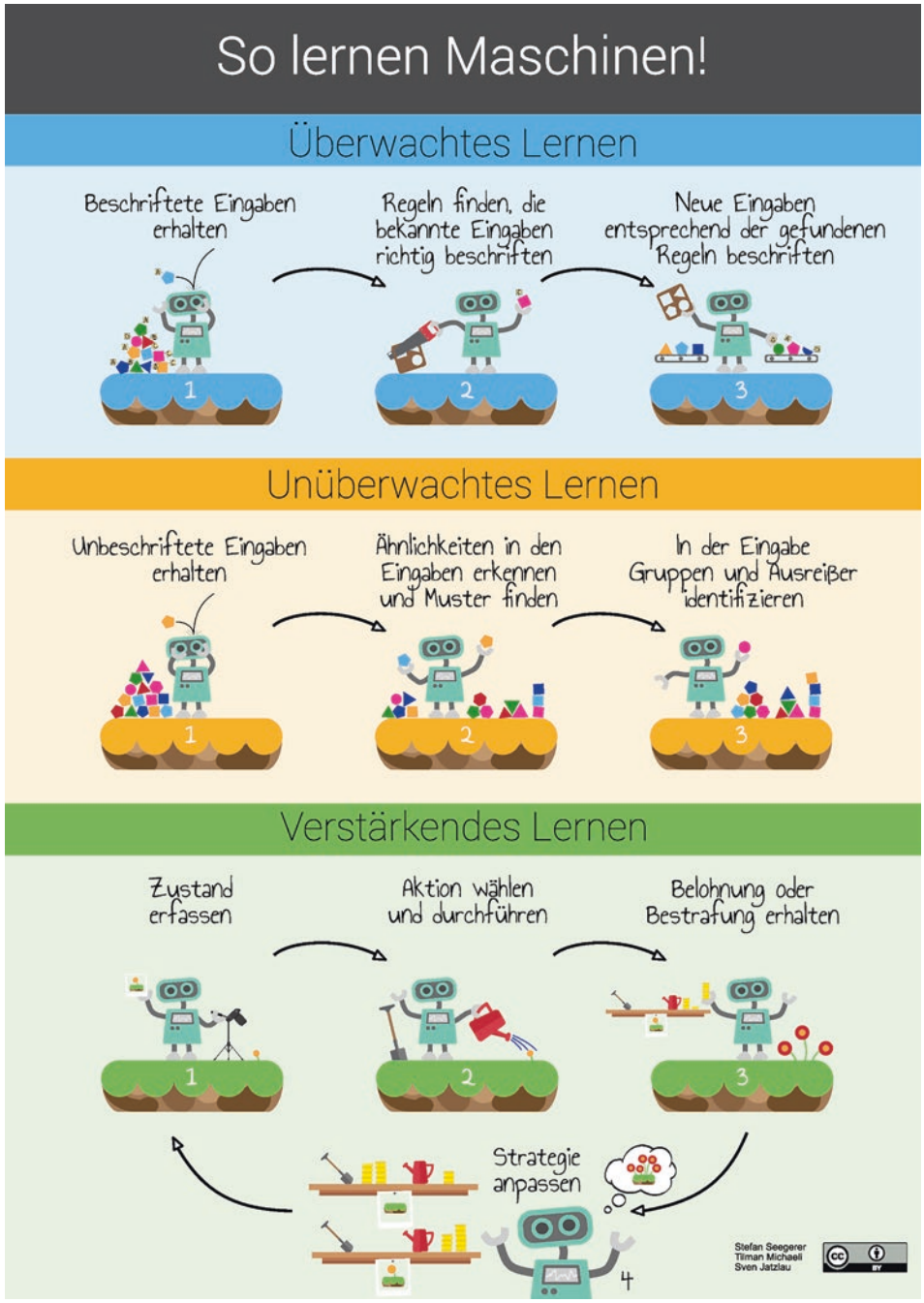

Abb. 1.1 Die verschiedenen Paradigmen des Maschinellen Lernens Bild: Stefan Seegerer, Tilman Michaeli, Sven Jatzlau (Lizenz: CC-BY)

oder der SARSA-Algorithmus, aber auch Policy-Gradienten Methoden wie „actor-critic"-Verfahren oder modellbasierte Methoden (Abb. 1.1).

Beim „Überwachten Lernen" wird die innere Repräsentation entsprechend einem vorgegebenen Sollwert angepasst, wogegen beim „Unüberwachten Lernen" dieser „Lehrer" nicht zur Verfügung steht. Hier müssen die Repräsentationen allein aus den vorhandenen Eingaben herausgebildet werden. Dabei geht es darum, Regelmäßigkeiten in den Eingabedaten herauszufinden, z. B. für eine nachträgliche Weiterverarbeitung. Während also beim „Unüberwachten" Lernen Zusammenhänge durch den Algorithmus herausgefunden werden müssen, – als Hilfsmittel dienen ihm hierbei innere Bewertungsfunktionen oder Ähnlichkeitsmaße –, wird beim Überwachten Lernen die richtige Antwort, bspw. die „richtige" Kategorie, extern aus der Umwelt, d. h. von einem Lehrer, zur Verfügung gestellt. Bei „Überwachten" Lernmethoden sind also während der Trainingsphase richtige Lösungen bekannt, z. B. die Unterscheidung von Bildern mit Melanom oder harmloser Pigmentierung. „Unüberwachte" Lernmethoden können dagegen, bspw. zur Optimierung der räumlichen Gestaltung eines Supermarktes oder dessen Preispolitik dienen, indem z. B. erkannt wird, welche Waren oft zusammen gekauft werden.

Man könnte meinen, dass das System beim „Unüberwachten Lernen" im Gegensatz zum „Überwachten Lernen" frei von äußerer Einflussnahme innere Repräsentationen bilden kann. Tatsächlich muss aber zuvor auf die eine oder andere Weise entschieden werden, an Hand welcher Merkmale und auf welche funktionale Weise das „Clustering" stattfinden soll, denn ohne eine solche Festlegung von Relevanz hätte das System keinerlei Möglichkeiten, sinnvolle Einteilungen zu bilden.

Erfolgt die Erzeugung und Anpassung von inneren Repräsentationen beim „Unüberwachten Lernen" gemäß einer internen Fehlerfunktion, welche zwar vorher von außen verankert wurde, aber dann ohne Unterstützung durch einen externen Lehrer auskommt, dann spricht man auch von „kontrolliertem Lernen". Stammt mindestens ein Teil des Fehlersignals, also des Feedbacks, aus der Umwelt des Systems, so liegt mithin eine Bewertung der Ausgabe durch eine Art externe Belohnung vor. Womit wir das Setting des „Reinforcement Learning" erhalten. Dabei kann die Bewertung von Aktionen auch durch Simulationen in einem Modell erfolgen, bspw. in einem Brettspiel, durch Resultate bei simulierten Spielen. Dabei kann der Algorithmus z. B. auch frühere Versionen von sich selbst als Gegenspieler verwenden.

Um Rückmeldungen, insbesondere aus großen oder komplexen Zustandsräumen, zu verarbeiten, kann ein Reinforcement Learning Algorithmus auch auf überwachte Lernverfahren zurückgreifen. Ein besonders prominentes Beispiel für eine Kombination von Reinforcement-Learning mit Deep-Learning ist hierbei das schon erwähnte AlphaGo-Zero-System, das zunächst nichts als die Spielregeln von Go kannte. Mittlerweile spielt die Maschine deutlich besser, als die besten menschlichen Go-Spieler weltweit. Ein ähnliches System für das Schachspiel heißt z. B. „Leela Chess Zero". Das Spiel Go ist deutlich komplexer als Schach und man glaubte lange, dass es auf Grund der astronomischen Anzahl von Zugmöglichkeiten unmöglich sei, einem Computer das Go-Spiel auf menschlichem Niveau beizubringen, – ein Grund dafür, dass die aktuellen Ergebnisse so