Aiur ist gefallen!

Wie die DeepMind-KI AlphaStar Profispieler in StarCraft 2 besiegt hat

DeepMind markiert mit einem Sieg über zwei Profispieler in StarCraft 2 einen Meilenstein der KI-Entwicklung. Wir erklären, wie die KI AlphaStar trotz unvollständiger Information schier unbegrenzte Aktionsräume meistert.

Von Pina Merkert

StarCraft 2 gilt nicht nur als eines der anspruchsvollsten Echtzeitstrategiespiele im E-Sport, es stellt auch eine außergewöhnlich große Herausforderung für künstliche Intelligenzen dar. Denn StarCraft erfordert gleichzeitig eine sehr taktische Kontrolle über Dutzende Spielfiguren (in der StarCraft-Szene als Micro-Management bzw. „Micro“ bekannt) und eine Strategie für den Bau von Gebäuden, das Erforschen von Upgrades und den Bau neuer Spielfiguren („Macro“). Dabei müssen Spieler mit unvollständigen Informationen auskommen, da der „Nebel des Krieges“ meistens die Aktivitäten des Gegners verschleiert. Spielfiguren (Einheiten) lüften den Nebel nur in einem kleinen Umkreis, sodass Spieler mit ihnen zum Gegner ziehen müssen, um dessen Spielfiguren und Gebäude zu sehen und daraus Rückschlüsse auf seine Strategie und Taktik zu ziehen („Scouting“). Professionelle StarCraft-Spieler erkennen bereits an Kleinigkeiten, welche Strategie ein Gegner einschlägt, und passen die eigene Strategie darauf an. Das geht, da StarCraft wie Schere-Stein-Papier funktioniert: Für jeden Angriff gibt es eine passende Verteidigung. Und aus einem gekonnt verteidigten Angriff ergibt sich ein Vorteil für einen Gegenangriff, gegen den der Gegner wiederum mit einer eigenen Strategie antwortet. Dank dieser Dynamik behauptet sich StarCraft seit vielen Jahren als eines der populärsten und komplexesten E-Sports-Spiele.

Künstliche Intelligenz wird immer wieder daran gemessen, ob sie in Spielen das menschliche Vorbild übertreffen kann. Als Deep Blue 1996 den damaligen Weltmeister Garri Kasparow im Schach besiegen konnte, wurde das rund um die Welt als entscheidender Meilenstein für die Entwicklung von KI wahrgenommen. In den Jahren danach traten KIs immer wieder in Spielen mit unterschiedlichen Eigenschaften gegen menschliche Meister an. 2016 schlug AlphaGo Fan Hui und Lee Sedol – zwei der besten Go-Spieler der Welt. Go hat im Vergleich zu Schach wesentlich mehr Zugmöglichkeiten pro Spielrunde. 2017 schlugen die KIs Libratus und DeepStack professionelle Poker-Spieler im Texas Hold’em. Beim Poker haben Spieler anders als beim Schach und Go nur unvollständige Informationen über die Spielmöglichkeiten der Gegner.

Schwerer als Go und Poker

Für eine KI ist StarCraft eine enorme Herausforderung. Im Prinzip kann sie zu jedem von 60 Einzelbildern pro Sekunde eine ganze Batterie an Spielzügen in Auftrag geben, deren Auswirkungen aber möglicherweise erst nach einer Stunde Spielzeit relevant werden. Gleiches gilt für den Gegner. Der Entscheidungsbaum ist damit um viele Größenordnungen breiter als bei Go. Daher ist es unmöglich, wie beim Schach systematisch alle möglichen

Spielzüge der nächsten Zeit zu bewerten und den effektivsten auszuwählen. Dazu kommt die unvollständige Information: Nur durch aktive Spielzüge beim Scouting erfährt eine KI, wie ihr Gegner genau spielt. Diese Information braucht sie, um eine effektive Strategie auszuwählen. Beim Scouting kommt es aber meist zum Kampf und der Spieler verliert die forschende Figur oft gegen die gegnerische Armee.

DeepMind kooperiert mit Blizzard

Gerade weil StarCraft eine solche Herausforderung darstellt, kooperiert die Google-Tochter DeepMind schon lange mit Blizzard, dem Hersteller von StarCraft. Gemeinsam entwickeln sie ein API namens „PySC2“, das StarCraft 2 so erweitert, dass Computer nicht mehr die Spielgrafik interpretieren müssen, sondern mit Feature-Karten direkt Informationen über das Spielgeschehen bekommen. Solche Feature-Karten bestehen beispielsweise aus einer Matrix, in der in einer Zelle der Typ einer gegnerischen Spielfigur steht. Zellen mit 0 zeigen an, dass dort keine Spielfigur steht. So aufbereitete Daten kann ein Computer viel leichter verarbeiten als die hübsch berechnete 3D-Grafik, aus der Menschen bei StarCraft alle Informationen ziehen. Das Problem, die Spielgrafik direkt zu interpretieren, bleibt auch weiter ungelöst.

Das StarCraft-API PySC2 gibt es seit 2017 als Open Source bei GitHub (Repository und Blogposts dazu siehe ct.de/yxm8). Seitdem nutzen diverse KIs das API und treten in einer eigenen, von Blizzard organisierten Liga gegeneinander an. Diese frühen StarCraft-Bots spielten bislang auf einem recht niedrigen Niveau.

DeepMind beschloss im Sommer 2018, sein StarCraft-Team personell aufzustocken, um ähnlich wie bei Go professionelle Spieler zu schlagen. DeepMind forscht fast ausschließlich an neuronalen Netzen und entschied sich das Problem mit Reinforcement Learning [1] und tiefen neuronalen Netzen anzugehen. Aus diesen Anstrengungen ging die KI „AlphaStar“ hervor. AlphaStar nutzt keine wirklich neue Idee, sondern kombiniert einen Blumenstrauß an aktuellen Techniken aus der Forschung an neuronalen Netzen. DeepMind listet diese in ihrem Blogpost zu AlphaStar nur auf; im Abschnitt zur Technik geben wir jeweils eine ganz kurze Einordnung, wozu die Tricks und Kniffe dienen. Ein wirklich vollständiges Bild erhält man aber nur, wenn man die zugehörigen Forschungspaper liest. Die Wichtigsten haben wir unter ct.de/yxm8 verlinkt.

Wie AlphaStar spielt

Als AlphaStar den besten StarCraft-Spieler im Team von DeepMind verlässlich besiegen konnte, wandten sich die KI-Forscher an die Spieldesigner bei Blizzard, die ihnen den deutschen Profispieler „TLO“ aus der E-Sports-Mannschaft „Team Liquid“ als Gegner vorschlugen. Da TLO in StarCraft eigentlich Zerg (eine von drei „Rassen“ in StarCraft) spielt, AlphaStar aber bislang nur Protoss gegen Protoss auf einer einzelnen Karte beherrscht, zogen sie später noch TLOs polnischen Teamkollegen „MaNa“ hinzu.

DeepMind wählte für die jeweils fünf Spiele gegen die beiden Profis TLO und MaNa jeweils einen anderen Agenten aus der AlphaStar-Liga aus. Die beiden Menschen konnten daher nicht gezielt nach einer Schwäche in der Strategie eines einzelnen Agenten suchen und wurden von Match zu Match mit sehr unterschiedlichen Taktiken konfrontiert. Menschliche Profis spielen mit ähnlich großer Variation in ihren Strategien, damit ihre Gegner sich nicht so leicht vorbereiten können.

AlphaStar sieht über die Feature-Karten des StarCraft-API gewissermaßen das gesamte Spielfeld auf einmal. Die Agenten können überall auf dem Spielfeld Befehle geben, ohne dafür den Blickwinkel der Kamera verschieben zu müssen. Die Anzahl an Aktionen pro Minute hat DeepMind auf ein für menschliche Profis übliches Maß von etwas mehr als 300 begrenzt. AlphaStar brauchte für die Berechnungen zu einem einzelnen Frame etwa 300 Millisekunden, was sogar über der Reaktionszeit von Menschen liegt.

In der sehr empfehlenswerten Aufzeichnung des Livestreams zu den Matches (siehe ct.de/yxm8) erklärt der bekannte StarCraft-Kommentator Artosis, warum AlphaStar auf einem nie da gewesenen Niveau spielt: Die Strategien unterscheiden sich von den bekannten Strategien menschlicher Spieler nur in Details: So baut AlphaStar in allen Varianten mehr Drohnen als Menschen, vermutlich um Verluste bei frühen Angriffen des Gegners vorzubeugen. Außerdem stellten nur wenige der AlphaStars dem Gegner am Eingang der Heimatbasis Gebäude in den Weg. Menschen nutzen diese Strategie sehr oft zur Verteidigung.

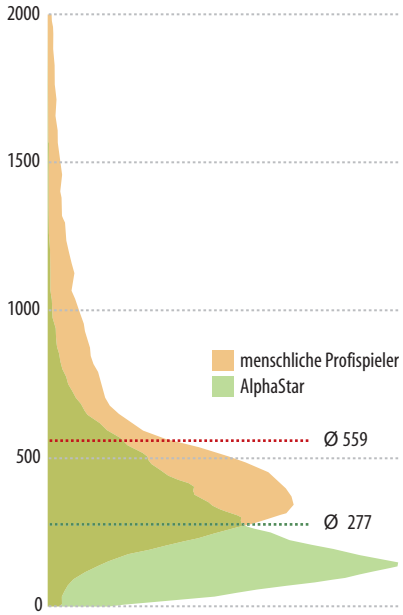
Viel auffälliger als die langfristige Strategie war das Micro-Management von AlphaStar. Die KI steuerte Spielfiguren ausgesprochen raffiniert und vermied dadurch Verluste. Auch setzte der Computer bevorzugt auf Angriffstaktiken, die ein besonders ausgefeiltes Micro-Management (kurz: Micro) erfordern. Menschen können sich bei solchen Taktiken oft nicht auf genügend viele Spielfiguren gleichzeitig konzentrieren. AlphaStar nutzte sein überlegenes Micro bei allen Spielen, um sich einen Vorteil gegen die menschlichen Profis zu verschaffen, den diese strategisch nicht wettmachen konnten: Sowohl TLO, als auch Mana verloren fünf Spiele in Folge gegen die KI-Agenten.

Dass AlphaStar vor allem beim Micro Überlegenheit demonstriert, lässt sich aus der Funktionsweise des Reinforcement Learning erklären: Verändert AlphaStar sein Vorgehen beim Micro, erfährt der Agent nach kurzer Zeit, ob diese Änderung zum Verlust der Spielfigur oder zum Besiegen des Gegners geführt hat. Die kurzfristigen Entscheidungen lassen sich leicht bewerten. Änderungen an der Strategie wirken sich hingegen erst viel später aus, sodass dem Lernalgorithmus meist nur schwache Gradienten zur Verfügung stehen. Das Lernsignal ist dann weniger stark und der Agent braucht zum Lernen erheblich länger. Daher lernt AlphaStar zuerst ein nahezu perfektes Micro, wäh-

Das Warp-Prisma setzte die beiden Immortals mehrfach neben den Drohnen der KI ab und flüchtete, wenn die KI ihre Stalker zurückzog. Die KI hätte das Warp-Prisma mit einem einzigen Phoenix abschießen können. Doch diese konnte die KI nicht bauen.

Aktionen pro Minute

Dieses Histogramm zeigt, wie viele Befehle AlphaStar und Profispieler pro Minute dem Spiel geben. Die Anzahl schwankt stark nach Spielsituation. Menschliche Profis setzen in seltenen Fällen über 2000 Befehle pro Minute ab. Im Durchschnitt sind es aber 559. AlphaStar ist so begrenzt, dass er im Durchschnitt 277 Befehle pro Minute gibt. Dass die KI in einer Minute mehr als 1000 Befehle gibt, kommt praktisch nicht vor.



rend das Macro länger braucht und zum Ende der Trainingszeit auch nicht das gleiche Level an Perfektion erreicht.

Spiel 11

Nach dem klaren Sieg lud DeepMind MaNa zur Präsentation der Ergebnisse ein weiteres Mal nach London ein. Der Profi sollte den kommentierten Livestream (siehe ct.de/yxm8) mit einem letzten live ausgetragenen Spiel abrunden. DeepMind hatte für dieses Spiel extra eine weitere Variante von AlphaStar trainiert. Entgegen seiner Geschwister konnte dieser AlphaStar nicht die ganze Karte auf einmal sehen. Er musste wie ein Mensch die Kamera steuern, um je einen Bildausschnitt zu sehen und konnte auch nur in diesem Bildausschnitt Befehle geben.

Im Training musste sich dieser neue AlphaStar in Spielen gegen seine älteren Geschwister in der AlphaStar-Liga beweisen, die den zusätzlichen Einschränkungen nicht unterworfen waren. Zu Beginn des Trainings hatte er mit dem Bildausschnitt zu kämpfen, doch seine Spielstärke wuchs stetig und erreichte im Verlauf einer Woche fast das gleiche Niveau wie die besten anderen Agenten in der Liga. DeepMind war daher zuversichtlich, dass auch dieser AlphaStar MaNa besiegen könnte.

Zu Beginn des Spiels sah auch alles nach einem weiteren Sieg für AlphaStar aus: Die KI nutzte ihr überlegenes Micro-Management, um nach Minuten bereits einen wirtschaftlichen Vorteil gegenüber MaNa herauszuspielen. MaNa antwortete mit einer riskanten Strategie, bei der er mit einem fliegenden Transporter wenige kampfstarke Einheiten heimlich hinter AlphaStars Drohnen absetzte. Diese Taktik funktioniert normalerweise nur einmal mit begrenztem Schaden, weil der Gegner bereits mit dem Bau eines einzelnen Jagdfliegers eine effektive Abwehr dagegen besitzt. Doch diese Variante von AlphaStar konnte diesen Jagdflieger einfach nicht bauen. Stattdessen baute sie eine unwirksame andere Einheit im gleichen Gebäude. MaNa konnte den Angriff daher mehrmals wiederholen, worauf AlphaStar seine Armee jeweils zurückziehen musste, statt angreifen zu können. Diese Untätigkeit nutzte MaNa aus, baute eine schlagkräftige Armee und zerstörte mit ihr jedes einzelne Gebäude der KI. Ein Mensch hätte an AlphaStars Stelle die Niederlage früher erkannt und kapituliert. Aber DeepMind hatte den Befehl zum Kapitulieren nicht in AlphaStar einprogrammiert.

Die Technik hinter AlphaStar

Deep Learning eignet sich gut, einer KI statistisch fundierte Intuitionen anzutrainieren, die ihr helfen, Entscheidungen zu treffen, die sie zum Sieg führen. Bei AlphaGo [2] hatte DeepMind diese Idee bereits benutzt, um die relevantesten Äste für den zugrunde liegenden Monte-Carlo-Tree-Search-Algorithmus auszuwählen. Da StarCraft aber noch viel mehr Handlungsmöglichkeiten bietet als Go, konnte DeepMind nicht auf Monte-Carlo-Tree-Search aufbauen. Stattdessen trainierte das Team Agenten, bei denen ein neuronales Netz nach Transformer-Artchitektur (Paper siehe ct.de/yxm8) Sequenzen von Spielzügen generiert. Da sie das mit Long-Short-Term-Memory (LSTM) [3] kombinieren, ähnelt die Idee der Funktionsweise von Google Translate (siehe ct.de/yxm8).

Entscheidungshilfe bietet ein Value-Network, ein zweites neuronales Netz, das darauf trainiert ist, aus den Informationen über den Spielstand und der Entscheidung zum aktuellen Spielzug eine Wahrscheinlichkeit vorherzusagen, ob der Agent das Spiel gewinnt. Für die Entscheidungen zu einzelnen Spielzügen geht der „Auto-Regressive-Policy-Head“ davon aus, dass sie unabhängig voneinander zum Spielausgang beitragen. Damit ergeben sich bedingte Einzelwahrscheinlichkeiten für jede geplante Entscheidung. Normalerweise wären diese Wahrscheinlichkeiten nicht nur von der Entscheidung abhängig, sondern auch davon, an welcher Stelle in der Sequenz AlphaStar die Entscheidung eingereiht hat. Da das die zu lernenden Wahrscheinlichkeiten unnötig verkompliziert, kombiniert DeepMind das mit von Google Brain entwickelten Pointer-Networks. Die machen die bedingten Einzelwahrscheinlichkeiten unabhängig von der Position eines Spielzugs in der vom Transformer erzeugten Sequenz.

Die Universität Oxford hatte ihre Counterfactual-Multi-Agent-Policy-Gradients (COMA) bereits 2017 mit StarCraft-2-Agenten evaluiert. Dort steuerte je ein Agent eine einzelne Einheit (Micro). Zum Trainieren der Agenten verwendet COMA aber eine „Centralized Value Baseline“. Das ist eine Funktion, die sich das Gesamtergebnis des Zusammenspiels aller Agenten betrachtet und die Agenten dahingehend lobt oder tadelt, wie sie zum Erfolg des Gesamtsystems beitragen. Da jeder Spielzug in der vom Transformer berechneten Sequenz aus einem Befehl für eine einzelne Spielfigur besteht, kann eine solche Funktion individuelles Feedback zu einzelnen Entscheidungen liefern, während die „Centralized Value Baseline“ das Gesamtbild betrachtet.

Überraschend ist bei AlphaStar, dass DeepMind anders als bei ihren Quake 3 spielenden Reinforcement-Learning-Agenten auf ein modellfreies System gesetzt hat. Statt AlphaStar zu zwingen, ein Modell des Spielgeschehens zu erstellen verlässt sich DeepMind darauf, dass AlphaStar alle nötigen Informationen über die Welt und das Spielgeschehen in den Parametern und Aktivierungen seiner neuronalen Netze darstellt. Viele Forscher gingen zuvor davon aus, dass solch ein impliziter Ansatz an der Komplexität von StarCraft scheitern müsste.

Im Blogpost zum Livestream (siehe ct.de/yxm8) äußert DeepMind die Überzeugung, dass sich die Struktur von AlphaStar neben dem Spielen von StarCraft auch für andere sequenzbasierte Auf-

gaben wie Übersetzung und Video- und Textgenerierung eignet. Der Vorteil gegenüber bestehenden Systemen bestünde darin, dass dieses System besser langfristige Strategien verfolgen kann. Beispielsweise hatten KIs bislang beim Erzeugen eines Texts Probleme, bei einem Thema zu bleiben. Von AlphaStar inspirierte KIs lassen hier auf stringentere Texte hoffen.

Imitation

Im Prinzip kann AlphaStar mit dieser Struktur langfristige Strategien verfolgen. Beispielsweise Drohnen zum Abbauen von Mineralien schicken, Warpknoten bauen, die wiederum Stalker produzieren, die den Gegner angreifen. Das Umsetzen einer solchen Strategie dauert in StarCraft 2 mehrere Minuten, in denen dem Agent Millionen verschiedenster Befehle zur Auswahl stehen. Initialisiert man AlphaStar mit Zufallszahlen, erzeugt er auch zufällige Spielzüge, die aber in (fast) allen Fällen nicht zum Erfolg führen. Beim Reinforcement Learning kann ein Agent seine Parameter mit erfolgreichen Beispielen viel gezielter anpassen als mit Negativbeispielen. Mit Zufallsbefehlen lernt der Agent auch mit Tausenden von Beispielen meist nicht einmal die Grundzüge des Spiels.

Bevor AlphaStar auf eigene Faust spielen darf, nimmt ihn DeepMind daher per Imitation Learning an die Hand. Dafür verwandelt DeepMind StarCraft in ein Problem des überwachten Lernens (Supervised Learning), das wesentlich mehr und vor allem positive und damit gezielte Lernsignale liefert. DeepMind nutzte dafür tausende Replays von Spielen, die Menschen in Blizzards Online-Arena BattleNet ausgefochten haben. Mit diesen Spielen als Vorlage sollte AlphaStar zunächst lernen, die exakt gleichen Spielzüge wie der gewinnende Spieler zu erzeugen. Jede Abweichung bestrafte das System mit dem Ändern der Parameter, sodass AlphaStar alle grundsätzlichen Strategien für StarCraft lernte. Der so trainierte Agent spielte nach Angaben von DeepMind bereits auf dem Niveau erfahrener Hobbyspieler (Gold-Level im BattleNet), aber nicht besser als Profis.

Wettbewerb

Ein per Imitation vortrainierter Agent kann, anders als untrainierte Agenten, immerhin sinnvoll ganze StarCraft-Spiele bestreiten – auch wenn er keine Profis besiegt. DeepMind kopierte diesen Agenten ein paar Mal und variierte ihn jeweils ein wenig. Diese leicht unterschiedlichen AlphaStars ließ DeepMind in der „AlphaStar League“ gegeneinander antreten.

Über die Value-Funktion kann AlphaStar unterschiedliche Ziele verfolgen: So kann ein Agent eine besonders hohe Belohnung erhalten, wenn er einen bestimmten Gegner besiegt. Ein anderer Agent bekommt die hohe Belohnung vielleicht nur, wenn er eine ganze Gruppe an Gegnern verlässlich besiegen kann. Ein dritter bekommt vielleicht eine höhere Belohnung, wenn er bestimmte Spielfiguren baut.

Siegreiche Agenten bekamen nach diesem Schema immer neue Varianten, die sich in der Liga beweisen mussten, während Verlierer nach und nach aus der Liga flogen. DeepMind achtete dabei auf große Diversität. Da StarCraft zu jeder Spielfigur eine effektive andere Spielfigur als Antwort bereithält, gibt es selbst für die besten Strategien erfolgreiche Gegenstrategien. DeepMind passte die Ziele neuer Agenten daher oft so an, dass sie nach Strategien gegen den aktuellen Spitzenreiter suchten.

Dadurch steigerte sich das Spielniveau der AlphaStar-Liga im Laufe des Trainings immer weiter. Da die neuen Agenten nicht mehr auf Replays menschlicher Spieler angewiesen waren, konnten sie neue Strategien entwickeln, die Menschen bei StarCraft noch nie eingesetzt hatten.

Umgerechnet auf Spiele in Echtzeit (DeepMinds KI-Version von StarCraft kann beim Training schneller als in Echtzeit spielen) sammelte jede AlphaStar-Variante etwa 200 Jahre an ununterbrochener Spielerfahrung in StarCraft 2 an. Auf zahlreichen Rechenknoten mit Googles KI-Beschleuniger TPU3 dauerte das Training etwa eine Woche.

Ein Meilenstein

Nie zuvor hat eine KI menschliche Profis in einem so anspruchsvollen Spiel wie StarCraft 2 besiegt. Die KI punktet nicht nur mit überlegener Arithmetik: AlphaStar besitzt auch die nötige Intuition, um auf Taktiken der Menschen strategisch zu reagieren. Damit beweist die KI Flexibilität und die Fähigkeit, langfristige Pläne zu verfolgen. Am Ball zu bleiben war lange eine Schwäche künstlicher Intelligenz.

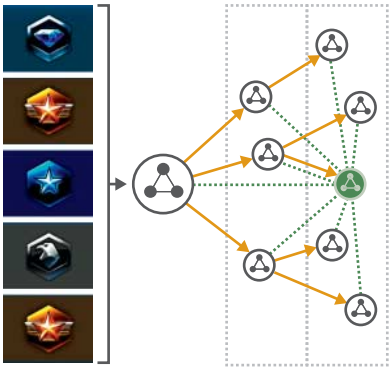
Das lässt abseits vom Spiel auf viele ernsthafte Einsatzmöglichkeiten für die Technik hoffen. Von einem Sprachmodell mit Weitblick würden neben Textgeneratoren auch Sprachassistenten und Hotline-Bots profitieren. Für automatische Übersetzungen lässt die Technik auf Formulierungen hoffen, die besser den Kontext des gesamten Texts miteinbeziehen. Bis dahin ist es aber noch ein weiter Weg, denn bei realen Anwendungen bekommen AlphaStar und seine Nachfolger kein so klares Feedback zu Sieg und Niederlage wie in StarCraft.

Beim E-Sport werden StarCraft-Profis einen genauen Blick auf AlphaStars Spielstil werfen. Möglicherweise wird man im BattleNet in Zukunft häufiger 18 statt 16 Drohnen in Protoss-Basen sehen. Und auch die Taktik des Verbauens der Rampe am Eingang der Basis werden sicherlich einige Spieler auf die Probe stellen. Spannend wird, wie AlphaStar andere Rassen auf anderen Karten spielt.

(pmk@ct.de) c't

Die AlphaStar-Liga

Nachdem die KI AlphaStar am Beispiel tausender Spiele grundsätzliche Taktiken gelernt hat, muss sie in einer Liga gegen jeweils leicht veränderte Kopien von sich antreten. Die künstliche Evolution der Liga überleben nur die stärksten KIs, die nach einer Woche Training besser als Menschen spielen.



Literatur

[1] Sebastian Stabinger, Zuckerbrot und Peitsche, Einer selbst gebauten KI per verstärkendem Lernen beibringen Pong zu spielen, c't 21/2018, S. 166
[2] Harald Bögeholz, Jubel und Ernüchterung, Google AlphaGo schlägt Top-Profi 4:1 im Go, c't 7/2016, S. 44
[3] Sebastian Stabinger, Langes Kurzzeitgedächtnis, Mit rekurrenten neuronalen Netzen Texte verschlagworten, c't 19/2017, S..170

Blogpost, Video bei YouTube:
ct.de/yxm8