

KI-Techniken

In diesem Kapitel werden die zwei wesentlichen bis heute entwickelten und vielfach verwendeten KI-Techniken erklärt – Expertensysteme und Neuronale Netze.

Fokus liegt dabei auf dem Verständnis des Grundprinzips, um eine Einordnung im Bezug auf die Versprechen der KI zu ermöglichen. Auf mathematische Herleitung, die Darstellung der Algorithmen sowie der verschiedenen Varianten wird dabei verzichtet.

Expertensysteme

Expertensysteme bilden die Idee einer Maschine, die mit ihrer Logik große Wissensbestände verarbeiten kann, direkt ab. Die Funktionsfähigkeit dieser Technik wurde früh bewiesen und die Aussicht der Umsetzung faszinierte die Branche einige Dekaden. Als die vorher knappe Rechenleistung in den 1980-ern endlich verfügbar wurde, zeigte sich aber, dass diese Logik nicht aufging – man hatte sich über den Charakter der eigenen Fragestellungen getäuscht.

Entwicklung

Erste Computersysteme wurden zu Beginn der 1960er über den militärischen Bereich hinaus für Unternehmen nutzbar, anfangs vor allem für Buchhaltung. Zur gleichen Zeit versuchte man mit Computern bereits, Menschen-ähnliche Intelligenz umzusetzen. Mitte der 1960er wurde an der Stanford Universität ein *Expertensystem* als erfolgreiches Beispiel für *artificial intelligence* präsentiert.

Die Frage „Was ist Intelligenz?“ wird bei diesem Ansatz recht einfach beantwortet: Wenn eine Frage richtig beantwortet wird. Im einfachen Falle speichert man Frage-Antwort-Paare im Computer, sodass dieser via Textvergleich zu einer Frage eine passende Antwort findet. Heute findet man diesen Ansatz unter Titeln wie 'Häufig gestellte Fragen' oder *FAQ*, etwa bei Webseiten mit Produktinformationen.

Die Bedeutung einer Frage wird dabei allerdings nicht erkannt. Eine einfache Verneinung wie „Gibt es Fertig-Pizza ohne Salz?“ bringt das Verfahren schon an seine Grenze.

Expertensysteme (ES) gehen weiter und verarbeiten auch den Bedeutungsgehalt von Sprache, indem sie einen Satz in Aussagen zerlegen, und dabei Gegenstände (Objekte) und Eigenschaften (Adjektive) unterscheiden. Sie speichern die Aussagen in Regeln, die unabhängig von grammatikalischen Eigenschaften wie Zahl (Numerus) oder Zeit (Tempus) zuordenbar sind.

Zu Beginn der Entwicklung in den 1960ern sah man, dass der Umfang von Regeln für ein Universalgelehrten-System zu groß wäre, um es mit damals absehbarer Computerleistung zu realisieren. Man begnügte sich nun damit, das Wissen spezieller Fachgebiete umzusetzen – daher *Expert system*.

Wiederum braucht es erst menschliche Experten, die das Wissen des Fachgebiets kennen und in der Logik des ES formulieren können. Sie erschaffen eine Wissensbasis aus eindeutigen Regeln, die dann bei einer Abfrage vom System ausgewertet werden.

Logiksysteme

Bei der Umsetzung dieses Konzepts in Software stellte sich die Aufgabe, die in der Geschichte der Mathematik bereits entwickelten Logik-Formalisten einerseits in Programmiersprache abzubilden, andererseits den Anwendern der Maschine verständlich darzustellen.

Betrachten wir zuerst das Verhältnis von Logik und Programmierung, bevor wir den *Dialog* untersuchen.

Um Fragen richtig beantworten zu können, kann man zuerst die Zahl der zulässigen Fragen begrenzen. Richtig und falsch sind klar definiert, wenn man sich in einem *logischen System* bewegt, das eindeutige, unmissverständliche Aussagen erzwingt. Im einfachsten Fall der *Binärlogik* sind Aussagen auch *nur* richtig *oder* falsch. Ist Sieben größer als Fünf? Ist Blau eine Zahl? Hat der Bäcker am Sonntag geöffnet?

Mehr Differenzierungen werden in Logiksystemen zugänglich, die *mehrwertige* Aussagen erlauben wie „Katzen und Hunde sind Tiere“, auch mit *Quantoren* wie „Katzen haben vier Beine“. *Prädikatenlogik* ergänzt diese Systeme um weitere aussagenlogische Unterscheidungen wie:

„Es gibt Tiere mit weniger und mit mehr als vier Beinen.“

Weitere Fälle lassen sich via Folgerungen verarbeiten:

Wenn die Ampel rot zeigt, bleiben die Autos stehen. Wenn die Sonne scheint, entstehen Schatten. Wenn die Lampe brennt, wird Strom verbraucht.

Die *Modallogik* erweitert mit der Unterscheidung zwischen *möglich* und *notwendig* noch einmal den Raum der logisch fassbaren Aussagen. Eine weitere Unterscheidung ergibt sich aus dem Umgang mit *Wahrscheinlichkeiten*.

Transferleistung

Die Arbeit von Expertensystemen (ES) besteht nun darin,

- a) die Regeln von den Fachleuten aufzunehmen, die dann
- b) bei der Auswertung vom System kombiniert werden können.

Diese Fähigkeit, Folgerungen zu ziehen, geht über das Verarbeiten von Frage-Antwort-Paaren hinaus und verschaffte dem Verfahren das Attribut 'intelligent'.

Die Summe der Regeln wird in Anlehnung an eine *Database* hier in einer *Knowledge base* abgelegt. Die Kombination der Regeln zur Beantwortung einer konkreten Anfrage übernimmt dann die *Inference engine* (sinngemäß: Ableitungsmaschine).

Voraussetzung für Schlussfolgerungen sind Regeln, die über ihre Subjekte, Prädikate und Objekte sprachlich *anschlussfähig* sind. Denn das System kann Fragen nur anhand von Textvergleich den vorhandenen Regeln zuordnen.

Aus „Tomaten sind rot“ und „Tomaten gehören zu den Gemüsen“ kann das System folgern: „Es gibt rote Gemüse“ In dieser Weise lassen sich eine Vielzahl von Regeln sammeln und kombinieren.

Vorteilhaft erweist sich die Möglichkeit, Aussagen in *Variablen* abzulegen, auf die dann später Bezug genommen werden kann. „Das für den Salat verwendete Gemüse ergibt püriert eine Ockerfarbe.“ In der menschlichen Sprache simpel, für ein Expertensystem eine Herausforderung.

Programmierung

Die Verarbeitung der Expertenregeln durch die *Inference engine* muss dann von Programmierern in einer Computersprache umgesetzt werden. Klassische Programmiersprachen beruhten bis dahin auf einer eindeutigen Trennung zwischen Daten und Programm: Texte wie „Tomate“ werden als Inhalt von Datensätzen oder Variablen gehalten, davon getrennt die Logik im Programm formuliert wird.

Nun aber musste man Texteingaben wie „Broccoli sind grün“ in Logik *umwandeln*, die später vom Computer zur Berechnung oder Bewertung der Aussage „Es gibt rote und grüne Gemüse“ diente.

1958 wurde mit der Programmiersprache *LISP* (List Processing) eine Lösung entwickelt. Daten und Logik werden hier gleichwertig als Liste verarbeitet – „grün“ ist eine Liste von Zeichen, „Brokkoli sind grün“ eine Liste von Worten, und auch der Ablauf eines LISP-Programms wird über Listen gesteuert – die alle den gleichen *Datentyp* nutzen. Programmierung besteht hier im wesentlichen in der Umwandlung von Listen in andere, die selbst wieder ausführbar sind. Nach LISP folgten weitere Sprachen für Expertensysteme wie PROLOG.

Dialog

Die Auswertung des KI-Wissens soll wiederum auch für Menschen ohne Computerkenntnisse möglich sein. Die Programmierung eines ES umfasst daher die Dialogführung mit einem Anwender.

Die in der menschlichen Sprache üblichen Redundanzen – sachlich unnötige Wiederholungen – oder Unschärfen müssen dabei durch Gegenfragen geklärt werden, etwa: „Meinen Sie mit 'Salatblätter' a) eine Anzahl von Blättern der gleichen oder b) verschiedener Sorten?“