Zusammenfassung

Expertensysteme sind ein Teilbereich des Forschungsgebiets "Künstliche Intelligenz", welcher in jüngster Zeit auch im Agrarbereich verstärkte Aufmerksamkeit findet. Im Gegensatz zu konventionellen Programmen sind Expertensysteme vornehmlich auf die Lösung algorithmisch nicht lösbarer Probleme ausgerichtet. Dabei wird insbesondere auf deklarative Programmiersprachen ("Logik-Programmierung") zurückgegriffen, und neben Faktenwissen verstärkt auch heuristisches und assoziatives Wissen verarbeitet. Ziel von Expertensystemen ist es, die spezifischen Fähigkeiten hochspezialisierter Experten in ihrer Wissensdomäne abzubilden, und somit deren Wissen einem größeren Nutzerkreis verfügbar zu machen. Der Aufsatz gibt eine Einführung in das Gebiet der Expertensysteme, beschreibt mögliche Anwendungsbereiche und zeigt einige Hauptprobleme bei der Erstellung derartiger Systeme auf.

Abstract

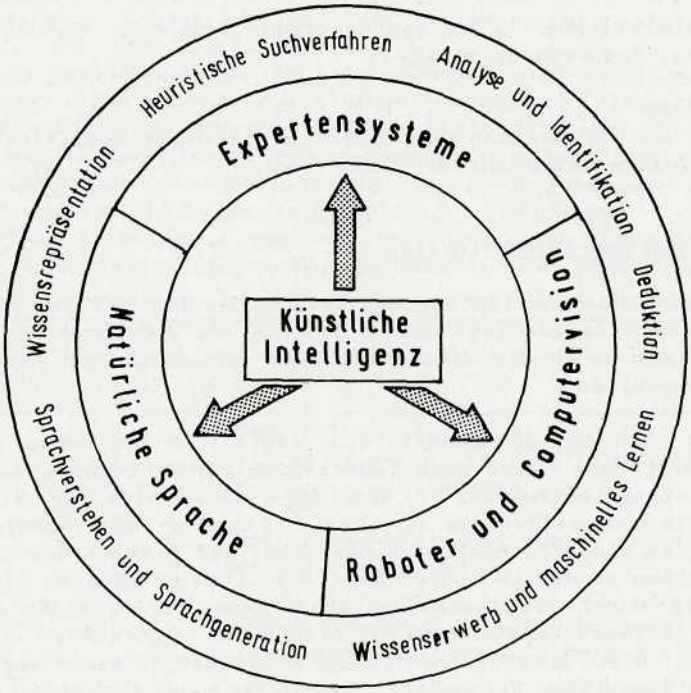
Expert Systems as one branch of Artificial Intelligence Research is attracting increasing attention in agriculture. In contrast to conventional programs, expert systems are mainly concerned with the solution of non-algorithmical problems. The programming of expert systems is mostly based on declarative languages (logic programming). Next to data and facts, extensive use is made of heuristics and associative knowledge. The aim of expert systems is to represent the knowledge of an expert in his specific knowledge domain, thus making his capabilities available for a greater user community. The paper gives an introduction in the field of expert systems. It describes some applications and application areas and discusses some of the major problems in building expert systems.

30.1 Einleitung

Zu den mehr unangenehmen Erfahrungen eines EDV-Neulings mit Computern gehört in der Regel, daß Computer "dumm" sind, d.h. daß sie nichts können, was nicht explizit vorgesehen ist. Daß sie trotzdem manchem Benutzer heute als intelligent erscheinen, ist ausschließlich der Qualität der Hardware- und Softwareprodukte zuzuschreiben, die eine entsprechend benutzerfreundliche Umgebung bieten. Die Idee einer Maschine, die tatsächlich zu intelligenten Verstandesleistungen fähig ist, erweckt hingegen in der Regel mehr Horrorvisionen von Androiden, wie sie in unzähligen Beispielen aus der Science-Fiction-Welt drastisch dargestellt sind. Trotzdem widmet sich jedoch heute eine exponentiell wachsende Zahl von Forschern einem Gebiet, welches sich "Artificial Intelligence - AI" oder in Übersetzung "Künstliche Intelligenz - KI" nennt. Nachdem das KI-Fieber inzwischen auch einige Institutionen im bundesdeutschen Agrarbereich befallen hat, soll es das Ziel dieses Aufsatzes sein, einige grundlegende Konzepte und Methoden eines Teilbereiches dieser Forschung vorzustellen, der sich "Expert Systems" oder "Expertensystem" nennt.

30.2 Expertensysteme als Teilbereich der KI-Forschung

Der Forschungszweig "Künstliche Intelligenz" wird im allgemeinen als Teilgebiet der Informatik beschrieben. Dabei gestaltet sich eine exakte Abgrenzung des Gebiets relativ schwierig, wie aus dem Fehlen einer allgemein anerkannten Definition deutlich wird. Die Spannweite reicht jedoch über den Bereich der Informatik weit hinaus. SAVORY beschreibt Artificial Intelligence allgemein als eine Sammlung computergestützter Technologien, welche einige der natürlichen Fähigkeiten des Menschen emulieren. Basis des Forschungsgebietes ist folglich zunächst einmal die Auseinandersetzung mit den Grundlagen der menschlichen Intelligenz. Hieran ist der Informatiker nur sekundär beteiligt, während primär Psychologen, Mediziner, Linguisten und verwandte Disziplinen gefordert sind. Nur oberflächliches Reflektieren läßt deutlich werden, daß ein Großteil menschlicher Verhaltensweisen Intelligenz erfordert und somit potentielles Anwendungsgebiet für KI-Forschung darstellt. Folgende Bereiche bilden derzeit den Schwerpunkt der KI-Forschung (vergl. Übersicht 1 auf Seite 329)



Übersicht 1. Schwerpunkte der KI-Forschung

1. **Computervision**
   Automatische Lokalisation und zweifelsfreie Identifikation beliebiger zwei- oder dreidimensionaler Strukturen

2. **Roboter**
   Automatisierung einer begrenzten Zahl von Arbeitsgängen durch computergestützte Erkennungs- und Handhabungssysteme, evt. mit beschränkter Lernfähigkeit

3. **Natürliche Sprache**
   Verstehen und Generieren natürlicher Sprache, automatische Übersetzung

4. **Problemlösung**
   Gezielte Lösung von Problemen in einem vordefinierten Lösungsraum mit Hilfe von wissensbasierten Techniken und Heuristiken

5. **Deduktion**
   Beweisen mathematischer Sätze und sonstiger logischer Verknüpfungen; Programmsynthese, Programmverifikation

6. **Expertensysteme**
   Nachbildung der Fähigkeiten von Experten mit Hilfe spezifischer Verfahren wie Problemlösungstechniken oder Deduktion

30.2.1 Begriff und Begriffsdefinition

Wie aus der vorstehenden Auflistung deutlich wird, ist mit dem Begriff Expertensystem nicht ein spezielles, exakt definiertes Verfahren verbunden. Das Gebiet wird vielmehr durch die Zielvorstellung umrissen, bestimmte Fähigkeiten von Experten nachzubilden.

Wodurch zeichnet sich nun ein Experte aus? Besitzt er das Wissen einer 24-bändigen Enzyklopädie? Wohl kaum, denn Experten werden allgemein aufgrund ihrer Fähigkeiten auf einem bestimmten, oft eng begrenzten Gebiet geschätzt. Doch auch derjenige, der die gesammelten, in der Spezialliteratur beschriebenen Fakten und Lösungsverfahren in einem Fachgebiet beherrscht, ist nicht notwendigerweise ein Experte. Zusätzlich zu dem in Algorithmen und Daten erfaßbaren Wissen zeichnet sich der Experte durch Problemlösungskapazitäten aus, die ihm die Fähigkeit geben, Probleme zu lösen, die sich nicht direkt auf Algorithmen oder Daten zurückführen lassen. Diese zusätzlichen Fähigkeiten beruhen meist auf ausgedehnter Erfahrung in dem jeweiligen Fachgebiet. Kennzeichen der Expertise sind:

- hohes Maß an Faktenwissen
- Beherrschung einer großen Zahl an Anwendungsregeln
- Persönliche Erfahrung, die generell nicht über alleiniges Literaturstudium verfügbar ist
- meist die Einschränkung auf ein stark begrenztes Fachgebiet.

Diese Kombination von Wissen und Erfahrung verhilft dem Experten zu der Fähigkeit, praktische (Alltags-)Probleme, die nicht Lehrbuchcharakter haben, in kurzer Zeit zu lösen bzw. die Verfahrenswege zu ihrer Lösung aufzuzeigen. Suboptimale Problemlösung, wie sie im Alltagsleben die Regel ist, läßt sich somit vereinfachend darauf zurückführen, daß wirkliche Experten selten sind und nicht überall verfügbar sein können. Die Experten selbst hingegen leiden unter dem Problem, daß sie sich einen Großteil ihrer Zeit nicht den eigentlichen Aufgaben widmen können, da sie laufend mit Problemen belastet werden, die für sie oft nur Routinearbeiten darstellen. Naheliegend ist daher die Zielvorstellung, ein computerbasiertes System zu erstellen, welches portabel ist und zu beliebiger Zeit an beliebigem Ort die besonderen Fähigkeiten eines Experten verfügbar macht.

30.2.2 Zielbereiche von Expertensystemen

Das potentielle Anwendungsgebiet von Expertensystemen kann als nahezu unbegrenzt angesehen werden, da es auf alle Disziplinen ausgedehnt werden kann, in denen

Probleme anhand erfaßbarer Lösungsmechanismen angegangen werden. Eine Einschränkung erfährt diese Aussage dahingehend, daß alle Aufgaben, die sich mit Hilfe von "Konventionellen" Algorithmen lösen lassen, d.h. durch Anwendung der gängigen Programmierungsverfahren und -sprachen, generell nicht als Probleme für die KI-Forschung angesehen werden. Somit sind auch KI-Forschung Expertensysteme nicht als zukünftiger Ersatz heutiger Programmierverfahren einzustufen.

Im Gegensatz zu den traditionellen Datenverarbeitungsverfahren, die sich überwiegend auf numerische Lösungswege stützen, umfassen KI-Systeme in der Regel nicht-numerische Verfahren der symbolischen Repräsentation, symbolische Schlußfolgerungssysteme (Inferenzsysteme) und Heuristiken. Gemäß der obigen Abgrenzung verfügen sie über problemfeldspezifische Lösungsverfahren und sind auch in der Lage, die gezogenen Schlußfolgerungen explizit zu begründen. Letztlich erfahren auch die Probleme selbst eine gewisse Eingrenzung, da sie alle einer der in Übersicht 2 aufgelisteten Kategorien zugeordnet werden können.

Anwendungsgebiet	Beispiel
Diagnose	Medizinische Diagnose; Störungsdiagnose in elektrischen und mechanischen Systemen
Vorhersage	Vorhersage wirtschaftlicher Entwicklungen; Wettervorhersage
Interpretation	Sprachverstehen und Sprachübersetzung; Interpretation von Bildern und Strukturen
Test und Fehlersuche	"Debugging" von Programmen; automatisches Editieren; Fehlersuche in EDV-Anlagen
Reparatur	Fehlerkorrektur in Schaltkreisen; Navigationskorrektur
Design	Design von Maschinen, Geräten und Gebäuden; CAD-Anwendungen
Systemplanung	Zusammenstellung von Komponenten zu anwendungsfähigen Systemen
Kontrolle	Überwachung von Produktionsprozessen und Verfahrensabläufen
Instruktion	Interaktive Wissensvermittlung, -einübung und -kontrolle

Übersicht 2. Kategorien für die Zielbereiche von Expertensystemen

30.2.3 Historische Entwicklung

Die Entwicklung der ersten Expertensysteme geht zurück auf die grundlegenden Erfahrungen und Mißerfolge, die in den fünfziger Jahren in der neuen Forschungsrichtung Artificial Intelligence gemacht wurden. Die Zielvorstellung maschineller Intelligenz erwuchs zu gleicher Zeit wie die ersten EDV-Anlagen selbst, die von den Medien auch gern als "Elektronengehirne" bezeichnet wurden. Die Anfänge der Forschung waren durch die heute als "naiv" apostrophierte Vorstellung gekennzeichnet, daß man ein Supersystem zur Lösung von Problemen aus den unterschiedlichsten Bereichen aufbauen könne, da sich menschliche Intelligenz auf bestimmte Grundmuster zurückführen lasse. Es bedürfe daher nur einer geringen Anzahl von präzise isolierten Verstandesregeln und eines extrem leistungsfähigen Computers, um unabhängig von einem bestimmten Wissensgebiet die Problemlösungskapazitäten eines Super-Menschen zu generieren. Das bekannteste Ergebnis dieser Forschungen wurde der "General Problem Solver" von Newell, Shaw und Simon, an dem noch bis 1969 gearbeitet wurde. Viele grundlegende Erkenntnisse resultierten aus diesen Arbeiten, doch schließlich kam man zu der Erkenntnis, daß sich die Komplexität menschlichen Denkens und intelligenter Problemlösung nicht auf eine beschränkte Anzahl genereller Lösungsstrategien und -mechanismen reduzieren läßt.

Logische Konsequenz dieser Erkenntnis war die Konzentration auf Problemlösungsmechanismen für abgegrenzte, oft sehr spezifische Wissensgebiete, was zur Entwicklung der Expertensysteme führte. Einige der bekanntesten Anwendungen sollen kurz skizziert werden:

1. Bereits 1965 wurde mit DENDRAL in Stanford ein Expertensystem vorgestellt, welches in seiner spezifischen Domäne zumindest das Examenswissen der meisten Chemiker übersteigt. Es wurde ein Rahmen des Stanford Heuristic Programming Project von FEIGENBAUM u.a. entwickelt, und ist in der Lage, anhand von Daten über Massenspektographie und atomare Massenresonanz die mögliche Strukturformel einer organischen Substanz aus einer Anzahl von Hypothesen zu identifizieren.

2. Am Massachussets Institute of Technology (MIT) wurde 1971 MACSYMA vorgestellt, ein System, dessen Wissensdomäne die symbolische Algebra ist (Integral- und Differentialrechnung), und welches über Vergleichstechniken und algebraische Vereinfachungen in der Lage ist, benutzerseitig eingegebene Gleichungen zu lösen.

3. Das wohl bekannteste Expertensystem wurde MYCIN und das daraus entwickelte EMYCIN als Expertensystem-Shell (vergl. 4.). MYCIN ist das erste in einer Reihe von Expertensystemen für medizinische Anwendungen. MYCIN fragt über interaktiven Dialog mit dem Arzt patientenspezifische Symptome ab und ist in der Lage, Blut- und Meningitisinfektionen zu diagnostizieren und geeignete Behandlungsmethoden vorzuschlagen.

4. Als Supplement zu MYCIN wurde THEIRESIAS entwickelt, welches dem Benutzer ermöglicht, im Dialog an beliebiger Stelle die Anfrage "Why?" oder "How?" einzugeben, woraufhin das System eine Begründung für seine Schlußfolgerungen oder die gerade gestellte Frage abgibt. Derartige Erklärungskomponenten sind heute Bestandteil aller Expertensysteme.

5. Analog zu der in MYCIN zugrundeliegenden Wissensrepräsentationsform wurde PROSPECTOR für ein völlig anderes Gebiet entwickelt. PROSPECTOR erfragt im Dialog Angaben über geologische Merkmale eines Gebiets und über Mineral- und Gesteinsfunde und erbringt als Ergebnis ein Wahrscheinlichkeitsbild für die Auffindung bestimmter Bodenschätze in der Region.

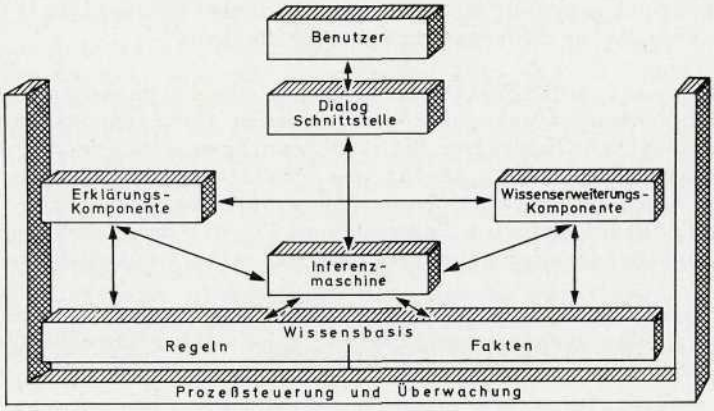
6. Digital Equipment entwickelte schließlich die In-House-Anwendungen R1 und XSEL, zwei Expertensysteme, welche im eigenen Vertriebsnetz der Firma angewandt werden. R1 konfiguriert DEC-Rechenanlagen vom Typ VAX-11/780 anhand von Kundenspezifikationen, stellt die räumlichen Beziehungen zwischen den Komponenten her und erstellt Einbaupläne für die technische Installation. XSEL (eXpert SELler) ist eine Ergänzung zu R1, die den Verkäufer unterstützen soll, eine Anlage so auszuarbeiten, daß sie alle Kundenwünsche erfüllt.

Die Beispiele zeigen nur einen kleinen Ausschnitt der wichtigsten Expertensystemanwendungen in den Vereinigten Staaten. Eine weiterführende Übersicht geben z.B. Mitchie (1982) oder Stede (1984).

30.3 Elemente von Expertensystemen

Die Übersicht über Entwicklungstendenzen im Bereich der Expertensysteme macht deutlich, welche Problemlösungskapazitäten durch diese Systeme erschlossen werden können. Es stellt sich jedoch die Frage nach dem grundsätzlichen Unterschied zwischen konventionellen, algorithmischen Programmen und Expertensystemen.

Konventionelle EDV-Anwendungen greifen in der Regel auf einen Datenpool zurück, in dem die grundlegenden Informationen gespeichert sind. Hinzu tritt als zweite Komponente die Problemlösungsvorschrift, d.h. das Programm, welches die Algorithmen zur Manipulation der Daten in codierter Form enthält. Bei Expertensystemen wird die zentrale Grundlage durch die **Knowledge Base** (Wissensbasis) gebildet (Übersicht 3 auf Seite 334). Die Wissensrepräsentation erfolgt einerseits wie in konventionellen Lösungen über die Datenbasis, in der **Fakten** oder **Zusicherungen** gespeichert sind. Als "zweites Bein" tritt die **Regelbasis** hinzu, in welcher mit Hilfe von Inferenzregeln die logischen Zusammenhänge zwischen den Fakten dargestellt werden.



Übersicht 3. Wissensbasis

Die Inferenzregeln werden durch die Problemlösungskomponente oder Inferenzmaschine aufgerufen und mit ihrer Hilfe durch logische Schlüsse neue Fakten oder Zusicherungen erzeugt. Die Inferenzmaschine führt somit die eigentliche Bearbeitung der vom Benutzer gestellten Aufgabe durch. Die Kommunikation erfolgt dabei über eine Kommunikationsschnittstelle, wobei zu erwarten ist, daß dieser Dialog bei höher entwickelten Systemen in natürlicher Sprache erfolgen wird. Zum Zwecke des Erwerbs und der Erweiterung vorhandenen Wissens wird weiterhin eine Wissenserweiterungskomponente benötigt. Entscheidende Bedeutung für die Adaption des Systems in der Praxis kommt letztlich der Erklärungskomponente zu, die dem Benutzer ermöglicht, durch Anfragen die Lösungswege und Schlußfolgerungen des Systems nachzuvollziehen.

30.3.1 Wissensrepräsentation

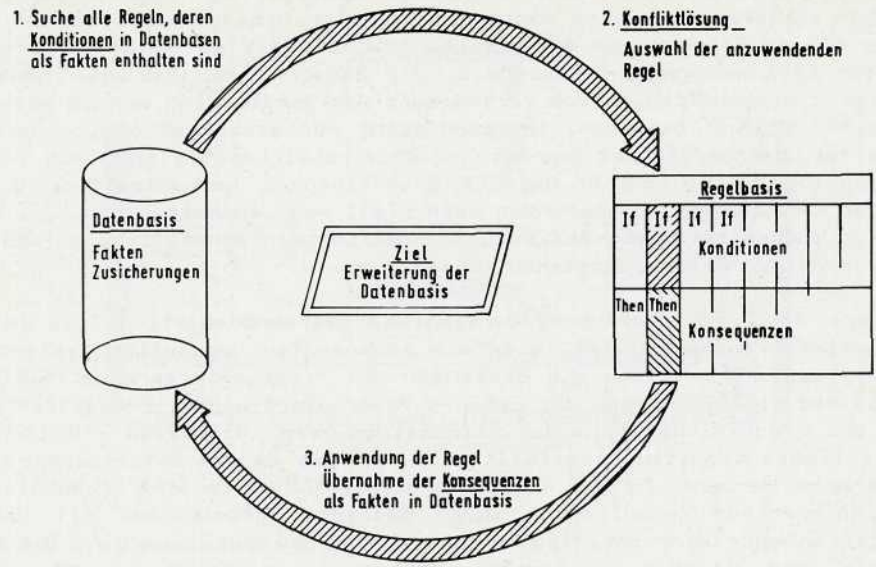
Mit Hilfe traditioneller, prozeduraler Programmiersprachen wie FORTRAN oder PASCAL stellt die Wissensrepräsentation im allgemeinen ein geringeres Problem dar, soweit sich die Probleme auf numerische oder quasi-numerische Sachverhalte zurückführen lassen. Erheblich schwieriger gestaltet sich hingegen die Darstellung der semantischen Aspekte von Daten, d.h. des Beziehungsgefüges von Daten untereinander. In konventionellen Lösungen werden diese Beziehungen über Algorithmen im Programmcode festgelegt. Die Fähigkeiten von Experten sind jedoch insbesondere deshalb geschätzt, weil sie in der Lage sind, neben den algorithmisch klar zu fassenden Zusammenhängen weitere Erkenntnisse durch ein hohes Maß an persönlicher Erfahrung und ausgeprägtem Assoziationsvermögen zu erschließen. Erfahrungswissen ist meist eng verbunden mit sogenanntem vagen Wissen und mit heuristischen Problemlösungswegen. Vages Wissen ist im Gegensatz zu Faktenwissen mit mehr oder minder großen Unsicherheiten behaftet. Heuristische Verfahren werden bei Problemen angewandt, für die keine eindeutige Lösungsstrategie vorliegt, bei denen jedoch aufgrund subjektiver Erfahrungen oder "Faustregeln" eine gewisse Wahrscheinlichkeit für den Erfolg bestimmter Strategien besteht.

Der Abbildung und Verarbeitung derartiger Wissensstrukturen widmet sich ein Zweig der KI-Forschung, der sich "Knowledge Engineering" nennt. Verschiedene Formen der deklarativen, also nicht-prozeduralen Wissensrepräsentation wurden im Rahmen dieser Forschungen entwickelt. LISP als die am meisten verbreitete Sprache für KI-Anwendungen erlaubt z.B. die Beschreibung von Daten durch eine Liste ihrer Eigenschaften. Durch Verknüpfung von Listen kann man zu sogenannten 'Semantischen Netzen' gelangen, in denen nicht nur einzelne Objekte durch ihre Eigenschaften gekennzeichnet werden, sondern gleichzeitig auch ein logisches Beziehungsgefüge verschiedener Objekte untereinander hergestellt wird. Diese Beziehungen können Mengenbeziehungen sein (Teil von, enthalten in ...), aber es lassen sich ebenso zeitliche Beziehungen oder Logiken darstellen, so daß letztlich sehr komplexe Gefüge entstehen können.

Eine andere Form der nicht-prozeduralen Wissensrepräsentation, die in vielen Expertensystemen enthalten ist, sind die sogenannten Production Systems (Produktionssysteme). Sie bilden die Basis für die Programmiersprache PROLOG, die weitläufig als die KI-Sprache der Zukunft favorisiert wird. Produktionssysteme bestehen aus Produktionsregeln der allgemeinen Form: Situation - Aktion, d.h. "falls bestimmte Konditionen erfüllt sind, können daraus entsprechende Konsequenzen gezogen werden". Im Fall von vagem Wissen läßt sich eine Produktionsregel ebenso einschränkend formulieren, indem mögliche Konsequenzen mit Unsicherheitskriterien oder Wahrscheinlichkeitsmerkmalen gekoppelt werden. Die Produktionsregeln sind in ihrer Repräsentationsform völlig unabhängig von dem tatsächlich aktuell verfügbaren Wissen. Die Verbindung von Produktionsregeln und Faktenwissen in der Wissensbasis wird über die Inferenzkomponente hergestellt. Die Inferenzmaschine hat die Aufgabe, geeignete Regeln auszuwählen und über deren Anwendung zu neuen Erkenntnissen zu gelangen. Dabei unterscheidet man grundsätzlich zwei unterschiedliche Strategien:

- Forward Chaining oder Datengetriebene Strategie
- Backward Chaining oder Zielgetriebene Strategie

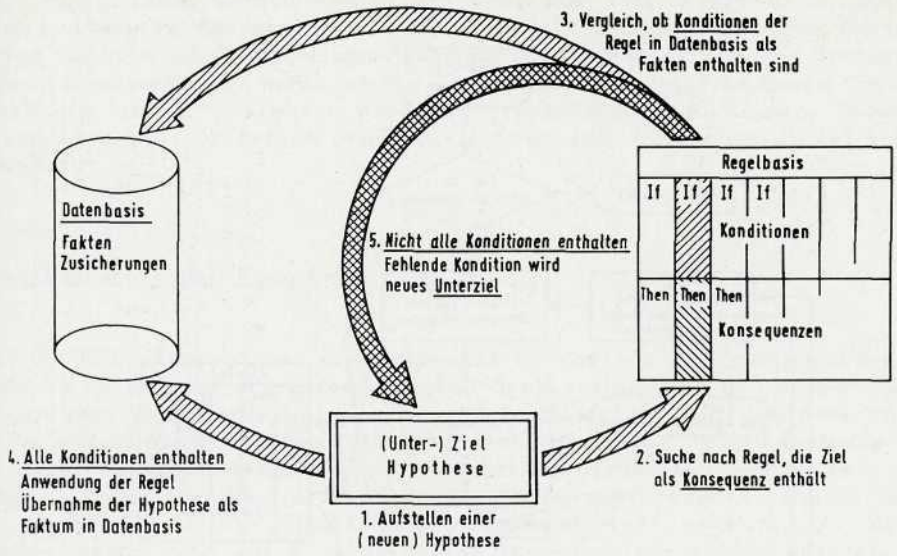
Bei der datengetriebenen Strategie bilden die vorhandenen Daten in Form von Fakten oder Zusicherungen den Ausgangspunkt (Übersicht 4 auf Seite 336). Die Inferenzkomponente wählt zunächst alle Regeln aus, deren Konditionsteil durch den Inhalt der Datenbasis verifiziert werden kann. Falls dies auf mehrere Regeln zutrifft, tritt ein Konfliktlöser in Aktion, der die zunächst anzuwendende Regel auswählt. Die durch die Anwendung der Regel sich ergebenden Konsequenzen werden als neue Fakten oder Zusicherungen der Datenbasis hinzugefügt, wodurch der Kreislauf geschlossen wird.



Übersicht 4. Forward Chaining

Eine Konfliktlösung kann nach verschiedenen Kriterien erfolgen. Die einfachste Form ist die Anordnung der Regeln in der Regelbasis nach der Reihenfolge der gewünschten Anwendung. Ebenso kann aber auch bestimmten Regeln bei festgelegten Datenkonstellationen der Vorzug eingeräumt werden; es können Regelanwendungen gegenseitig ausgeschlossen werden, etc..

Die zielgetriebene Strategie verfolgt den umgekehrten Weg (Übersicht 5 auf Seite 337). Ausgangspunkt ist zunächst ein Ziel, d.h. eine Hypothese, die das System in Hinblick auf die Lösung des Gesamtproblems aufstellt. Um diese Hypothese zu verifizieren, muß zunächst eine Regel gefunden werden, in der das Ziel als Konsequenz (Aktion) enthalten ist (Falls mehrere derartige Regeln vorliegen, tritt wieder der Konfliktlöser in Aktion). Die gefundene Regel ist daraufhin zu prüfen, ob sie angewandt werden kann, d.h., ob die Konditionen dieser Regel durch den Inhalt der Datenbasis abgedeckt sind. Falls dies gegeben ist, kann die Regel angewandt werden und entsprechend die Hypothese als neues Faktum (Zusicherung) in die Datenbasis aufgenommen werden. Fehlen jedoch Konditionen der Regel in der Datenbasis, so werden diese zunächst zum Unterziel, welches nach gleichem Schema erfüllt werden muß, wobei sich wiederum weitere Unterziele ergeben können, so daß es eventuell zu einer Schachtelung von Hypothesen kommen kann.
Der etwas kompliziertere funktionale Ablauf des "Backward Chaining" ist im Flußdiagramm in Übersicht 6 auf Seite 338 noch einmal aufgeschlüsselt. Ein simplifiziertes Beispiel, welches kaum einen Experten benötigt, soll den Ablauf verdeutlichen.
Das Beispielsystem besitzt nur 3 Regeln:



Übersicht 5. Backward Chaining

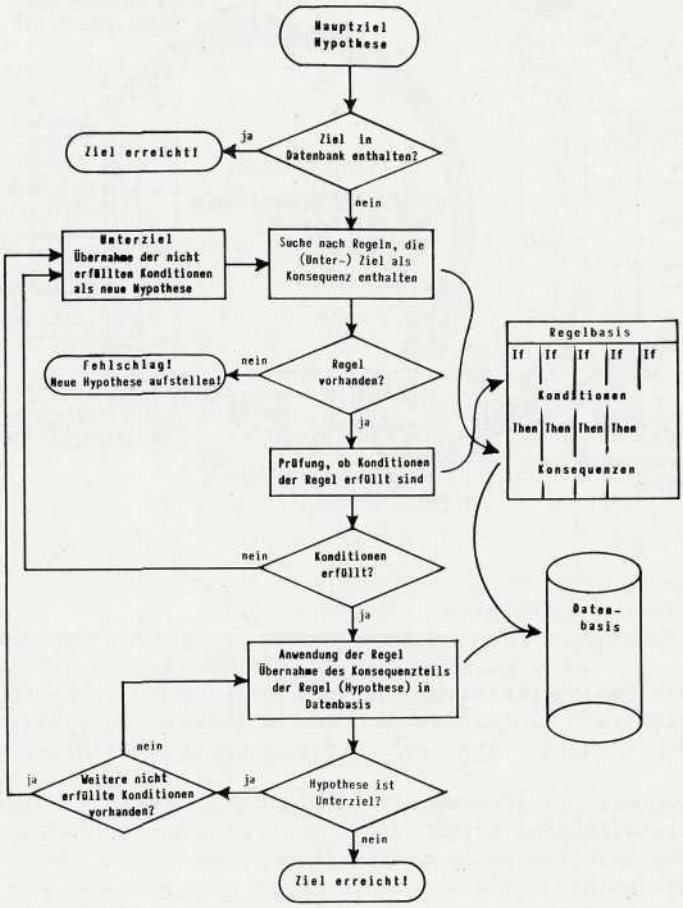
Regel 1: IF Kuh hat abgekalbt
AND nicht Kuh steht trocken
THEN Kuh steht in Laktation

Regel 2: IF Kuh seit 12 Stunden nicht gemolken
AND nicht Kuh ist krank
AND Kuh steht in Laktation
THEN Kuh melken

Regel 3: IF Kuh ist besamt
AND Trächtigkeitskontrolle positiv
AND 33. Trächtigkeitswoche überschritten
THEN Kuh steht trocken

Die Datenbasis hat folgenden Inhalt:
Kuh seit 12 Stunden nicht gemolken
Kuh hat abgekalbt
Kuh ist besamt
Kuh ist nicht krank
Trächtigkeitskontrolle positiv
nicht 33. Trächtigkeitswoche überschritten

Über Rückwärtsverkettung soll nun geprüft werden, ob die Kuh zu melken ist; die Hypothese lautet "Kuh melken". Eine überprüfung der Regeln ergibt, daß Regel Nr. 2 das Ziel als Konsequenz enthält. Ein Vergleich der Konditionen von Regel 2 mit der Datenbasis ergibt jedoch, daß nicht alle Konditionen in der Datenbasis enthalten sind. Die dritte Kondition von Regel 2 wird daher neues Unterziel: Kuh steht in Laktation. Um das Unterziel zu verifizieren, könnte Regel 1 angewandt



Übersicht 6. Flußdiagramm für Backwards Chaining

werden. Hier ist jedoch die zweite Kondition nicht erfüllt, so daß sich ein weiteres Unterziel ergibt. Durch Anwendung von Regel 3 kann dieses Unterziel als Zusicherung in die Datenbasis aufgenommen werden. Daraufhin läßt sich Regel 1 anwenden und letztlich auch das Hauptziel durch Anwendung von Regel 2 verwirklichen. Nach Anwendung der drei Regeln enthält die Datenbasis drei neue Fakten:

- nicht Kuh steht trocken
- Kuh steht in Laktation
- Kuh melken

Inwieweit datengetriebene oder zielgetriebene Strategien in einem Expertensystem zur Problemlösung angewandt werden, hängt von der Struktur des Problems ab.

Häufig ist es vorteilhaft, Systeme so aufzubauen, daß zunächst allgemeine Daten direkt vom Benutzer abgefragt und in der Datenbasis als Grundfakten gespeichert werden. Anschließend werden anhand der gegebenen Fakten mit Hilfe des Forward Chaining weitere Fakten ermittelt. Lassen sich aus der Summe der Fakten zielgerichtet bereits aussichtsreiche Hypothesen im Hinblick auf die Gesamtproblemstellung formulieren, so schaltet die Inferenzmaschine auf Backward Chaining um, und versucht so auf direktem Wege zur Problemlösung zu kommen. Dabei werden selbstverständlich zusätzlich benötigte Fakten auch weiterhin direkt vom Benutzer abgefragt.

30.4 Aufbau von Expertensystemen

Zum Bau von Expertensystemen sind generell mindestens 2 Personen erforderlich: der Experte selbst mit dem erforderlichen Spezialwissen in der Wissensdomäne und der "Knowledge Engineer", welcher es versteht, das Wissen des Experten zu extrahieren und zu einem nutzungsfähigen System umzuformen. Dem Knowledge Engineer stehen dabei zunächst spezifische Programmiersprachen wie PROLOG und LISP zur Verfügung. Diese Sprachen erleichtern die Programmierarbeit immens, doch wird dieser Vorteil durch ein erhebliches Effizienzdefizit erkauft. Im KI-Bereich wird daher nicht mehr auf konventionelle Time-Sharing-Systeme gesetzt, sondern es wurden spezifische Rechner für KI-Anwendungen konfiguriert. Diese persönlichen Rechenanlagen erfordern derzeit noch Investitionen zwischen ca. DM 100.000 und DM 250.000.

Darüberhinaus gelangen inzwischen sogenannte Expertensystem-Shells auf den Markt. Eine Shell stellt ein Rahmen-Expertensystem dar, welches bereits die gesamte Systemarchitektur fertig enthält. Dem Knowledge Engineer obliegt es einzig, diese Shell mit den domänenspezifischen Daten, Regeln und Begründungen zu füllen. Der Prototyp der Expertensystem-Shells wurde EMYCIN, welche aus dem MYCIN-Projekt in Stanford hervorgegangen ist. Eine Reihe amerikanischer Entwicklungen befinden sich auf dem Markt, und NIXDORF bietet mit TWAICE ebenfalls eine erste deutsche Expertensystem-Shell an.

Durch diese Vereinfachung der Wissensimplementierung wird sicherlich der Entwicklung weiterer Expertensysteme Vorschub geleistet. Dies darf jedoch nicht darüber hinwegtäuschen, daß das eigentliche Problem die Auswahl der geeigneten Aufgabe und die Wissensaquisition selbst darstellt. Trotz einer großen Zahl von Entwicklungen sind die meisten bestehenden Systeme von vornehmlich akademischem Wert, da sie entweder triviale Probleme behandeln oder die falsche Zielgruppe ansprechen oder unzureichend formuliert sind.

Das Problem der Wissensaquisition selbst wird von Anfängern im Bereich des Knowledge Engineering oft völlig unterschätzt. Experten zeigen oft die überraschende Eigenschaft, daß sie zwar hervorragende Problemlösungskapazitäten aufweisen, sich jedoch als total unfähig erweisen bei der Beschreibung des Lösungsweges, wie sie zu ihrem Ergebnis gekommen sind. Experten arbeiten meist sehr stark mit Assoziationen, die für den Laien oder Pseudo-Experten kaum nachvollziehbar sind. Ebenso lösen sie Probleme oft nicht Schritt für Schritt in einer causal-logischen Folge, sondern arbeiten sprunghaft unter Auslassung von Teilschritten. Die meisten Experten tendieren dahin, neue Probleme in der Weise anzugehen, daß sie sie mit ihnen bekannten Situationen zu identifizieren versuchen. Erst wenn alle dem Laien mehr zufällig erscheinenden Assoziationen keinen Lösungsweg versprechen, wird der Experte den "mühsamen" Weg der auf grundsätzlichen Prinzipien aufbauenden prozeduralen Lösung einschlagen. Aufgrund dieser

Beobachtung haben Informationstheoretiker geschlossen, daß Experten Probleme eigentlich nicht sehr "intelligent" lösen. Tatsächlich ist ihr Lösungsweg eher als "intuitiv" zu charakterisieren, und es ist daher die wohl schwierigste Aufgabe beim Bau eines echten Expertensystems, diese Lösungswege zu analysieren und in Daten und Regeln zu fassen.

Die zunehmende Popularität "Wissensverarbeitender Systeme" in Deutschland läßt jedoch darauf hoffen, daß zukünftig auch verstärkt Ausbildungsmöglichkeiten in diesem Bereich angeboten werden und so vielleicht einmal in Übersetzung des "Knowledge Engineers" der Beruf des "Wissensingenieurs" kreiert wird.

30.5 Literatur

1)Buchanan, B., Shortliffe, E., 1984
Rule-based Expert Systems
Addison-Wesley

2)Duda, R., et.al., 1981
Development of the Prospector Consultation System for Mineral Exploration
SRI International

3)Hayes-Roth, F., 1983
Building Expert Systems
Addison-Wesley, London

4)Hayes-Roth, F., Waterman, D.A., Lenat, D.B., 1980
Heuristic Programming Projekt
Stanford University

5)Johnson, P.,1984
The Expert mind: A new challenge for the Information Scientist
Beyond Produktivity. North Holland, Elsevier Science Publ.

6)Mc Dermott, J., 1980
A Rule-based Configurer of Computer Systems
Computer Science Department, Carnegie-Mellon University

7)Michie, D., 1982
Introductory Readings in Expert Systems
New York, Gordon and Breach

8)Newell, A., Simon, H., 1972
Human Problem Solving
Englewood Cliffs, Prentice-Hall

9)Noelke, Savory, 1984
Prologsysteme im Vergleich
Angewandte Informatik, 3, pp 108-112

10)Savory, S., 1985
Künstliche Intelligenz und Expertensysteme
Oldenbourg Verlag, München

11)Slagle, J.R., 1963
A heuristic program that solves symbolic integration problems in freshman calculus
ACM Vol 10, pp 507-520

12)Suppes, P., 1971
Computer-assisted instructions at Stanford
Technical Rep. No. 174, Institut for Mathematical Studies in the Social Science, Stanford University

13)Stede, M., 1983
Einführung in die künstliche Intelligenz
Sprendlingen, Luther-Verlag