

Intelligente Systeme

Wissensbasierte Systeme

Prof. Dr. R. Kruse C. Braune C. Moewes

{kruse, braune, cmoewes}@iws.cs.uni-magdeburg.de

Institut für Wissens- und Sprachverarbeitung

Fakultät für Informatik

Otto-von-Guericke Universität Magdeburg

Übersicht

- 1. Wissensbasierte Systeme**
2. MYCIN – Verallgemeinertes regelbasiertes System
3. Wissensrepräsentation und Schlussfolgern
4. Wissensmodellierung
5. Epistemische Logik

Wissensrepräsentation und Schlussfolgern

aus der *kognitiven Psychologie* ist bekannt, dass menschliches Problemlösen und Lernen folgende Dinge involviert:

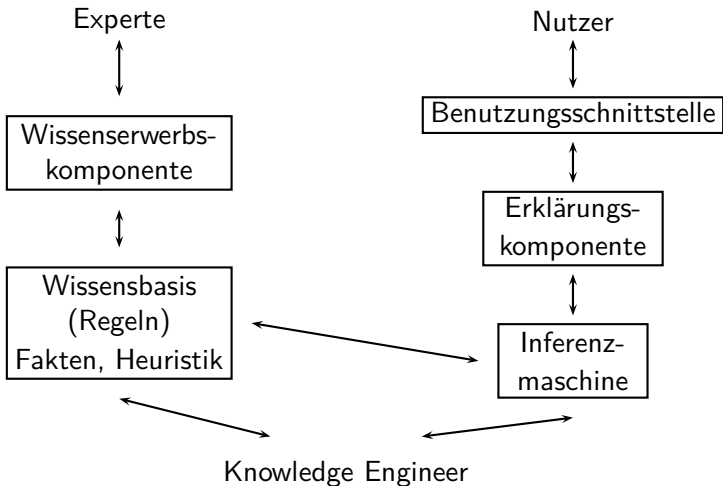
- Repräsentation und Ausnutzung von *mehreren Arten von Wissen*
- Kombination von *verschiedenen Methoden zum Schlussfolgern*:
regelbasiertes Schließen, modellbasiertes Schließen,
bedingungs-basiertes (constraint-based) Schließen

Fragen:

- Wie können die reale Welt und das Schlussfolgern mit dem Computer modelliert werden?
- Wie kann man Kenntnisse der Technologien gewinnen, die zum Entwurf von wissensbasierten Systemen gebraucht werden?

wissensbasiertes System = System, dass mittels Wissen schlussfolgern kann

Wissensbasierte Systeme



Beispiel: „Darlehensvergabe“

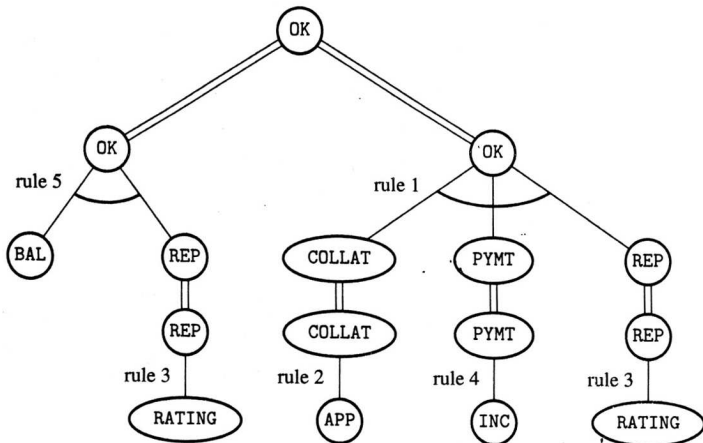
<i>OK</i>	Darlehen sollte bewilligt werden.
<i>COLLAT</i>	Sicherheit für Darlehen ist genügend.
<i>PYMT</i>	Antragsteller kann Darlehenszahlungen tilgen.
<i>REP</i>	Antragsteller hat gute finanzielle Reputation.
<i>APP</i>	Beurteilung der Sicherheit ist ausreichend größer als Größe des Darlehens.
<i>RATING</i>	Antragsteller hat gute Bonität.
<i>INC</i>	Einkommen des Antragstellers ist größer als seine Ausgaben.
<i>BAL</i>	Antragsteller hat ausgezeichnete Bilanz.

Beispiel: „Darlehensvergabe“

Regeln zur Entscheidungsfindung:

- $\text{COLLAT} \wedge \text{PYMT} \wedge \text{REP} \Rightarrow \text{OK}$
- $\text{APP} \Rightarrow \text{COLLAT}$
- $\text{RATING} \Rightarrow \text{REP}$
- $\text{INC} \Rightarrow \text{PYMT}$
- $\text{BAL} \wedge \text{REP} \Rightarrow \text{OK}$

Entscheidungsbaum (Und/Oder-Baum)



Modi der Verwendung

Anfragemodus:

- Fakten werden aus Datenbank geholt oder Benutzer wird gefragt

Erklärungsmodus:

- System beantwortet Fragen, wie Entscheidung gefällt wurde

Beispiele:

- Nutzer: „Warum glaubst du, dass *das Darlehen bewilligt werden soll?*“
- System: „Weil *die Sicherheit für das Darlehen genügend ist, und der Antragsteller die Darlehenszahlungen tilgen kann, und der Antragsteller eine gute finanzielle Reputation hat.*“

Anfrage- und Erklärungsmodus

- Nutzer: „Warum glaubst du, dass *der Antragsteller eine gute finanzielle Reputation hat?*“
- System: „Weil die Schufa gesagt hat, dass *der Antragsteller eine gute Bonität hat.*“
- Nutzer: „Warum hast du mich gefragt, ob *das Einkommen des Antragstellers größer ist als seine Ausgaben* oder nicht?“
- System: „Weil ich versucht habe, herauszufinden ob *der Antragsteller die Darlehenszahlungen tilgen kann* oder nicht.“
- Nutzer: „Warum versuchst du herauszufinden, ob *der Antragsteller die Darlehenszahlungen tilgen kann* oder nicht?“
- System: „Weil es eines der Kriterien zum Herausfinden ob *das Darlehen bewilligt werden sollte* oder nicht.“
- Nutzer: „Warum versuchst du herauszufinden, ob *das Darlehen bewilligt werden sollte* oder nicht.“
- System: „Weil du mich gefragt hast?!“

Wissensbasierte Systeme

- einzelne Regel = (im Idealfall) Wissenseinheit
- Menge von Regeln zusammen mit Ausführungsstrategie = Programm für Lösung eines Problems oder Problemklasse
- Prolog-Programm: *Regelbasiertes System*
- regelbasierte Systeme: historisch erste Formen von KI-Systemen
- lange Zeit als prototypisch für *Expertensysteme* angesehen
- inzwischen: nicht jedes Expertensystem benutzt Regeln als Hauptinferenzmechanismus
- Regelmodule: wichtige Bestandteile von Expertensystemen
- Regelsysteme gewinnen erneut Bedeutung
z.B. bei Kodifizierung von *Geschäftsprozess-Regeln*

XCON/R1

Beispiel für ein Konfigurationssystem

- *XCON/R1*: Werkzeug für Konfiguration von *DEC Vax-Computern*
- seit ca. 1980 entwickelt und löst folgende Aufgaben:
 - Identifikation fehlender Komponenten,
 - Platzierung der Komponenten bezüglich Bussen, Schnittstellen und Gehäusen
- implementiert in *OPS5* (vorwärtsverkettetes regelbas. System)
- eine der ersten erfolgreichen kommerziellen Anwendungen von regelbasierten Expertensystemen
- hatte großen Einfluss auf industrielles Interesse an Expertensystemen
- wurde ständig weiterentwickelt: heute ca. 10.000 Regeln
- löst routinemäßig Aufgaben mit 100–200 Komponenten

Beispiel einer XCON/R1 Regel

IF

the most current active context is distributing massbus devices, and there is a single-port disk drive that has not been assigned to a massbus, and there are no unassigned dual-port disk drives, and the number of devices that each massbus should support is known, and there is a massbus that has been assigned at least one disk drive and that should support additional disk drives, and the type of cable needed to connect the disk drive to the previous device on the massbus is known

THEN

assign the disk drive to the massbus.

Beispiel: Implementierung über Prolog

„Roboterarm bewegt sich, wenn Batterie i.O. ist und Gewicht nicht zu schwer ist“

- Programm entspricht der Aussage:

$$\begin{aligned} & (\text{BAT_OK} \wedge \text{LIFTABLE} \wedge \\ & (\text{BAT_OK} \wedge \text{LIFTABLE} \Rightarrow \text{MOVES})) \Rightarrow \text{MOVES} \end{aligned}$$

:- MOVES

Ziel (Anfrage)

BAT_OK :-

Fakten

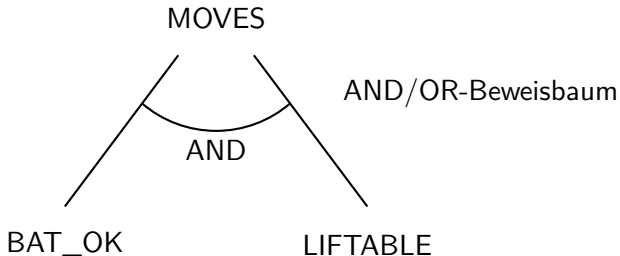
Prolog beweist

LIFTABLE :-

MOVES :- BAT_OK, LIFTABLE Regeln

Aussage mit „depth-first“ und „backward chaining“

Beispiel: Schlussfolgern mit Hornklauseln



Anmerkung:

- Klausel 1 gegen Klausel 2 oder Klausel 3 geht nicht
- aber Klausel 1 vs. Klausel 4 liefert 2 neue Ziele (siehe Baum)
- diese können sofort bewiesen werden

Schlussfolgern mit Hornklauseln

Beispiel: Blockwelt

Ist Bauklotz A über C , wenn A auf B und B auf C ?

Prolog:

$:-$ Above(A, C)

On(A, B) $:-$

On(B, C) $:-$

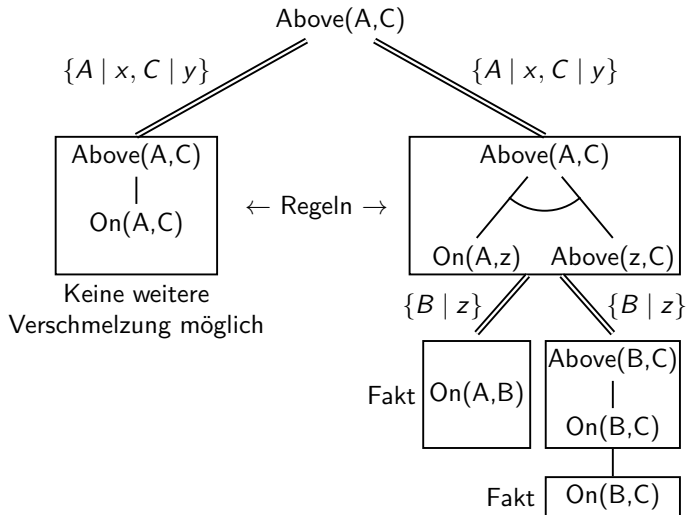
Above(x, y) $:-$ On(x, y)

Above(x, y) $:-$ On(x, z), Above(z, y)

PK1:

$$(\forall x, y, z) \text{ On}(x, y) \Rightarrow \text{Above}(x, y)$$
$$(\forall x, y) (\exists z) \text{ On}(x, y) \wedge \text{Above}(z, y) \Rightarrow \text{Above}(x, y)$$

Schlussfolgeren mit Hornklauseln



Übersicht

1. Wissensbasierte Systeme
- 2. MYCIN – Verallgemeinertes regelbasiertes System**
3. Wissensrepräsentation und Schlussfolgern
4. Wissensmodellierung
5. Epistemische Logik

MYCIN – Verallg. regelbasiertes System

- medizinisches Diagnosesystem, das auch unsicheres Wissen verarbeitet
- Wissensrepräsentation mittels Regeln
- Regeln sind mit Unsicherheitsfaktor (certainty factor) behaftet, der zwischen -1 und +1 liegt

Beispielregel 35:

PREMISE: (\$AND (SAME CNTXT GRAM GRAMNEG)
(SAME CNTXT MORPH ROD)
(SAME CNTXT AIR ANAEROBIC))

ACTION: (CONCLUDE CNTXT IDENTITY BACTEROIDES TALLY .6)

MYCIN – Verallg. regelbasiertes System

Bedeutung:

- IF: 1) The gram stain of the organism is gramneg, and
2) The morphology of the organism is rod, and
3) The aerobicity of the organism is anaerobic,

THEN: There is suggestive evidence (0.6)
that the identity of the organism is
bacteroides

Sicherheitsfaktoren (CF): *Grad des Glaubens* (degree of belief) an eine Hypothese

- $CF(A \rightarrow B) > 0$: A unterstützt B
- $CF(A \rightarrow B) = 1/-1$: B ist definitiv wahr bzw. falsch, wenn A vorliegt
- $CF(A \rightarrow B) < 0$: A liefert Evidenz gegen B
- $CF(A \rightarrow B) = 0$: indifferente Haltung

MYCIN – Notationen

- $CF(A \rightarrow B) \in [-1, 1]$ Sicherheitsfaktor von Regel $A \rightarrow B$
- $CF[B, \{A\}]$ Sicherheitsfaktor der Konklusion B auf Basis der CF der Regel $A \rightarrow B$ und Prämisse A
- i.A. $CF[B, \{A_1, \dots, A_n\}]$ Sicherheitsfaktor von B auf Basis der n Regeln $A_1 \rightarrow B, \dots, A_n \rightarrow B$ und n Prämissen A_1, \dots, A_n
- $CF[B]$ Gesamtsicherheitsfaktor von B bezüglich Regelbasis,

$$CF[B] := CF[B, \{A \mid A \rightarrow B \text{ gehört zur Regelbasis}\}]$$

MYCIN – Beispiel

$$A \rightarrow B [0.80]$$

$$C \rightarrow D [0.50]$$

$$B \wedge D \rightarrow E [0.90]$$

$$E \vee F \rightarrow G [0.25]$$

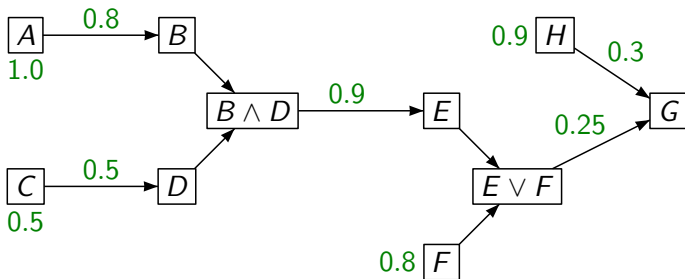
$$H \rightarrow G [0.30]$$

$$A [1.00]$$

$$C [0.50]$$

$$F [0.80]$$

$$H [0.90]$$



MYCIN – Propagationsregeln

Konjunktion

$$CF[A \wedge B] = \min\{CF[A], CF[B]\}$$

Disjunktion

$$CF[A \vee B] = \max\{CF[A], CF[B]\}$$

serielle Kombination

$$CF[B, \{A\}] = CF(A \rightarrow B) \cdot \max\{0, CF[A]\}$$

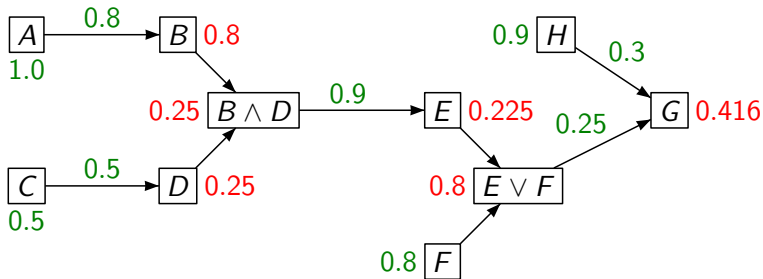
parallele Kombination: für $n > 1$ ist

$$CF[B, \{A_1, \dots, A_n\}] = f(CF[B, \{A_1, \dots, A_{n-1}\}], CF[B, \{A_n\}])$$

wobei Funktion $f : [-1, 1]^2 \rightarrow [-1, 1]$ wie folgt definiert:

$$f(x, y) := \begin{cases} x + y - xy & \text{wenn } x, y > 0, \\ x + y + xy & \text{wenn } x, y < 0, \\ \frac{x+y}{1-\min\{|x|, |y|\}} & \text{sonst.} \end{cases}$$

MYCIN – Beispiel (Fortsetzung)



$$f(0.3 \cdot 0.9, 0.25 \cdot 0.8) = 0.27 + 0.2 - 0.27 \cdot 0.2 = 0.416$$

MYCIN – Funktionsweise

- Arzt oder Benutzer bekommt während Konsultation vom System Fragen gestellt
- nach ca. 50–60 Fragen gibt MYCIN hypothetische Diagnose aus
- passende Therapie kann dann darauf basieren
- starke Erklärungskomponente (*question answering module*), wird bei Konsultation verwendet
- Argumentationskette lässt sich transparent machen durch *reasoning status checker*

MYCIN – Probleme mit Sicherheitsfaktoren

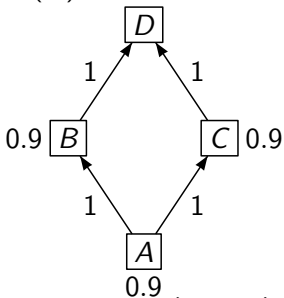
- ethische und rechtliche Schwierigkeiten — z.B. Verantwortung bei Fehldiagnose
- sehr stark vom behandelnden Arzt abhängig, sehr zeitaufwendig (> 30 min in einer typischen Sitzung)
- zur damaligen Zeit (1970–1975) technisch sehr aufwendig aufgrund der fehlenden Rechenleistung und Vernetzung
- dennoch: sehr wertvoller Ansatz zur Demonstration der Mächtigkeit dieser Art der Wissensrepräsentation und Schlussfolgerung
- Problem auch heute noch: Expertenwissen für derartiges System verfügbar zu machen

War MYCIN ein Fehler?

- MYCIN liefert manchmal wenig plausible Ergebnisse

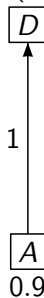
Beispiel: $CF(A) = 0.9$, $CF(D) = ?$

$$CF(D) = 0.9 + 0.9 - 0.9 \cdot 0.9 = 0.99$$



vs.

$$CF(D) = 0.9$$



CF ist zu hoch, weil (gleiche) Evidenz auf verschiedenen (parallelen) Pfaden übertragen wird!

Übersicht

1. Wissensbasierte Systeme
2. MYCIN – Verallgemeinertes regelbasiertes System
- 3. Wissensrepräsentation und Schlussfolgern**
4. Wissensmodellierung
5. Epistemische Logik

Wonach suchen wir, um eine Maschine zu verkörpern?

- Geist
- Gedanken
- Intelligenz
- Rationalität
- Neuroanatomie
- **Wissen**

Pyramide des Wissens

Giarratano und Riley (2005)

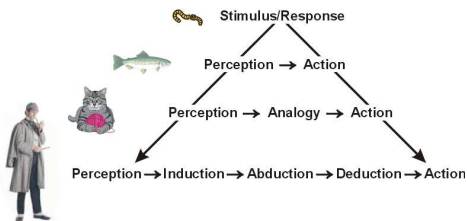


- Weisheit: Nutzen von Wissen in einer vorteilhaften Weise
- Metawissen: Regeln über Wissen
- Wissen: Regeln über das Nutzen von Informationen
- Informationen: potentiell nützlich für Wissen
- Daten: potentiell nützliche Informationen
- Rauschen: keine offensichtlichen Informationen

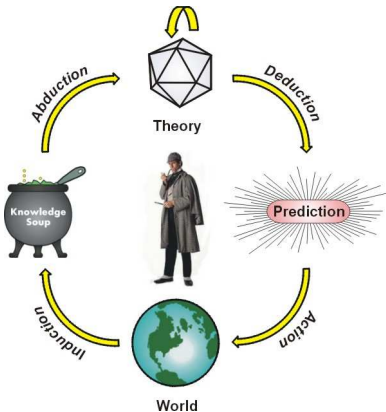
Intellektuelle Ursprünge

- vor 2000 Jahren: der **Syllogismus** (Logik von Aristoteles, Konklusionen)
- im 17. Jhd.: die **Algebra der Gedanken** (Leibniz: menschliches Schlussfolgern kann auf Berechnungen zurückgeführt werden, regelbasierte Manipulation)
- im 19. Jhd.: **Boole'sche Logik**
- auch 19. Jhd.: **analytische Maschine** (Babbage, Ada Lovelace)
- im 20. Jhd.: Turings Ideen über Gedanken und Berechenbarkeit
- 1927 (McCarthy): Schlussfolgern kann auf Berechnungen reduziert werden

Kognitive Prozesse



Entwicklung der Kognition



Peirce'sche Logik
des Pragmatismus

Beispiel: Das unordentliche Zimmer

von D.M. Gabbay (2010)

Eine Mutter geht in das Schlafzimmer ihrer Teenager-Tochter. Ihr erster Eindruck ist, dass dort eine große Unordnung herrscht. Es liegt überall irgendetwas herum. Der Eindruck der Mutter ist, dass dies untypisch für ihre Tochter sei, sich so zu verhalten. Was war passiert?
Vermutung: Ihre Tochter hat Probleme mit einem Jungen.

Weitere Analyse: Die Mutter bemerkt ein zusammengebrochenes Regal. Hat ihre Tochter es zerbrochen? Nach weiterer Betrachtung bemerkt sie anhand des Musters der Unordnung, dass das Regal zusammenbrach durch das zu hohe Gewicht und einfach alles verteilte. Aber eigentlich ist das keine Unordnung, sondern ergibt (anziehungskräftegemäß) Sinn.

Beispiel: Das unordentliche Zimmer

verschiedene Modi von Schlussfolgerungen werden hier genutzt:

1. **Schließen mittels neuronaler Netze;** Sie erkennt die Unordnung sofort, so wie wir ein Gesicht erkennen.
2. **nicht-monotone Deduktion:** Die Mutter schließt aus dem Kontext und ihrem Wissen über ihre Tochter, dass das Mädchen nicht so unordentlich ist. „Was ist passiert?“ fragt sie demnach.
3. **Abduktion:** Sie bietet eine sinnvolle Erklärung an, dass das Mädchen Probleme mit einem Jungen hat: Es ist normal in dem Alter.
4. **Deduktion:** Dann wendet sie Deduktion an und erkennt, dass die Unordnung von der Schwerkraft herrührt. Diese Deduktion ist nicht mehr ein „Neuronaler Netz“-Eindruck. Es ist eine sorgfältige Berechnung.

Merke: Punkt 4 kann ein „Neuronaler Netz“-Eindruck sein: jemand der viele Fälle von zerbrochenen Regalen sieht, erkennt dieses Muster.
Punkt 2 könnte auch ein Bayes-Netz gewesen sein.

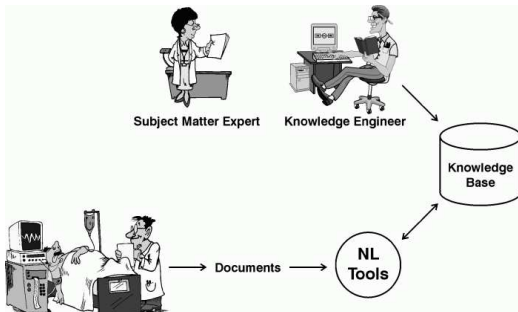
Übersicht

1. Wissensbasierte Systeme
2. MYCIN – Verallgemeinertes regelbasiertes System
3. Wissensrepräsentation und Schlussfolgern
- 4. Wissensmodellierung**
5. Epistemische Logik

Knowledge Engineer

seine Aufgabe: Entwurf von effizienten regelbasierten Systemen

- arbeitet mit den Experten zusammen und beobachtet diese, um Expertisen zu extrahieren und um diese in eine Menge von Regeln zu implementieren
- kennt sich aus mit Expertensystemen, aber nicht notwendigerweise in der Anwendungsdomäne
- kennt die Möglichkeiten der Technologien und weiß sie zu nutzen



Vergleich zur klassischen Programmierung

Eigenschaft	klass. Programmierung	Expertensystem
gesteuert von...	Anordnung der Aussagen	Inferenzmaschine
Regelung und Daten	implizierte Integration	explizite Separation
Lösung durch...	Algorithmus	Regeln
Eingabe	als richtig angenommen	unvollständig, falsch
unerwartete Eingabe	schwer händelbar	sehr zugänglich
Ausgabe	immer richtig	variiert mit dem Problem
Erklärung	generell sequentiell	opportunistische Regeln
Programmwurf	strukturiert	kaum oder keine Struktur
Modifizierbarkeit	schwer	möglich
Erweiterung	in großen Sprüngen	inkrementell

Vergleich zur klassischen Programmierung

Algorithmen + Datenstrukturen = Programme

Wissen + Inferenz = wissensbasiertes System

Übersicht

1. Wissensbasierte Systeme
2. MYCIN – Verallgemeinertes regelbasiertes System
3. Wissensrepräsentation und Schlussfolgern
4. Wissensmodellierung
- 5. Epistemische Logik**

Epistemische Logik

Anna zieht eine Karte von einem Kartenstapel mit drei verschiedenen Karten (0, 1, 2). Sie zieht die Karte 0 ohne es zu wissen. Karte 1 wird zurück in den Stapel gelegt. Karte 2 wird mit dem Bild nach unten auf den Tisch gelegt. Nun schaut Anna sich ihre Karte an. **Was weiß Anna?**

- Anna hat die Karte 0.
- Anna weiß nicht, dass Karte 1 auf dem Tisch liegt.
- Anna betrachtet es als möglich, dass Karte 1 auf dem Tisch liegt.
- Anna weiß, dass entweder Karte 1 oder Karte 2 in dem Stapel liegt.
- Anna kennt ihre eigene Karte.

Beispiel: Beschreibung des Wissens

- es gibt einen Agenten Anna: $\{a\}$
- aussagenlogische Variable q_a für „Karte q wird von Anna gehalten“ mit $q = 0, 1, 2$
- $K_a\varphi$ bedeutet „Anna weiß, dass φ “
- $\hat{K}_a\varphi$ ($\neg K_a\varphi$) bedeutet „Anna betrachtet es als möglich, dass φ “
- Anna hat die Karte 0: 0_a
- Anna weiß, dass sie die Karte 0 hat: K_a0_a
- Anna weiß nicht, dass Karte 1 auf dem Tisch liegt: $\neg K_a1_t$
- Anna betrachtet es als möglich, dass Karte 1 nicht auf dem Tisch liegt: $\neg K_a\neg 1_t$ (sie weiß nicht, dass nicht 1_t)
- Anna weiß, dass entweder Karte 1 oder Karte 2 in dem Stapel liegt: $K_a(1_h \vee 2_h)$
- Anna kennt ihre eigene Karte: $K_a0_a \vee K_a1_a \vee K_a2_a$

Warum geht es bei epistemischer Logik?

Episteme (Griechisch) = Wissen

Ich weiß, dass p

$K_a p$

Er weiß nicht, dass p

$\neg K_b p$

Er weiß, ob p

$K_b p \vee \neg K_b p$

Er weiß, dass ich weiß, dass sie weiß nicht, dass p

$K_b K_a \neg K_c p$

Sprache:

$$\varphi := p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid K_a \varphi$$

Ein Beispiel: Anja weiß, dass Ben weiß, dass Anja p weiß. Ben weiß nicht, dass Anja weiß, dass Ben nicht weiß, dass Anja p weiß.

Die Merkmale des Wissens (Axiome)

1. Distributionsaxiom (K): wenn ein Agent p weiß und weiß, dass $\varphi \rightarrow \psi$, dann muss der Agent auch ϕ wissen:
$$K_a\varphi \wedge K_a(\varphi \rightarrow \psi) \rightarrow K_a\psi$$
2. Wahrheitsaxiom (T): wenn ein Agent Fakten weiß, dann müssen diese Fakten wahr sein (Wissen impliziert Richtigkeit): $K_a\varphi \rightarrow \varphi$
3. Wissensgeneralisierungsregel (N): wenn φ wahr ist in jeder Welt, die ein Agent als mögliche Welt ansieht, dann muss der Agent φ in jeder möglichen Welt kennen: $M \models \varphi \rightarrow M \models K_i\varphi$
4. Positive Selbstwahrnehmung (KK): Agenten wissen, was sie wissen: $K_a\varphi \rightarrow K_aK_a\varphi$
5. Negative Selbstwahrnehmung: Agenten wissen, was sie nicht wissen: $\neg K_a\varphi \rightarrow K_a\neg K_a\varphi$

Logische Allwissenheit

$$K_a\varphi \wedge K_a(\varphi \rightarrow \psi) \rightarrow K_a\psi$$

Unser epistemische Agent a kennt alle seine logischen Konsequenzen. Wenn Q eine logische Konsequenz von P ist, dann gibt es keine mögliche Welt, wo P wahr ist, aber Q nicht.

Beispiel:

Wenn a weiß, dass Primzahlen nur durch sich selbst und 1 teilbar sind, dann weiß a , dass 8683317618811886495518194401279999999 eine Primzahl ist.

Mögliche Weltbedeutung

- Teile Menge der mögl. Welten ein in solche, die kompatibel sind mit dem Wissen des Agenten, und solche die es nicht sind.
- $K_a\varphi$: in allen möglichen Welten, die kompatibel zum Wissen von a sind, gilt φ
- Um den Gedanken auszudrücken, dass für Agenten a die Welt w' kompatibel ist mit seinem Informationszustand oder dass sie **zugänglich** ist von der möglichen Welt w , in der sich a momentan befindet, benötigt man eine Zugänglichkeitsrelation R (\rightarrow), um zwischen w und w' zu entscheiden.
- $w \rightarrow w'$ bedeutet, dass wenn man in der anderen Welt w lebt, dass w' dann eine der scheinbaren Welten ist, die a für möglich hält.
- Reflexivität besagt, dass eine Relation $w \rightarrow w'$ von jeder Welt zu sich selbst gibt. Reflexivität bedeutet, dass die eigentliche Welt immer eine der Welten ist, die als möglich betrachtet wird.

Kripke-Modell

Ein *Kripke-Modell* ist eine Struktur $M = \langle S, R, V \rangle$ wobei

1. Zuständen (oder möglichen Welten) enthalten sind in $S \neq \emptyset$ und
 2. R eine Zugänglichkeitsrelation $R_a \subseteq S \times S$ für jedes $a \in A$ induziert (R soll die möglichen Welten von jedem a abspeichern)
 3. Beurteilung(sfunktion) $V : P \rightarrow 2^{|S|}$ sei eine Interpretationsfunktion, die bestimmt welche Sätze der Sprache in welchen Welten (Zuständen) wahr sind.
- Falls R_a aus M eine *Äquivalenzrelation* ist, dann nennen wir M ein **epistemisches Modell**. In diesem Fall schreiben wir \sim_a anstatt R_a und repräsentieren das Modell als $M = \langle S, \sim, V \rangle$.
 - Die Wahrheitszuweisung erklärt uns, ob eine Aussage $p \in P$ wahr oder falsch ist in bestimmten Zuständen. Nur, weil in einer Welt etwas wahr ist, heißt das nicht, dass es in einer anderen Welt ebenfalls ist. Um zu zeigen, dass φ in einer bestimmten Welt wahr ist, schreiben wir $(M, s) \models \varphi$

Epistemisches Modell

Gegeben sei eine Beschreibung einer Situation

Der Modellierer folgende Dinge versucht zu bestimmen:

- die Menge der relevanten Aussagen P
- die Menge der relevanten Agenten A
- die Menge der Zustände S
- eine Ununterscheidbarkeitsrelation R_a über diese Zustände eines jeden Agenten a

Beispiel: Epistemisches Modell

- $S = \{012, 021, 102, 120, 201, 210\}$
- $\sim_a = \{(012, 012), (012, 021), (021, 021), \dots\}$
- $V(0_a) = \{012, 021\}, V(1_a) = \{102, 120\}, \dots$

Hexa1, 012 $\models K_a 0_a$

\iff

für alle $t : 012 \sim_a t$ impliziert Hexa1, $t \models 0_a$

\iff

Hexa1, 012 $\models 0_a$ und Hexa1, 021 $\models 0_a$

\iff

$012 \in V(0_a) = \{012, 021\} \wedge 021 \in V(0_a) = \{012, 021\}$

	<u>012</u>	021
102	a	120
201		210

Beispiel: Zwei Agenten

Anna und Ben ziehen die Karten 0 und 1 aus den Karten 0,1,2.
Karte 2 wird mit dem Bild nach unten auf den Tisch gelegt.

<u>012</u>		021
102	a	120
201		210

- Ben hält es nicht für möglich, dass Anna die Karte 1 hat: $\neg \hat{K}_b 1_a$
- Anna hält es für möglich, dass Ben es für möglich hält, dass sie die Karte 1 hat: $\hat{K}_a \hat{K}_b 1_a$
- Anna weiß, dass Bill es für möglich hält, dass sie die Karte 0 hat: $K_a \hat{K}_b 0_a$

Das Rätsel der weisen Personen

Teilnehmer: Abelard (A), Heloise (H), der König (K).

Als Allgemeinwissen sei ihnen allen folgendes bekannt:

- Es gibt drei Hüte: zwei rote Hüte und einen weißen Hut.
- Der König setzt jeweils einen Hut auf die Köpfe von A und H.
- A und H können ihren eigenen Kopf nicht sehen aber
- A und H können den Hut der anderen Person sehen.

Nun findet die folgende Diskussion statt:

K: „Abelard, kennst du die Farbe deines Hutes?“

A: „Nein.“

K: „Heloise, kennst du die Farbe deines Hutes?“

H: „Ja.“

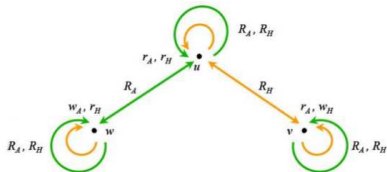
Frage: Welche Farbe hat der Hut von H?

Epistemische Analyse

r_A : Abelard trägt einen roten Hut, r_H : Heloise trägt einen roten Hut
 w_A : Abelard trägt den weißen Hut, w_H : Heloise trägt den weißen Hut

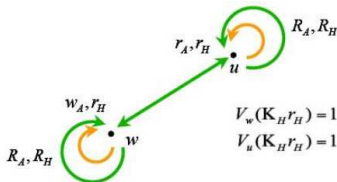
K : „Abelard, kennst du die
Farbe deines Hutes?“

A: „Nein.“



$K_{A}r_A$ ist wahr in v aber falsch
in u

$K_{A}w_A$ ist falsch in w



$$V_w(K_H r_H) = 1$$

$$V_u(K_H r_H) = 1$$

Weniger
Zugänglichkeitsbeziehungen
entsprechen weniger Ignoranz
und somit mehr Wissen

Wahrheit

- Eine atomare aussagenlogische Formel p ist wahr in einer Welt s genau dann, wenn s in der Menge der möglichen Welten ist, die zu p zugewiesen sind:

$$M.s \models p \iff s \in V(p)$$

- Die Formel $K_a\varphi$ ist wahr in einer Welt s ($M, s \models K_a\varphi$) genau dann, wenn für alle t mit $s \sim_a t$ gilt, dass $M, t \models \varphi$

Von Wissen zu Glauben

- Wahrheitsaxiome:
 $K_a\varphi \rightarrow \varphi$: es ist unmöglich, etwas zu wissen, das nicht wahr ist.
- Logik des Glaubens:
 $B_a\varphi$: Agent a glaubt, dass φ wahr ist.
- „Wenn Georg weiß, dass Anna glaubt, dass es regnet, dann weiß Anna, dass Ben es weiß.“ $K_G B_A \text{rain} \rightarrow K_G K_B B_A \text{rain}$
- Wissen impliziert Glauben: $K_i\varphi \rightarrow B_i\varphi$

Beschränkungen des Schlussfolgerns über Andere

- Viele Erwachsene haben Schwierigkeiten, ohne Stift und Papier über mehr als zwei Ordnungen zu schlussfolgern: „Ich weiß nicht, ob du weißt, dass Jan weiß, dass ich weiß, dass . . . “
- Epistemische Logik ist ein idealisiertes Modell des menschlichen Schlussfolgerns über Wissen.
- Aber es kann trotzdem ein sehr nützliches Werkzeug sein.
- Anwendung: Netzwerksicherheit und Kryptographie, Erforschung von sozialem Verhalten und Koalitionen