

Seminar Methoden Wissenbasierter Systeme

Regelbasierte Systeme

RSysteme
Regelbasierte

Überblick

- Was sind Regeln?
- Die Wissensbasis eines regelbasierten Systems
- Inferenz in einem regelbasierten System
- Das Problem der Widersprüchlichkeit
- Erklärungskomponente

RSysteme
Regelbasierte

Überblick

- Was sind Regeln?
- Die Wissensbasis eines regelbasierten Systems
- Inferenz in einem regelbasierten System
- Das Problem der Widersprüchlichkeit
- Erklärungskomponente

RSysteme
Regelbasierte

Was sind Regeln?

Definition: Regeln sind *formalisierte* Konditionalsätze der Form:

Wenn (**if**) A dann (**then**) B

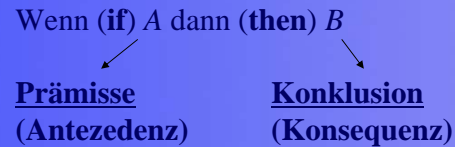
Wenn ^{A-Aussage} A **wahr** (erfüllt, bewiesen) ist
Dann schließe, dass auch ^{B-Aussage} B **wahr** ist

A und B sind **Aussagen**

RSysteme
Regelbasierte

Was sind Regeln?

Definition: Regeln sind *formalisierte* Konditionalsätze der Form:



R Systeme
Regelbasierte

Was sind Regeln?

Definition: Regeln sind *formalisierte* Konditionalsätze der Form:

Wenn (**if**) A dann (**then**) B

Wenn die Prämisse einer Regel erfüllt ist, die Regel also Angewendet werden kann, so sagt man auch, die Regel **feuert**.

Bsp:

Wenn man beim Fremdgehen erwischt wird, dann hat man Pech.

Was sind Regeln?

Definition: Regeln sind *formalisierte* Konditionalsätze der Form:

Wenn (**if**) A dann (**then**) B

➤ **Deterministische Regel**

Eine Regel, die immer gilt, also ohne Ausnahme.

➤ **Implikation:**

$$A \Rightarrow B \equiv \neg A \vee B$$

R Systeme
Regelbasierte

Was sind Regeln?

2 Bedingungen der Form der Regeln

- Die Verknüpfung \vee (oder) darf nicht in der Prämisse einer Regel auftreten
- Die Konklusion einer Regel soll nur aus **einem** Literal, also einem positiven oder negierten Atom, bestehen

Regeln, die diesen beiden Bedingungen nicht genügen, müssen im Bedarfsfalle vereinfacht werden. Hierbei kommt die klassische Logik zum Einsatz

R Systeme
Regelbasierte

Was sind Regeln?

Beispiel: Prämisse und Konklusion einer allgemeinen Regel
if A then B können durchaus komplexe Formeln sein.

Wenn es morgen regnet oder schneit, gehen wir ins
Kino oder bleiben zu Hause

Verletzt beide Bedingungen

- Die Verknüpfung \vee (oder) darf nicht in der Prämisse einer Regel auftreten
- Die Konklusion einer Regel soll nur aus **einem** Literal, also einem positiven oder negierten Atom, bestehen

- ◆ Wenn es morgen regnet und wir nicht ins Kino gehen, dann bleiben wir zu Hause.
- ◆ Wenn es morgen regnet und wir nicht zu Hause bleiben, dann gehen wir ins Kino.

R Systeme
Regelbasierte

Was sind Regeln?

Regelumformungen:

1. Ersetze die Regel

if $K_1 \vee K_2 \vee \dots \vee K_n$ then $D_1 \wedge D_2 \wedge \dots \wedge D_m$

durch die $n \times m$ Regeln

if K_i then D_j $i \in \{1, \dots, n\}, j \in \{1, \dots, m\}$

2. Ersetze die Regel

Konjunktion von Literalen

if K then $L_1 \vee L_2 \vee \dots \vee L_n$

durch die Regeln

if $K \wedge (\bigwedge_{k \neq k_0} \neg L_k)$ then L_{k_0} $k_0 \in \{1, \dots, n\}$

R Systeme
Regelbasierte

Anwendung sehen wir beim später kommenden Beispiel

Überblick

- Was sind Regeln?
- Die Wissensbasis eines regelbasierten Systems
- Inferenz in einem regelbasierten System
- Das Problem der Widersprüchlichkeit
- Erklärungskomponente

R Systeme
Regelbasierte

Die Wissensbasis eines regelbasierten Systems

Die **Wissensbasis** eines regelbasierten Systems enthält *Objekte* und deren Beschreibungen mittels einer (im Allgemeinen) endlichen Menge diskreter *Werte*.

Regeln repräsentieren Zusammenhänge zwischen Objekten oder Mengen von Objekten. Sie haben die Form:

Wenn (if) A dann (then) B

wobei A und B Aussagen über die Objekte sind.

R Systeme
Regelbasierte

Die Wissensbasis eines regelbasierten Systems

Beispiel (Geldautomat):

Parameter	mögliche Werte
Karte	{gültig, ungültig}
PIN	{richtig, falsch}
Versuche	{überschritten, nicht überschritten}
Kontostand	{ausreichend, nicht ausreichend}
Betrag	{ \leq Maximalbetrag, $>$ Maximalbetrag}
Auszahlung	{soll erfolgen, soll nicht erfolgen}
Karterückgabe	{ja, nein}

Regel: **if** Kontostand = *nicht ausreichend*
then Auszahlung = *soll nicht erfolgen*


Die Wissensbasis eines regelbasierten Systems


Beispiel (Geldautomat):


Z.B.  habe

 (Karte) = gültig

 (PIN) = richtig

 (Versuche) = nicht überschritten

 (Betrag) \leq Maximalbetrag

 (Kontostand) = nicht ausreichend

Wegen der Regel:

 (Auszahlung)

Überblick

- Was sind Regeln?
- Die Wissensbasis eines regelbasierten Systems
- **Inferenz in einem regelbasierten System**
 - **Regelnetzwerk**
 - **Vorwärtsverkettung**
 - **Rückwärtsverkettung**
- Das Problem der Widersprüchlichkeit
- Erklärungskomponente

Inferenz in einem Regelbasierten System

Was ist Inferenzregel

Die grundlegende Inferenzregel in einem regelbasierten System ist der *Modus ponens*:

if A then B (Regel)

A wahr (Faktum)

B wahr (Schlussfolgerung)

Die Regel ist dabei Teil des abstrakten Wissens, während das Faktum im Allgemeinen auf Beobachtungen basiert, also Teil des konkreten Wissens ist.

Inferenz in einem Regelbasierten System

Beispiel (Geldautomat):

if *Kontostand = nicht ausreichend*
 then *Auszahlung = soll nicht erfolgen* (Regel)
Kontostand = nicht ausreichend (Faktum)

Auszahlung = soll nicht erfolgen (Schlussfolgerung)



Inferenz in einem Regelbasierten System (Regelnetzwerke)

Regelnetzwerke

Beispiel: Unsere Wissensbasis enthalte die

Objekte: *A,B,C,D,E,F,G,H,I,J,K,L,M*
 Jeweils mit den Werten { *true, false* }

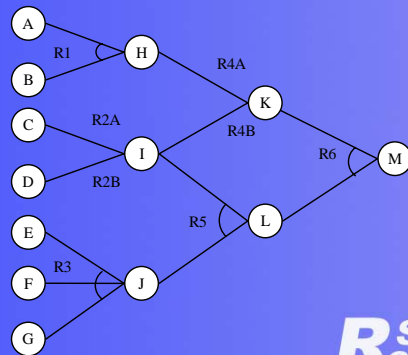
Regeln:

R1: if $A \wedge B$ then H
 R2: if $C \vee D$ then I \Leftrightarrow $\begin{cases} R2a: \text{if } C \text{ then } I \\ R2b: \text{if } D \text{ then } I \end{cases}$
 R3: if $E \wedge F \wedge G$ then J
 R4: if $H \vee I$ then K \Leftrightarrow $\begin{cases} R4a: \text{if } H \text{ then } K \\ R4b: \text{if } I \text{ then } K \end{cases}$
 R5: if $I \wedge J$ then L
 R6: if $K \wedge L$ then M



Inferenz in einem Regelbasierten System (Regelnetzwerke)

Beispiel:

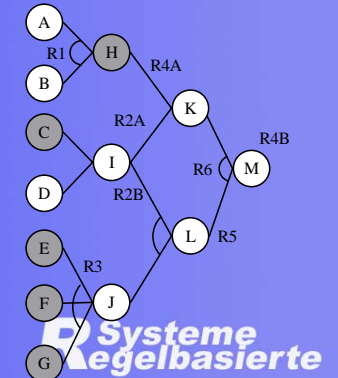


Die Kreisbögen symbolisieren konjunktive Verknüpfungen

Inferenz in einem Regelbasierten System (Vorwärtsverkettung)

Beispiel:

Fakten $F = \{H, C, E, F, G\}$

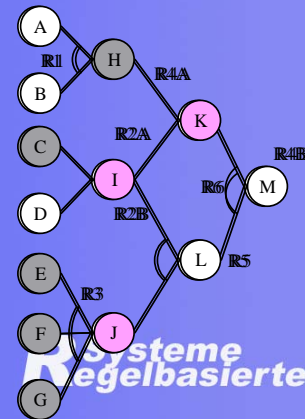


Inferenz in einem Regelbasierten System (Vorwärtsverkettung)

Beispiel:

Fakten $F = \{H, C, E, F, G\}$

$F := \{H, C, E, F, G\} \cup \{I, J, K\}$



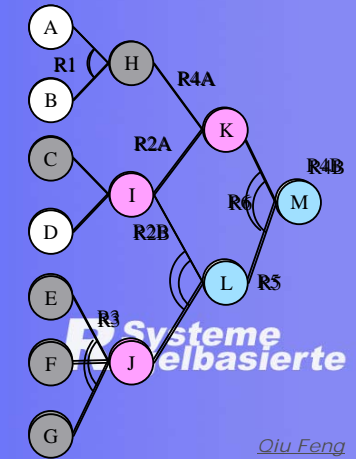
Inferenz in einem Regelbasierten System (Vorwärtsverkettung)

Beispiel:

Fakten $F = \{H, C, E, F, G\}$

$F := \{H, C, E, F, G\} \cup \{I, J, K\}$

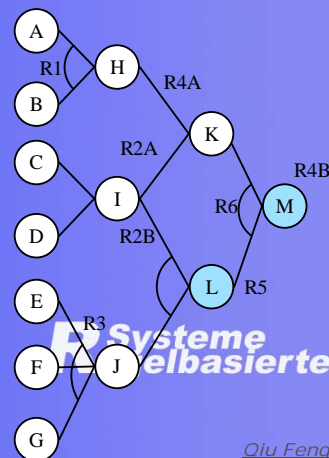
$F := \{H, C, E, F, G, I, J, K\} \cup \{L, M\}$



Inferenz in einem Regelbasierten System (Rückwärtsverkettung)

Beispiel:

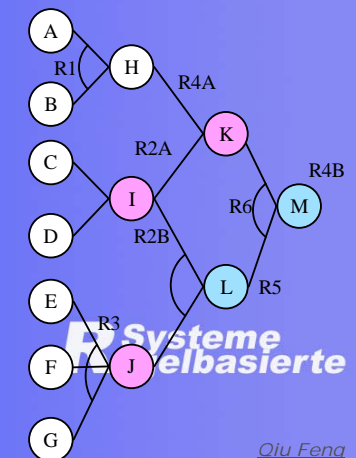
Sei $G = \{L, M\}$



Inferenz in einem Regelbasierten System (Rückwärtsverkettung)

Beispiel:

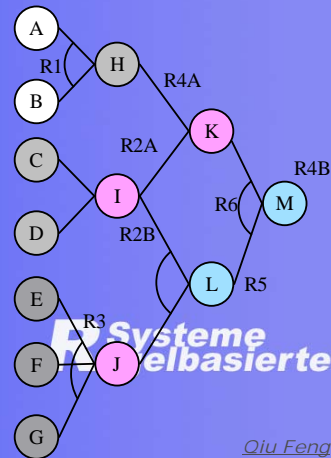
Sei $G = \{L, M\}$



Inferenz in einem Regelbasierten System (Rückwärtsverkettung)

Beispiel:

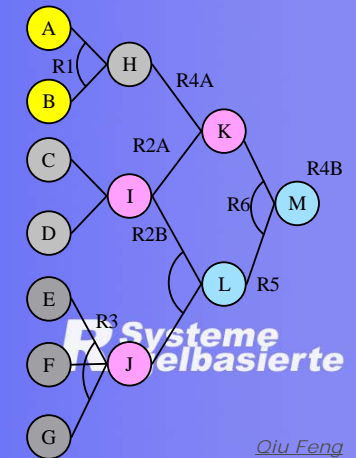
Sei $G = \{L, M\}$



Inferenz in einem Regelbasierten System (Rückwärtsverkettung)

Beispiel:

Sei $G = \{L, M\}$



Überblick

- Was sind Regeln?
- Die Wissensbasis eines regelbasierten Systems
- Inferenz in einem regelbasierten System
- **Das Problem der Widersprüchlichkeit**
- Erklärungskomponente

Das Problem der Widersprüchlichkeit

Im Zusammenhang mit syntaktisch einfachen Regeln haben wir schon das Problem angesprochen, dass die Regelbasis zu widersprüchlichen Ableitungen führen kann, sobald man auch Negation in den Fakten oder in der Konklusion von Regeln erlaubt. Dieses Phänomen tritt in der Praxis häufiger auf als vielleicht vermutet. Experten benutzen zur Ableitung ihrer Schlüsse oft unausgesprochene Annahmen oder übersehen, dass Regeln auch Ausnahmen haben können.

Das Problem der Widersprüchlichkeit

Beispiel:

Wir betrachten die aus den folgenden beiden Regeln bestehende Wissensbasis

if V then F

if $V \wedge P$ then $\neg F$

Sind in einem konkreten Fall V und P wahr, so zieht das System zwei Schlüsse, nämlich F und $\neg F$, die sich widersprechen.

RSysteme
Regelbasierte

Überblick

- Was sind Regeln?
- Die Wissensbasis eines regelbasierten Systems
- Inferenz in einem regelbasierten System
- Das Problem der Widersprüchlichkeit
- **Erklärungskomponente**

RSysteme
Regelbasierte

Erklärungskomponente

Regelbasierte Systeme verfügen im Allgemeinen über eine gute Erklärungsfähigkeit, indem sie die zur Schlussfolgerung Herangezogenen Regeln auflisten und so eine Argumentationskette nachbilden.

RSysteme
Regelbasierte

Erklärungskomponente

Beispiel:

Wir betrachten das Beispiel, das wir bei Vorwärtsverkettung gemacht haben

Objekte: $A, B, C, D, E, F, G, H, I, J, K, L, M$

Jeweils mit den Werten { true, false }

Regeln:

R1: if $A \wedge B$ then H	}	R2a: if C then I
R2: if $C \vee D$ then I		R2b: if D then I
R3: if $E \wedge F \wedge G$ then J	}	R4a: if H then K
R4: if $H \vee I$ then K		R4b: if I then K
R5: if $I \wedge J$ then L		
R6: if $K \wedge L$ then M		

Gegebene Fakten: H, C, E, F, G

RSysteme
Regelbasierte

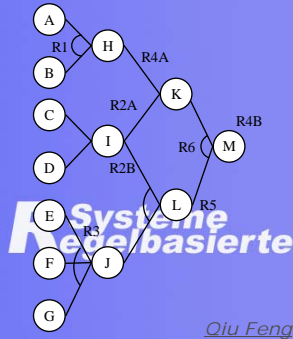
Erklärungskomponente

Beispiel:

- Regeln:
- R1: if $A \wedge B$ then H
 - R2: if $C \vee D$ then $I \Leftrightarrow$
 - R2a: if C then I
 - R2b: if D then I
 - R3: if $E \wedge F \wedge G$ then J
 - R4: if $H \vee I$ then $K \Leftrightarrow$
 - R4a: if H then K
 - R4b: if I then K
 - R5: if $I \wedge J$ then L
 - R6: if $K \wedge L$ then M

Gegebene Fakten: H, C, E, F, G

- Schlussfolgerungen:
- I wegen Regel **R2a**
 - K wegen Regel **R4a**
 - J wegen Regel **R3**
 - L wegen Regel **R5**
 - M wegen Regel **R6**



Systeme Regelbasierte

Vielen Danke für Euere Aufmerksamkeit

Systeme Regelbasierte