

Kapitel MK:V

V. Diagnoseansätze

- Diagnoseproblemstellung
- Diagnose mit Bayes
- Evidenztheorie von Dempster/Shafer
- Diagnose mit Dempster/Shafer
- Truth Maintenance
- Assumption-Based TMS
- Diagnosis Setting
- Diagnosis with the GDE
- Diagnosis with Reiter
- Grundlagen fallbasierten Schließens
- Fallbasierte Diagnose

Grundlagen fallbasierten Schließens

Einleitung

Beispiel 1:



Grundlagen fallbasierten Schließens

Einleitung

Beispiel 1:



Grundlagen fallbasierten Schließens

Einleitung

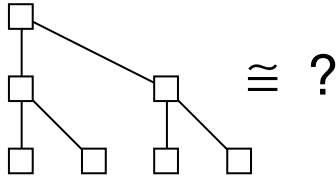
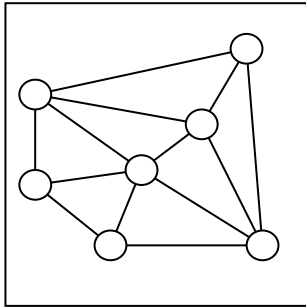
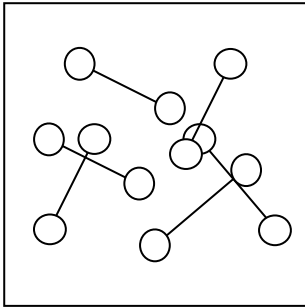
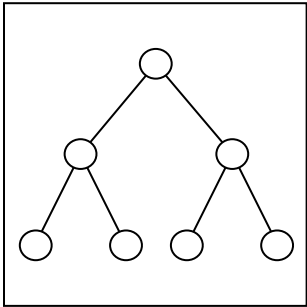
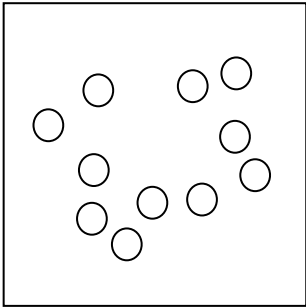
Einige Charakteristika des fallbasierten Schließens im Beispiel:

1. Erinnerung.
Man erinnert sich an eine vergleichbare Situation.
2. Anpassung.
Durch Fallanpassung ist man in der Lage, die gespeicherte Lösung an die neue Situation anzupassen.
3. Anwendung.
Die Lösung kann angewandt werden, auch wenn man nicht verstanden hat, wie sie funktioniert.

Grundlagen fallbasierten Schließens

Einleitung

Beispiel 2:



Grundlagen fallbasierten Schließens

Historie

- 1977** Theoretische Ursprünge des fallbasierten Schließens liegen in der Kognitionswissenschaft [Schank und Abelson 1977]:
Das allgemeine Verständnis des Menschen über Situationen ist eine Art Gedächtnisschema, das sogenannte *Script*.
Zweite wichtige Wurzel des CBR ist das „Analoge Schließen“.
Übertragung von Lösungsprinzipien aus verstandenen Problembereichen auf neue Situationen.
- 1983** Erste Realisierung eines fallbasierten Systems [Kolodner 1983]:
CYRUS beantwortet Fragen über die Tätigkeit des ehemaligen amerikanischen Außenministers Cyrus Vance.
- 2000** Eine Reihe von fallbasierten Systemen ist erfolgreich im Einsatz in der Industrie als auch im Dienstleistungsbereich.

Bemerkungen:

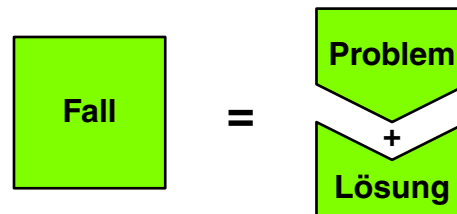
- ❑ Die englische Bezeichnung für fallbasiertes Schließen ist Case-Based Reasoning (CBR).
- ❑ Fallbasiertes Schließen bedeutet „Retrieve and Adapt“ und ist ein dem Menschen abgeschautes Paradigma. Vergleiche hiermit ein anderes wichtiges Schlussfolgerungsparadigma der KI: „Generate and Test“.

Grundlagen fallbasierten Schließens

Definition 21 (Fall, Fallbasis)

Ein Fall ist die Beschreibung einer Problemsituation zusammen mit den Erfahrungen, die während der Bearbeitung des Problems gewonnen werden konnten. Formal gesehen ist ein Fall ein Tupel $\langle P, S \rangle$, wobei P für die Problembeschreibung und S für die Problemlösung steht.

Eine Fallbasis \mathcal{C} ist eine geeignet organisierte Sammlung von Fällen.



Definition 22 (fallbasiertes Schließen [Weß 1996])

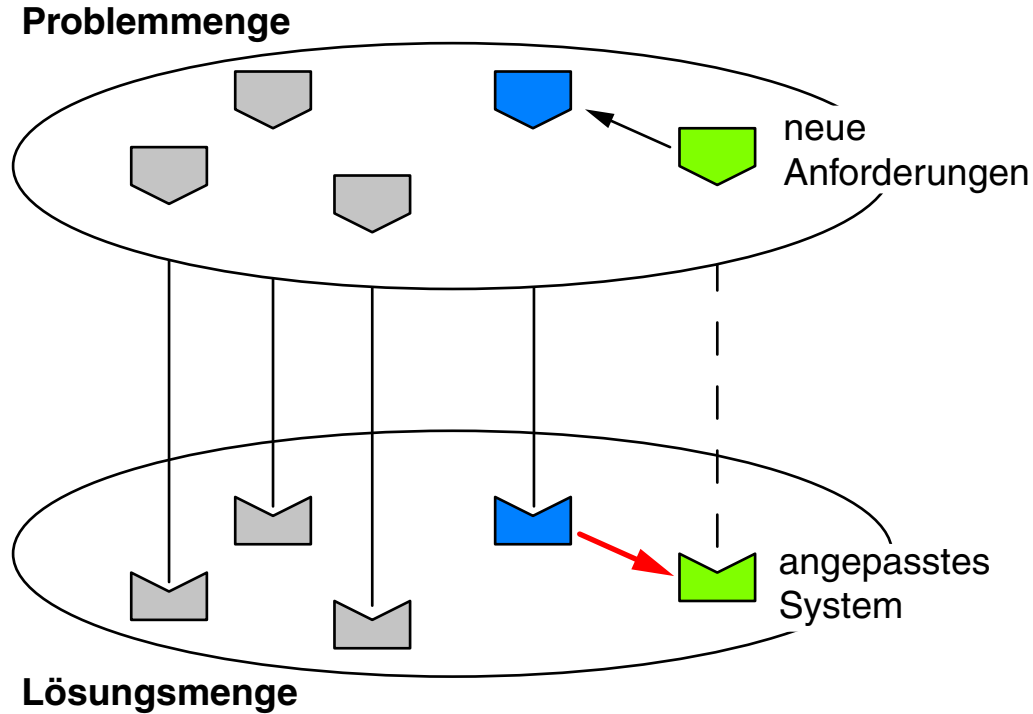
Fallbasiertes Schließen ist ein Ansatz zur Wiederverwendung von spezifischem Problemlösungswissen im Kontext eines aktuell zu lösenden Problems.

Grundlagen fallbasierten Schließens

Repräsentation eines Falles $\langle P, S \rangle \in \mathcal{C}$:

- ❑ **Attribut-Wert-Repräsentation.**
 P und S werden durch eine flache Datenstruktur dargestellt, die aus jeweils einer Liste mit Attribut-Wert-Tupeln besteht.
- ❑ **Objektorientierte Repräsentation.**
 P und S werden als Objekte einer objektorientierten Sprache repräsentiert. Ermöglicht die Darstellung einer Fallhierarchie.
- ❑ **Repräsentation durch Bäume und Graphen.**
Mit diesen Arten von Fallrepräsentationen können u. a. Lösungswege gut dargestellt werden.
- ❑ **Repräsentation durch MOPs und TOPs.**
Im Dynamic-Memory-Model werden Fälle durch MOPs (*Memory Organisation Packets*) und TOPs (*Thematic Organisation Packets*) dargestellt. Mit ihnen soll das menschliche Gedächtnis nachgebildet werden. MOPs als Gedächtnisstrukturen verwalten Informationen, wie Elemente des Gedächtnisses zueinander in Verbindung stehen.

Grundlagen fallbasierten Schließens

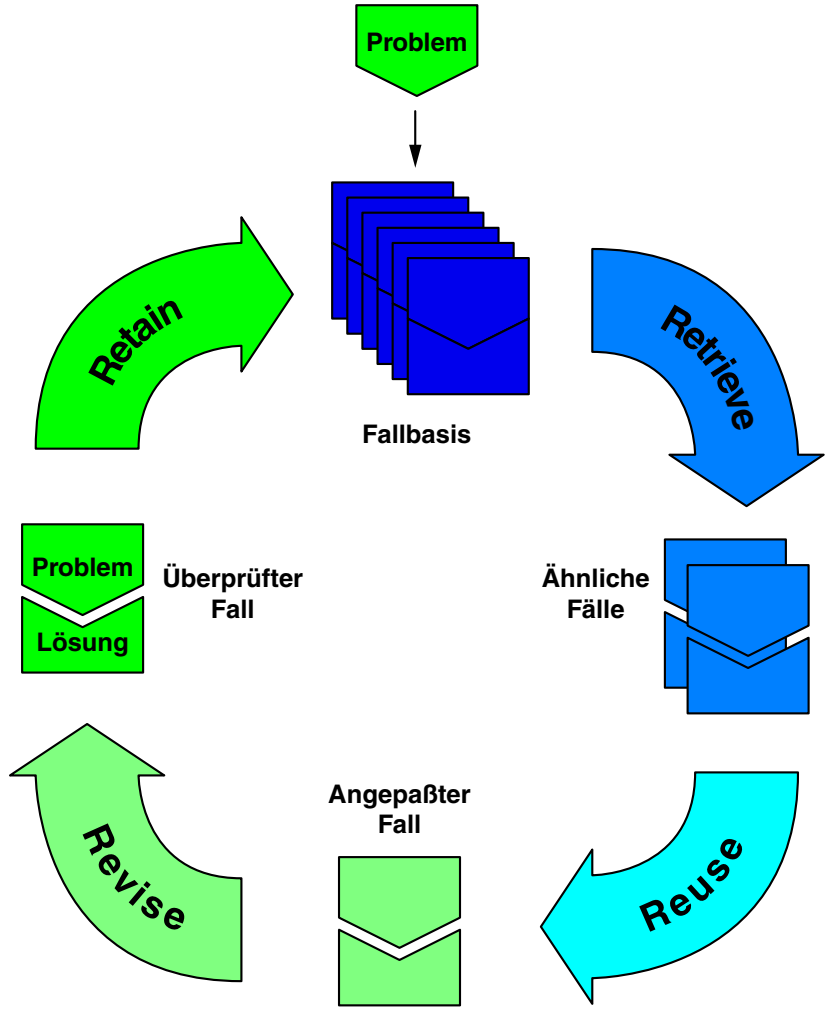


Paradigma des fallbasierten Schließens:

Aus der Ähnlichkeit zweier Probleme darf auf die Ähnlichkeit ihrer Lösungen geschlossen werden.

Grundlagen fallbasierten Schließens

Zyklus des fallbasierten Schließens [Aamodt/Plaza 1994]



Grundlagen fallbasierten Schließens

Zyklus des fallbasierten Schließens (Fortsetzung)

1. Retrieve.

Ein oder mehrere Fälle, die relevant für das gegebene Problem sind, werden aus einer Fallbasis ausgesucht.

2. Reuse.

Nach der Durchführung von eventuell notwendigen Adaptionen wird ein herausgesuchter Fall wiederverwendet.

3. Revise.

Nach einer Evaluierung des adaptierten Falls werden gegebenenfalls weitere Reparaturen durchgeführt.

4. Retain.

Der durch die Schritte 1-3 neu gewonnene Fall – bestehend aus Problem und Lösung – wird der Fallbasis hinzugefügt.

Grundlagen fallbasierten Schließens

Klassen fallbasierter Ansätze [Althoff 1992, Wess 1992]:

1. Fallvergleichende Systeme.

Ein neues Fallbeispiel wird in eine Menge von bekannten Fallbeispiele eingeordnet und so klassifiziert bzw. interpretiert.

Anwendungen:

Klassifikation, Wiedererkennung, Diagnose, rechtswissenschaftliche Fragestellungen

2. Falladaptierende Systeme.

Im Mittelpunkt des Problemlöseprozesses steht die Konstruktion einer neuen, auf die spezifische Situation zugeschnittene Lösung.

Anwendungen:

Arbeitsplanung, Scheduling, Konfiguration, Design

Grundlagen fallbasierten Schließens

	vollständiger Vergleich	partieller Vergleich
keine Anpassung	Datenbanksysteme	fallvergleichende Systeme
Anpassung der Lösung	Expertensysteme	falladaptierende Systeme

Grundlagen fallbasierten Schließens

	vollständiger Vergleich	partieller Vergleich
keine Anpassung	Datenbanksysteme	fallvergleichende Systeme
Anpassung der Lösung	Expertensysteme	falladaptierende Systeme

Was man aus Fällen lernen kann [Carbonell 1983-1986, Weiß 1996]:

1. Transformational Analogy bzw. transformationsorientiert.
Die im gefundenen Fall gespeicherte Lösung wird durch den Einsatz von problemspezifischem Wissen direkt verändert und so auf die aktuelle Problemsituation angepasst.
2. Derivational Analogy bzw. prozessorientiert.
Der zur Lösung des Falls führende Prozess wird im Kontext der aktuellen Problemstellung wiederholt (replay). Die dabei evtl. notwendigen Modifikationen der Lösung werden entsprechend durchgeführt.

Bemerkungen:

- ❑ Die transformationsorientierte Vorgehensweise ist das häufiger angewandte Prinzip.

Grundlagen fallbasierten Schließens

Retrieve

Kern des Retrieve-Schritts ist die Definition der Ähnlichkeit.

Seien die Fälle $C_x, C_y, C_u, C_v \in \mathcal{C}$ mit den Problembeschreibungen x, y, u und v gegeben. Zwei Konzepte zur Ähnlichkeitsbestimmung:

1. Repräsentationsorientierte Verfahren.

(a) Definition über ein Prädikat sim :

$$sim(x, y) = \begin{cases} 1 & y \text{ ist zu } x \text{ ähnlich,} \\ 0 & \text{sonst.} \end{cases}$$

(b) Definition über eine Präferenzrelation R :

$$R(x, y, u, v) \equiv \text{„}x \text{ ist zu } y \text{ mindestens so ähnlich, wie } u \text{ zu } v.\text{“}$$

(c) Erweiterung von R zur Nächsten-Nachbarrelation NN :

$$NN(x, z) \Leftrightarrow \forall y R(x, z, x, y)$$

Grundlagen fallbasierten Schließens

Retrieve (Fortsetzung)

2. Berechnungsorientierte Verfahren.

Die Ähnlichkeit wird implizit durch ein Maß sim ausgedrückt. Für zwei Problembeschreibungen x und y gilt, $sim(x, y) \in [0; 1]$.

Je größer der Wert von $sim(x, y)$, desto ähnlicher sind x und y .

Grundlagen fallbasierten Schließens

Retrieve (Fortsetzung)

Um dem menschlichen Verständnis von Ähnlichkeit nahe zu kommen, sollte ein Ähnlichkeitsmaß zwei Eigenschaften erfüllen:

1. Reflexivität (Selbstähnlichkeit).

$$\text{sim}(x, x) = 1 \quad \text{bzw.} \quad NN(x, x)$$

2. Symmetrie.

$$\text{sim}(x, y) = \text{sim}(y, x) \quad \text{bzw.} \quad NN(x, y) \models NN(y, x)$$

Grundlagen fallbasierten Schließens

Retrieve (Fortsetzung)

Um dem menschlichen Verständnis von Ähnlichkeit nahe zu kommen, sollte ein Ähnlichkeitsmaß zwei Eigenschaften erfüllen:

1. Reflexivität (Selbstähnlichkeit).

$$\text{sim}(x, x) = 1 \quad \text{bzw.} \quad NN(x, x)$$

2. Symmetrie.

$$\text{sim}(x, y) = \text{sim}(y, x) \quad \text{bzw.} \quad NN(x, y) \models NN(y, x)$$

Ähnlichkeitsmaße versus Distanzmaße:

1. Reflexivität: $d(x, x) = 0$
2. Symmetrie: $d(x, y) = d(y, x)$

Bemerkungen:

- Ähnlichkeitsmaße können aus Distanzmaßen abgeleitet werden. Beispiel:

$$\text{sim}(x, y) = 1 - \frac{d(x, y)}{1 + d(x, y)}$$

Grundlagen fallbasierten Schließens

Retrieve (Fortsetzung)

Ähnlichkeiten können in drei Klassen eingeteilt werden:

1. Globale Ähnlichkeit.

Ein globales Ähnlichkeitsmaß ist auf allen Attributen eines Falls definiert.

2. Lokale Ähnlichkeit.

Ein lokales Ähnlichkeitsmaß ist nur auf Teilfälle definiert. Das ist sinnvoll, wenn eine Lösung aus Teilfällen zusammengesetzt werden soll.

3. Oberflächliche Ähnlichkeit.

Die Verwendung einer oberflächlichen Ähnlichkeit dient zur Vorauswahl von potentiell ähnlichen Fällen. Dabei wird die Ähnlichkeit nur auf eine eingeschränkte Menge von Attributen angewendet.

Im Gegensatz zur lokalen Ähnlichkeit wird trotz Einschränkung der betrachteten Attribute der gesamte Fall berücksichtigt.

→ Berechnung sollte effizient durchführbar sein.

Grundlagen fallbasierten Schließens

Retrieve (Fortsetzung)

Ziel:

Zu einer gegebenen Problembeschreibung soll in der Fallbasis \mathcal{C} der Fall mit der ähnlichsten Problembeschreibung gefunden werden. Typische Realisierungen sind:

1. Sequentielles Retrieval.
Alle Fälle der Fallbasis werden sequentiell ausgewählt und unter Verwendung des definierten Ähnlichkeitsmaßes mit der Anfrage verglichen.
2. Retrieval über eine Indexstruktur.
Vergleich mit den Indizes in Datenbanken.
3. Retrieval über Entscheidungsbäume.
Ein populärer Ansatz sind die sogenannten *kd*-Bäume.

Bemerkungen:

- ❑ Es hängt von der konkreten Anwendung ab, ob die Bildung eines Index oder einer Entscheidungsbaumstruktur überhaupt möglich ist.

Grundlagen fallbasierten Schließens

Reuse

Ziel:

Adaption eines Falles – genauer: der Problemlösung eines Falles, um ihn wiederzuverwenden.

Klassifikation nach Art der Adaption:

- ❑ Nulladaption.
Die Lösung des gefundenen Falls wird direkt übernommen.
- ❑ Adaptation durch Substitution.
Durchführung geringfügiger Parameteränderungen.
- ❑ Adaptation durch Transformation.
Hinzufügen und Entfernen von Teilen eines gefundenen Falls.
- ❑ Generative Adaptation.
Nachbildung des Lösungswegs des ähnlichsten Falles.
- ❑ Adaption durch Kombination.
Teile verschiedener Fälle werden zu einer neuen Lösung kombiniert.

Grundlagen fallbasierten Schließens

Reuse

Ziel:

Adaption eines Falles – genauer: der Problemlösung eines Falles, um ihn wiederzuverwenden.

Techniken zur Adaption:

- ❑ Anwendung von Reparaturregeln.
- ❑ Lösen eines Constraint Satisfaction Problems.
- ❑ Metasuchverfahren wie genetische Algorithmen, simulated Annealing, etc.

Grundlagen fallbasierten Schließens

Revise und Retain

Der Revise-Schritt kann nur sinnvoll durchgeführt werden, wenn ein Fall überhaupt evaluierbar ist.

Typische Fragen innerhalb des Revise-Schrittes sind:

- ❑ Enthält der Fall wertvolles Problem-Lösungs-Wissen?
- ❑ Repräsentiert der Fall eine häufige Situation?
- ❑ Ist das im Fall gespeicherte Wissen übertragbar?

Bemerkungen:

- ❑ Der Retain-Schritt stellt die Lernkomponente in einem fallbasierten System dar.
- ❑ Sinnvoll ist eine Erweiterung der Fallbasis nur dann,
 - wenn die Lösung nur durch Adaptionen durch einen Anwender gewonnen werden konnte oder
 - wenn die Lösung zwar automatisch gewonnen wurde, aber die Adaptionsschritte sehr aufwendig waren.

Fallbasierte Diagnose

Beispiel

Falldatenbank C :

Merkmal	Fall1	Fall2	Fall3	Fall4
Alter a	2	10	5	2
Drehzahl d	800	450	850	800
Fassungsvermögen f	4 Kg	4,5 Kg	4 Kg	5 Kg
Trommel dreht sich t	nein	ja	nein	ja
Pumpt ab p	ja	nein	nein	ja
Überschwemmung u	nein	nein	nein	ja
Wasser wird heiß h	ja	ja	nein	manchmal
Lösung	Motor	Wasserpumpe	Sicherung	Schlauch

Fallbasierte Diagnose

Beispiel

Falldatenbank C :

Merkmal	Fall1	Fall2	Fall3	Fall4
Alter a	2	10	5	2
Drehzahl d	800	450	850	800
Fassungsvermögen f	4 Kg	4,5 Kg	4 Kg	5 Kg
Trommel dreht sich t	nein	ja	nein	ja
Pumpt ab p	ja	nein	nein	ja
Überschwemmung u	nein	nein	nein	ja
Wasser wird heiß h	ja	ja	nein	manchmal
Lösung	Motor	Wasserpumpe	Sicherung	Schlauch

Neuer Fall:

Merkmal	neuer Fall
Alter	3
Drehzahl	800
Fassungsvermögen	5 Kg
Trommel dreht sich	nein
Pumpt ab	ja
Überschwemmung	ja
Wasser wird heiß	ja
Lösung	?



Fallbasierte Diagnose

Beispiel (Fortsetzung)

- Ähnlichkeit des Waschmaschinenalters:

$$sim_a(x_a, y_a) = 1 - \left| \frac{x_a - y_a}{x_a + y_a} \right|$$

- Ähnlichkeit der Drehzahl:

$$sim_d(x_d, y_d) = \begin{cases} 1 & : |x_d - y_d| \leq 300 \\ 0 & : \text{sonst} \end{cases}$$

- Ähnlichkeit des Fassungsvermögens:

$$sim_f(x_f, y_f) = 2 - 2 \cdot |x_f - y_f|$$

- Ähnlichkeit des Trommelfehlers:

$$sim_t(x_t, y_t) = \begin{cases} 2 & : x_t = y_t \\ -2 & : \text{sonst} \end{cases}$$

Fallbasierte Diagnose

Beispiel (Fortsetzung)

- Ähnlichkeit des Pumpfehlers:

$$sim_p(x_p, y_p) = \begin{cases} 2 & : x_p = y_p \\ -2 & : \text{sonst} \end{cases}$$

- Ähnlichkeit des Überschwemmungsfehlers:

$$sim_u(x_u, y_u) = \begin{cases} 6 & : x_u = y_u \\ 0 & : \text{sonst} \end{cases}$$

- Ähnlichkeit des Heißwasserfehlers:

$$sim_h(x_h, y_h) = \begin{cases} 4 & : x_h = y_h \\ 2 & : x_h, y_h \in \{\text{ja, manchmal}\} \\ 1 & : x_h, y_h \in \{\text{manchmal, nein}\} \\ 0 & : \text{sonst} \end{cases}$$

Fallbasierte Diagnose

Beispiel (Fortsetzung)

Verrechnung lokaler Ähnlichkeiten zu einem globalen Ähnlichkeitsmaß:

$$\begin{aligned} sim(x, y) = & \frac{sim_a(x_a, x_a) + sim_d(x_d, y_d) + sim_f(x_f, y_f)}{3} \\ & + sim_t(x_t, y_t) + sim_p(x_p, y_p) \\ & + sim_u(x_u, y_u) + sim_h(x_h, y_h) \end{aligned}$$



$sim(M, M_1)$	$sim(M, M_2)$	$sim(M, M_3)$	$sim(M, M_4)$
8.6	0.5	0.8	8.8

Fall 1 und Fall 4 kommen für die Diagnose in Frage.

Unklar ist, wie die Diagnose interpretiert werden darf: Sind beide Defekte (Motor *und* Schlauch) vorhanden, oder widersprechen sich die Fehler?

Bemerkungen:

- ❑ Das Beispiel wirft weitere Fragen auf:
 - Wie verändern sich die Diagnosen, wenn einzelne (letztlich willkürlich gewählte) Funktionen modifiziert werden?
 - Was geschieht, wenn kein Fall vergleichbar ist?
 - Wie erkennt und verhindert man unerhebliche Merkmale (z. B. das Fassungsvermögen bei Waschmaschinen)?
 - Wieviele Fälle braucht man, um verlässliche Aussagen machen zu können?
- ❑ Ein Retain des neuen Falles im Beispiel ist sinnvoll, weil er eine neue Symptomkombination enthält.

Fallbasierte Diagnose

Diskussion

Um Diagnosen mit fallbasierten Systemen zu erstellen, müssen eine Reihe von Voraussetzungen erfüllt sein:

- ❑ Es sind genügend viele erfolgreich gelöste Fälle vorhanden.
- ❑ Die für eine Ähnlichkeitsbewertung relevanten Merkmale sind bekannt.
- ❑ Es lässt sich ein genügend differenziertes Ähnlichkeitsmaß finden.
- ❑ Die bekannten Lösungen lassen sich auf ähnliche Fälle übertragen bzw. in geeigneter Weise modifizieren.

Vorteile fallbasierte Ansätze:

- ❑ Einfacher Wissenserwerb.
Durch das Sammeln von Fällen ist eine ständige, automatische Verbesserung solcher Systeme vorstellbar.
- ❑ Hohe Akzeptanz.
Begründung von vorgeschlagenen Lösungen mit Hilfe der gelösten Fälle.